



# Self-Enhanced Attention for Image Captioning

Qingyu Sun<sup>1</sup> · Juan Zhang<sup>1</sup> · Zhijun Fang<sup>1</sup> · Yongbin Gao<sup>1</sup>

Accepted: 8 January 2024  
© The Author(s) 2024

## Abstract

Image captioning, which involves automatically generating textual descriptions based on the content of images, has garnered increasing attention from researchers. Recently, Transformers have emerged as the preferred choice for the language model in image captioning models. Transformers leverage self-attention mechanisms to address gradient accumulation issues and eliminate the risk of gradient explosion commonly associated with RNN networks. However, a challenge arises when the input features of the self-attention mechanism belong to different categories, as it may result in ineffective highlighting of important features. To address this issue, our paper proposes a novel attention mechanism called Self-Enhanced Attention (SEA), which replaces the self-attention mechanism in the decoder part of the Transformer model. In our proposed SEA, after generating the attention weight matrix, it further adjusts the matrix based on its own distribution to effectively highlight important features. To evaluate the effectiveness of SEA, we conducted experiments on the COCO dataset, comparing the results with different visual models and training strategies. The experimental results demonstrate that when using SEA, the CIDEr score is significantly higher compared to the scores obtained without using SEA. This indicates the successful addressing of the challenge of effectively highlighting important features with our proposed mechanism.

**Keywords** Image captioning · Visual model · Language model · Attention mechanism · CIDEr score

## 1 Introduction

Images and texts are two primary forms of information carriers. Images contain richer and more comprehensive information, but they are relatively abstract and less easily comprehensible. On the other hand, texts can express content more directly and intuitively. Therefore, the aim of image captioning is to enable machines to automatically generate concise and accurate textual descriptions based on the content of the images. Image captioning has extensive applications in diverse fields, including the medical domain, where it can be utilized to automatically identify lesions in organs and tissue structures within Computed Tomography

---

✉ Juan Zhang  
zhang-j@foxmail.com

<sup>1</sup> School of Electronic and Electrical Engineering, Shanghai University of Engineering Science, Shanghai, China

(CT) images and generate concise medical condition descriptions. This technology assists doctors in expediting the diagnostic process, enabling faster and more accurate assessments [1, 3, 4]. Image captioning can also be utilized for analyzing remote sensing images [38] and assisting individuals with visual impairments [40].

Image captioning models are primarily divided into visual and language models. CLIP is a widely adopted visual model in recent years [16, 35], achieved through joint training on images and text data, enabling it to simultaneously comprehend images and textual descriptions. When employing a pre-trained CLIP model, it takes an input image and encodes it into embedding vectors, strengthening the correlation between the image and text. Language models for image captioning have primarily gone through two major stages: the recurrent neural network (RNN) phase and the Transformer phase. RNN is a deep learning architecture specifically designed to handle data with temporal or sequential characteristics. Its design includes recurrent connections, allowing the network to pass information between different time steps [8, 21], effectively capturing temporal dependencies within the data. However, a significant improvement over RNN is the Long Short-Term Memory (LSTM) network. LSTM introduces three pivotal gating units: the forget gate, input gate, and output gate, enabling the model to intelligently decide when to retain, forget, or output information, playing a crucial role in enhancing sequence data modeling performance [47]. RNN's role is to receive image features extracted by a visual model and subsequently transform these features into natural language textual descriptions. This model establishes associations between visual and textual information, achieving cross-modal understanding and generation. However, when dealing with sequential data, especially in the presence of long sequences, the issue of long-term dependency becomes inevitably challenging [32]. While RNN theoretically holds the potential to address these long-term dependencies, in practice, transmitting information from early time steps to subsequent ones becomes exceptionally difficult in the context of extremely lengthy sequences. This is why more advanced sequence models, such as Transformers, have become the mainstream language models, as they are designed to effectively address long-term dependencies [37].

The Transformer model, owing to its distinctive structural characteristics, effectively addresses the issue of long-term dependencies during the training process. For example, the self-attention mechanism in the Transformer allows the model to establish weighted connections between different positions within an input sequence [33]. This enables each position to attend to information from other positions without being constrained by the length of the sequence [5]. As a result, this self-attention mechanism permits the model to establish long-range dependencies between different time steps without encountering the vanishing gradient problem. The Transformer model comprises two main components: the encoder and the decoder [27]. The encoder is constructed by stacking multiple encoder layers, with each layer primarily incorporating Multi-Head Attention and Feed-Forward Networks (FFNs). Similarly, the decoder is composed of multiple stacked decoder layers, each including Mask Multi-Head Attention, Multi-Head Attention, and Feed-Forward Networks (FFNs) [19, 24]. From this perspective, attention mechanisms serve as the core components of the Transformer model, exerting a crucial influence on its overall performance. However, in the task of image captioning, owing to the complexity of the input's visual features, attention mechanisms may struggle to accurately select essential features, potentially resulting in diminished model performance. Addressing this challenge involves pursuing two primary avenues of enhancement. Firstly, opting for more robust visual feature extraction models ensures that the model receives more informative inputs. Secondly, enhancing the language model's comprehension and expressive capabilities enables more effective encoding and association of

visual information. These approaches working in tandem hold promise for improving the model's performance in image description tasks.

In this paper, we introduce an innovative attention mechanism known as SEA to address the issue of accurately emphasizing important features in complex input data. Traditional attention mechanisms typically employ a softmax function to transform similarity scores into attention weights [34], which are then applied to input value vectors to generate attention context vectors. However, this allocation method comes with certain limitations. While normalization allows important features to receive higher weights, these weights cannot exceed 1. Multiplying any feature by a value less than 1 weakens its influence. Furthermore, due to the constraint of weights being limited to 1, the differences between different features are not adequately amplified, making it challenging for the model to better focus on key information. To overcome these limitations introduced by normalization, we introduce the SEA. After obtaining attention weights, we subject them to enhancement. The degree of enhancement depends on the attention weights' inherent capability to emphasize important features. If the attention weights themselves exhibit relatively weak emphasis on important features, the enhancement effect becomes more pronounced. We evaluate the performance of the SEA mechanism using various optimization strategies, including SCST and Camel, in experiments conducted on the COCO dataset. We also compare it with state-of-the-art methods trained on the same dataset. Additionally, we perform result comparisons on the COCO online testing server [43]. Experimental results demonstrate that our proposed solution achieves a new state-of-the-art performance level on the COCO dataset without relying on external data.

The essential objective and commitments of this paper are summarized as follows.

- Firstly, we identify the issues encountered by Transformers in image captioning and propose a novel direction for research in image captioning.
- We propose the SEA as a replacement for the attention mechanism in the Transformer decoder. The SEA aims to address the limitation of the attention mechanism in highlighting important features when dealing with different types of features. We validate the effectiveness of the SEA across multiple visual models and training strategies.
- SEA boosts the baseline models and achieves comparable performance on MS COCO public benchmark with a series of evaluation metrics (BLEU-1, BLEU-2, BLEU-3, BLEU-4, METEOR, ROUGE, CIDEr, SPICE).

## 2 Related Works

### 2.1 Image Captioning

In the context of generating descriptions for images, the traditional approach has involved using a two-step process known as an encoder-decoder architecture. In the early stages of this approach, Convolutional Neural Networks (CNNs) were typically responsible for the initial step, the encoding of visual features [10, 13, 26]. This process, however, faced challenges, particularly in extracting the most crucial information, as image captioning often hinge on a few pivotal objects in the image and the connections between them. To overcome these challenges, Anderson et al. [7].introduced the 'Bottom-Up and Top-Down Attention' model. This model selectively identifies and extracts significant features from a set of region proposals generated by Faster R-CNN, thus enhancing the quality of feature extraction. In a

related development, researchers such as Yang et al. [9], have explored the use of Graph Convolutional Networks (GCN) for image encoding. GCN leverages its graph-based structure to better capture relationships between different objects within an image. In the decoding phase, where the descriptive sentences are generated, there has been a shift away from Recurrent Neural Networks (RNNs). The Transformer architecture has gained prominence in this role, offering improved performance in image description tasks. Moreover, Transformer's influence extends beyond image description alone. Models like CLIP, built upon the Transformer framework, have demonstrated impressive results in encoding images. Leveraging their capacity to effectively manage multimodal data, CLIP has achieved notable success in the domain of image captioning. Furthermore, encoder models like Swin-Transformer have made significant contributions to advancing the state of the art in this field.

## 2.2 Feature Optimization

**Attention mechanisms**, which emphasizes important features and diminishes the influence of less significant ones, plays a pivotal role in image captioning tasks. During the caption generation process, certain words like 'the', 'of', etc., cannot be extracted from visual information alone and rely on the language model. Lu et al. [29], proposed an adaptive attention mechanism, allowing the model to autonomously decide when to rely on visual cues and when to depend solely on the language model. When using CNNs to extract visual features, it can be challenging to extract key information, as image descriptions often require only a few crucial objects in the image and their relationships. Huang et al. [6] proposed the AoA model, which enhances the existing attention mechanisms by incorporating a new attention mechanism. The core idea of the AoA mechanism involves generating an information vector and an attention gate to improve the model's attention mechanism. The information vector stores information from the current context and attention results, while the attention gate determines the relevance and importance of each channel in the information vector. By using element-wise multiplication, the AoA mechanism better captures essential features in the input data.

**Normalization** is a commonly used method to optimize features, with the primary objective of scaling the value ranges of different features to a common scale, typically within the range of 0 to 1 [42]. In the field of deep learning, various normalization techniques are prevalent, including Batch Normalization, Layer Normalization, and Instance Normalization. When discussing feature enhancement, it is essential to mention Adaptive Instance Normalization, which has been extensively used in style transfer research. In the work of Huang et al. [20], they introduced AdaIN (Adaptive Instance Normalization), which, for the first time, enabled real-time arbitrary style transfer. AdaIN finds applications in various image processing domains, such as the research conducted by Ling et al. [25], where AdaIN was employed for image composition. The process of harmonizing composite images can be viewed as a style transfer problem. Additionally, Kim et al. [23] applied style transfer to real-noise denoising by building a denoiser using Adaptive Instance Normalization. They also introduced a transfer learning scheme to transfer knowledge learned from synthetic noise data to the real-noise denoiser.

## 2.3 Training Strategies

Training strategies in image captioning revolve primarily around the selection of the model's loss function. Typically, the cross-entropy loss is widely employed [36] to optimize the model,

measuring the difference between the model's predictions and the ground truth labels. In image captioning tasks, the target labels are the textual descriptions of the images, and the model generates captions as its output. The cross-entropy loss quantifies the accuracy and semantic consistency of the generated captions by computing the cross-entropy between the generated captions and the target descriptions.

In recent years, with advancements in computational power, data availability, and the development of deep learning, reinforcement learning has made significant progress in image understanding. Rennie et al. [2], introduced reinforcement learning into image understanding and proposed the Self-critical Sequence Training (SCST) method. SCST employs the CIDEr metric as a reward measure, assigning scores to each generated sentence. In each training iteration, the generated sentences are compared against a baseline, and the model's parameters are adjusted using gradient-based optimization methods. SCST directly optimizes the quality of the generated captions, making them closer to human reference descriptions compared to traditional cross-entropy training. Thus, adopting SCST in image captioning tasks can lead to improved caption generation performance. Additionally, other training strategies have been explored in recent years, such as the Camel method proposed by Barraco et al. [18].

### 3 Method

In this section, we present the architectural diagram of the self-enhancing attention mechanism in the initial segment. Subsequently, a comprehensive exposition of the underlying principles governing the self-enhancing attention mechanism is provided. The subsequent segment delves into an in-depth exploration of both the encoding and decoding components of the Transformer model, elucidating the seamless integration of the self-enhancing attention mechanism into the decoding phase. Lastly, we expound upon the dual training phases intrinsic to image captioning, encompassing the interplay between cross-entropy training and reinforcement learning. This narrative culminates in a delineation of the intricate orchestration between these two training paradigms.

#### 3.1 Self-Enhanced Attention

As depicted in Fig. 1b, our approach incorporates a self-enhanced attention mechanism as an extension of the self-attention mechanism. This mechanism introduces a unique step in the utilization of attention weights ( $att_i$ ). In our self-enhanced attention mechanism, we apply an additional self-enhancement process to the  $att_i$ , resulting in the generation of reconstructed features  $I_i$  using these modified  $aug\_att_i$ . The internal workings of the self-enhanced attention mechanism can be summarized as follows:

Firstly, we obtain the feature information of each attention matrix and then develop enhancement strategies tailored to each attention matrix.

$$q_i = Q * W_i^q, k_i = K * W_i^k, v_i = V * W_i^v \quad (1)$$

In Eq. (1)  $W_i^q \in \mathbb{R}^{d_m \times d_q}$ ,  $W_i^k \in \mathbb{R}^{d_m \times d_k}$ ,  $W_i^v \in \mathbb{R}^{d_m \times d_v}$  are the parameter matrices of the input linear layers in Fig. 1a, where  $d_q$ ,  $d_k$ , and  $d_v$  are the dimensions of the query (Q), key (K), and value (V) respectively, and  $d_m$  is the dimension of the attention mechanism model. The  $i \in [1, h]$ , where  $h$  represents the number of attention heads in the multi-head attention mechanism. By adjusting the mapping of the matrices, we obtain the inputs  $q_i$ ,  $k_i$ ,  $v_i$  for our self-enhanced attention mechanism.

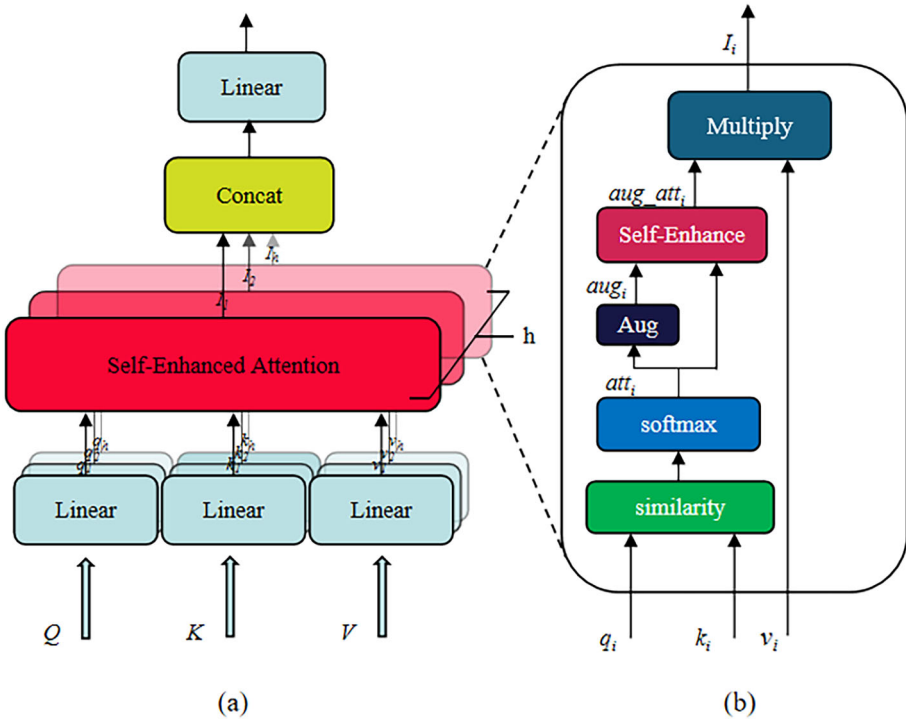


Fig. 1 A illustration of Multi-Head Enhanced Attention Mechanism (a) and Self-enhanced attention model (b)

$$att_i = softmax\left(\frac{q_i k_i^T}{\sqrt{d_k}}\right) \tag{2}$$

In Eq. (2), we compute the similarity between  $q_i$  and  $k_i$  using the dot product operation and obtain an attention vector that represents the weight distribution by applying *softmax* normalization.

$$aug_i = 1 - max(att_i) + min(att_i) \tag{3}$$

In normal attention mechanisms, we directly use  $att_i$  to reconstruct  $v_i$ . However, in our self-enhanced attention mechanism, we further process  $att_i$ . We cannot blindly amplify the  $att$  matrix as it would decrease the stability of the model. Instead, we determine the degree of amplification, referred to as  $aug_i$  based on the distribution characteristics of the  $att_i$  parameters. In Eq. (3) we use the maximum and minimum values as adjustment factors. If the maximum value in the  $att_i$  matrix is already large, it indicates that the  $att_i$  has already selected the most important features. If the maximum value is small while the minimum value is relatively large, it suggests that the  $att_i$  has not effectively identified important features, thus requiring forced intervention.

$$aug\_att_i = ((e^{aug_i} + 1) * a) * att_i \tag{4}$$

$$I_i = aug\_att_i * v_i \tag{5}$$

We substitute  $aug_i$  into Eq. (4), resulting in  $aug\_att_i$ . Here,  $a$  represents the suppression factor introduced to prevent excessive enhancement. Excessive enhancement can lead to significant variations in training results, causing instability and extremely slow training progress. We utilize  $aug\_att_i$  to reconstruct the  $v_i$  and obtain the output  $I_i$  of the self-enhanced attention mechanism.

$$MH\_EH(Q, K, V) = Concat(I_1, \dots, I_h) * W^I \tag{6}$$

In Eq. (6),  $W^I \in \mathbb{R}^{hd_v \times d_m}$  is the parameter matrix of the final output linear layer in Fig. 1a. The Multi-Head Enhanced Attention Mechanism ( $MH\_EH$ ) itself is composed of multiple self-enhanced attention mechanisms, we concatenate the  $[I_1, I_1, \dots, I_h]$  together to obtain the final output.

### 3.2 Image Captioning Encoder

Initially, we utilize a pre-trained CLIP model to extract visual features  $F = [f_1, \dots, f_k]$ , where  $f_i \in \mathbb{R}^{1 \times d}$  and  $d$  represents the dimension of the input vectors, from an image I. The visual features F are then processed through an encoder. The encoder consists of multiple identical Encoder Layers, each composed of a multi-head attention mechanism and a feed-forward network (FFN).

The multi-head attention mechanism enhances the representation capacity by using multiple independent attention heads in parallel. Each attention head has its own query matrix ( $W_i^q$ ), key matrix ( $W_i^k$ ), and value matrix ( $W_i^v$ ). This allows the model to learn different types of feature representations by combining different attention weights. Compared to a single attention mechanism, the multi-head attention mechanism is more expressive as it can capture different aspects of the input simultaneously, the multi-head attention is a combination of multiple self-attention mechanisms.

In the self-attention mechanism, the queries (Q), keys (K), and values (V) are identical and derived from the input visual features ( $Q = K = V = F$ ). We utilize self-attention mechanism to reconstruct the input visual features  $F$  and establish long-term dependencies among different features. The outputs of multiple self-attention mechanisms are concatenated to obtain the result of the multi-head attention mechanism. Subsequently, the obtained result is fed into the FFN layer for non-linear transformation and mapping:

$$FFN(x) = max(0, xW_1 + b_1)W_2 + b_2 \tag{7}$$

where  $W_1 \in \mathbb{R}^{d_m \times d_{ff}}$ ,  $W_2 \in \mathbb{R}^{d_{ff} \times d_m}$ ,  $b_1 \in \mathbb{R}^{d_{ff}}$ , and  $b_2 \in \mathbb{R}^{d_m}$  are learnable parameter matrices. FFN applies independent transformations to each position of the input  $x$  using two fully connected layers. This allows for better capturing of local features within the F. This constitutes the complete algorithm for an Encoder Layer. By processing the F through multiple identical Encoder Layers, the final output ECF is generated.

### 3.3 Image Captioning Decoder

During the training stage, it is necessary to input the authentic textual descriptions of the image to aid the decoder in extracting context information from the ECF.

As shown in Fig. 2, the structure of the decoder is similar to that of the encoder, as it is also composed of multiple Decoder Layers stacked together. In the decoder, we employ two attention mechanisms to extract important information from ECF. These two attention

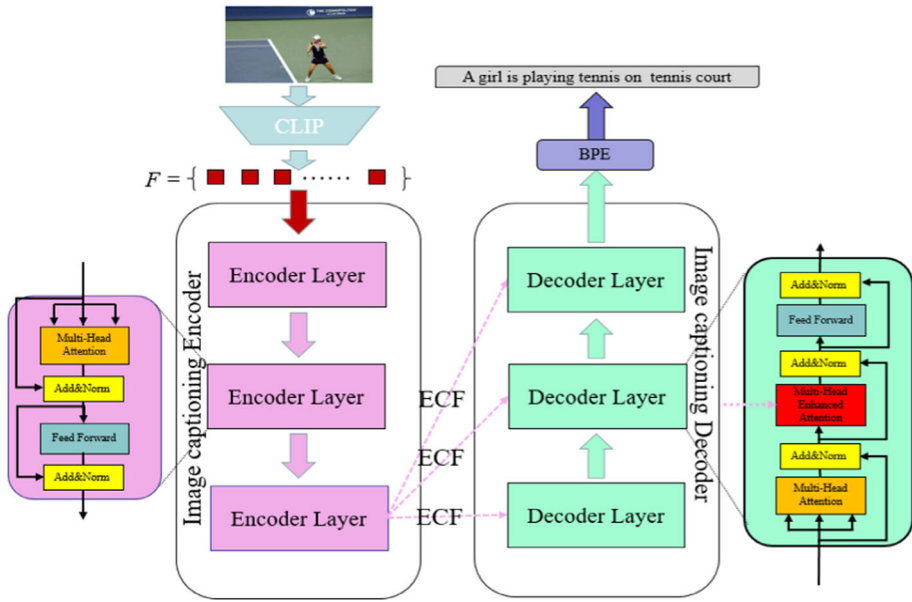


Fig. 2 The overall structure of the image captioning

mechanisms are the self-attention mechanism and the self-enhanced attention mechanism. The self-attention mechanism is responsible for encoding the generated context information by capturing long-term dependencies among the information. It encodes context information through self-encoding. On the other hand, the self-enhanced attention mechanism utilizes context information to extract crucial features from the ECF, helping the decoder better understand the input image’s features and generate semantically correct descriptions related to the image. Similarly, the decoder also includes an FFN layer to perform non-linear transformations and mappings on the outputs of the multi-head attention mechanism. Finally, we employ a learned linear transformation and a softmax function to transform the output from the decoder into a pair of predicted probabilities, denoted as

$$prob(w_{\tau}|w_{k<\tau}, F, \theta) \tag{8}$$

In Eq. (8),  $\{w_{\tau}\}_{\tau}$  is the sequence of words comprising the generated description,  $\tau$  indicates time,  $\theta$  indicates the set of parameters of the model.

### 3.4 Training Strategies

The entire training procedure comprises two stages. In the first training stage, we optimize the model by minimizing the joint cross-entropy (XE) loss, computed as follows:

$$L_{xe}(\theta) = - \sum_{\tau} \log prob(w_{\tau}|w_{k<\tau}, F, \theta) \tag{9}$$

At the second training stage, we jointly finetune the model using self-critical sequence training(SCST) approach. In SCST, the choice of a specific evaluation metric as the reward for fine-tuning the model is available. The CIDEr metric is commonly selected as the reward. CIDEr considers factors such as the theme and information richness of the sentence, making



it a more interpretable and accurate reflection of the quality of generated sentences compared to other metrics. During the cross-entropy training phase, the results with the highest CIDEr score are saved as the baseline for SCST. The core concept of the SCST approach is to utilize the reward obtained by the current model to serve as a baseline for the REINFORCE algorithm. In contrast to traditional cross-entropy training, SCST enables a more direct optimization of the quality of the generated descriptions, making them closer to the reference descriptions provided by humans.

$$L_{scst}(\theta) = -\frac{1}{k} \sum_{i=1}^k ((r(w^i) - b) \nabla_{\theta} \log \text{prob}(w^i)) \tag{10}$$

In Eq. (10), where  $w^i$  is the  $i$ -th sentence in the beam, which refers to a parameter in the beam search algorithm [7]  $r(\bullet)$  is the reward function, and  $b = (\sum_i r(w^i))/k$  is the baseline, computed as the mean of the rewards obtained by the sampled sequences.

**Train with Camel** In addition to the commonly used training strategies mentioned above, we have introduced the "Camel" training strategy. Camel can be applied in both the cross-entropy training phase and the SCST training phase. In the Camel training strategy, the language model is divided into two networks: the online network and a target network. The online network functions as the student network, while the target network serves as the teacher network. Throughout the training process, the target model remains fixed, and knowledge transfer from the target network to the online network is achieved through a specific knowledge distillation technique. In this experiment, we utilize mean squared error minimization, as expressed by the following formula:

$$\min_{\theta_o} \sum_{\tau} (\text{prob}_{i,\tau} - \text{prob}_{o,\tau})^2 \tag{11}$$

where  $\theta_o$  indicates the set of parameters of the online network. Therefore, after employing the Camel training strategy, the cross-entropy training phase is enhanced.

$$L_{xe}(\theta_o)' = \left( \min_{\theta_o} \sum_{\tau} (\text{prob}_{i,\tau} - \text{prob}_{o,\tau})^2 \right) * dlw + L_{xe}(\theta_o) \tag{12}$$

$$L_{scst\_o}(\theta_o)' = L_{scst\_o}(\theta_o) + \left( \min_{\theta_o} \sum_{\tau} (\text{prob}_{i,\tau} - \text{prob}_{o,\tau})^2 \right) * dlw \tag{13}$$

where  $dlw$  is a hyperparameter used to adjust the distillation weight, thereby controlling the relative weighting between the two loss terms, striking a balance between them. A larger distillation weight places more emphasis on the knowledge from the teacher model, whereas a smaller distillation weight prioritizes the self-training loss of the student model. Finally, the target network's parameters are updated according to the following formula:

$$\theta_t \leftarrow \beta \theta_t + (1 - \beta) \theta_o \tag{14}$$

where  $\theta_t$  indicates the set of parameters of the target network and  $\beta \in [0, 1]$  is a target decay rate.

## 4 Experiments

### 4.1 Datasets and Metrics

In this work, we used the COCO dataset [44], which is a publicly image dataset developed under the lead of Microsoft Corp. The dataset consists of over 100,000 images and 250,000

image annotations. These images were sourced from various internet sources and encompass a wide range of scenes and objects. The dataset covers 90 different categories of objects, including people, animals, vehicles, furniture, food, and more. Each image in the dataset is associated with a minimum of 5 manually annotated descriptions, each consisting of 5 to 40 words. For our experiments, we followed the Karpathy splits [45], where 113,287 images were used for training, 5,000 images for validation, and another 5,000 images for testing.

To assess the quality of the generated captions, we employed the standard evaluation script, which computes widely used automatic evaluation metrics. These metrics include Bilingual Evaluation Understudy (BLEU-1/4) [11], Metric for Evaluation of Translation with Explicit Ordering (METEOR) [14], Recall-Oriented Understudy for Gisting Evaluation (ROUGE) [12], Specific Image Caption Evaluation (SPICE) [17], and Consensus-based Image Description Evaluation (CIDER) [15], denoted as B-1/4, M, R, S, and C, respectively, for brevity. These metrics provide quantitative measures to assess the quality and similarity of the generated captions against the ground truth annotations.

## 4.2 Implementation Details

**Data Processing** For preprocessing the COCO image dataset used as input to the visual model, all images were uniformly resized to a resolution of  $384 \times 384$  pixels. Image captioning can be viewed as a form of machine translation where the input is an image. Consequently, when generating captions, there is a challenge of dealing with unknown or rare vocabulary words. Therefore, we employ the BPE (Byte-Pair Encoding) algorithm to address this issue. technique [46]. BPE is a subword tokenization method that segments words into subword units based on their frequency within the training corpus. By breaking words into smaller subword units, BPE facilitates handling out-of-vocabulary words and reduces the overall vocabulary size.

**Parameters in image captioning model** We set the number of layers in the encoder and decoder of the Transformer to 3, denoted as  $N = 3$ . During the training process, we introduced the Mesh-memory Transformer, for which the number of layers for Self-Enhanced Attention in each decoder is also set to 3. For the multi-head attention mechanism in the Transformer, we utilized 8 attention heads ( $h = 8$ ). The dimension of the input features, denoted as  $d_m$ , was set to 512, allowing for a comprehensive representation of information. To capture more intricate details, we set the dimension of the feed-forward layer, denoted as  $d_{ff}$ , to 2048.

During the training phase, we employed the Adam optimizer to optimize our model. We utilized a beam size of 5 for decoding during inference. The training was conducted on an RTX 3090 device, where the GPU memory usage was notably higher during the SCST fine-tuning phase compared to the cross-entropy learning phase. Therefore, we set the batch size to 100 for the cross-entropy learning phase and 30 for the SCST phase. Additionally, we employed the Mean Teacher training strategy from Camel and designed the distillation weight parameter. For the cross-entropy phase, the  $dlw$  is set to 0.1, and for the SCST phase, the distillation weight is set to 0.01. We referenced the Camel [18] for guidance in designing the remaining parameters used in our study.

**Table 1** Results on the COCO Karpathy-test split with the replacement of different attention mechanisms in various parts of the Transformer using SEA. The “Base” experiment serves as the baseline, with our visual model being CLIP-RN50×16. Enc(+ SEA) denotes the substitution of self-attention in the Transformer’s Encoder section with self-enhanced attention, while Dec(+SEA) signifies the replacement of attention mechanisms in the Transformer’s Decoder section with self-enhanced attention

Model	B-1	B-4	M	R	C	S
Base	77.5	37.6	29.0	58.0	122.6	21.9
Enc(+ SEA)	78.0	38.7	29.3	58.5	125.2	22.2
Dec(+ SEA)	78.7	39.3	29.6	58.9	126.8	22.5

### 4.3 Ablation Study

#### 4.3.1 Effect of Self-Enhanced Attention in Different Parts

We replaced the attention mechanisms in different sections of the Transformer with the SEA, as illustrated in Table 1. Subsequent to the replacement, the final results exhibited varying degrees of enhancement. In the case of Enc(+ SEA), the CIDEr score increased from 122.6 to 125.2, while in the case of Dec(+ SEA), the CIDEr score increased from 122.6 to 126.8. From the results, it is evident that replacing the attention mechanism in the Decoder section led to improved model performance. This can be attributed to the distinct Q and K employed in the attention mechanism of the Decoder, introducing a certain level of diversity among features. In contrast, the Encoder utilizes self-attention mechanisms where the input Q and K are identical, resulting in less divergence among features. Therefore, we can preliminarily conclude that the SEA is more suitable for handling diverse input features to a certain extent.

#### 4.3.2 Effect of Different Visual Models

We utilized the multi-modal CLIP model to extract image features. The CLIP model is a deep learning model based on the Transformer architecture, which comprises a multi-layer Transformer encoder and a text embedding module. The visual encoders in CLIP can be either ViT-like or CNN-like architectures. Therefore, in our experiments, we employed CLIP-ViT-B16, CLIP-ViT-B32, CLIP-RN50 × 4, and CLIP-RN50 × 16. Due to the significantly improved performance of CLIP-based models in the image captioning domain compared to Faster-RCNN, we did not include Faster-RCNN in our experiments. We evaluated the performance of these models both with SEA and using the Camel training strategy. It’s worth noting that CLIP-ViT-B32 in the table signifies the use of the CLIP-ViT-B32 visual model only, with the language model part being the basic Transformer.

The results from Table 2 indicate that CLIP-ViT-B32(+ SEA) achieved a Cider score of 118.7, CLIP-ViT-B16(+ SEA) scored 124.7, CLIP-RN50 × 4(+ SEA) scored 123.9, and CLIP-RN50 × 16(+ SEA) scored 126.8. This suggests that the performance of these models improved when SEA was applied. After incorporating the Camel training strategy, the enhancement in performance for CLIP-ViT-B32(+ SEA), CLIP-ViT-B16(+ SEA, + Camel), and CLIP-RN50 × 4(+ SEA, + Camel) was not particularly noticeable. However, for CLIP-RN50 × 16(+ SEA) using the Camel training strategy, there was a significant improvement, with the score increasing from 126.8 to 129.6.

**Table 2** Results on the COCO Karpathy-test split with different visual models when training with cross-entropy loss, the notation + SEA indicates the use of self-enhanced attention mechanism, and + Camel signifies the utilization of the Camel training strategy during the training process

Model	B-1	B-4	M	R	C	S
CLIP-ViT-B32	76.1	36.4	28.3	57.0	118.0	21.0
CLIP-ViT-B32(+ SEA)	76.6	37.0	28.3	57.2	118.7	21.3
CLIP-ViT-B32(+ SEA, + Camel)	76.5	36.6	28.4	57.1	118.8	21.3
CLIP-ViT-B16	77.7	38.4	29.3	58.2	124.0	22.0
CLIP-ViT-B16(+ SEA)	78.2	38.5	29.2	58.3	124.7	22.1
CLIP-ViT-B16(+ SEA, + Camel)	78.5	38.8	28.9	58.5	124.8	21.8
CLIP-RN50 × 4	77.3	37.7	28.6	57.7	121.1	21.4
CLIP-RN50 × 4(+ SEA)	77.5	38.0	28.9	58.0	122.9	21.7
CLIP-RN50 × 4(+ SEA, + Camel)	77.3	37.9	28.9	57.9	121.9	21.7
CLIP-RN50 × 16	77.5	37.6	29.0	58.0	122.6	21.9
CLIP-RN50 × 16(+ SEA)	78.7	39.3	29.6	58.9	126.8	22.5
CLIP-RN50 × 16(+ SEA, + Camel)	<b>79.6</b>	<b>39.9</b>	<b>29.7</b>	<b>59.2</b>	<b>129.6</b>	<b>22.9</b>

**Table 3** Performance on the COCO Karpathy-test split and CLIP-RN50 × 16 (+ SEA, + Camel) when trained with cross-entropy loss at different  $a$  values

$a$	B-1	B-4	M	R	C	S
0.6	78.8	39.3	29.5	58.9	126.9	22.4
0.8	78.7	39.3	29.7	58.9	127.1	22.6
1.0	<b>79.6</b>	<b>39.9</b>	<b>29.7</b>	<b>59.2</b>	<b>129.4</b>	<b>22.9</b>
1.2	78.7	38.7	29.4	58.7	125.7	22.4
1.4	78.4	38.9	29.2	58.6	125.4	22.2

### 4.3.3 Effect of Fusion Function With Different $a$

The parameter  $a$  is a hyperparameter in our study, with the aim of improving experimental results by fine-tuning its value. In our experiments, we conducted tests using five different settings for the group sizes, denoted as  $a \in \{0.6, 0.8, 1.0, 1.2, 1.4\}$ . By examining the data in Table 3, we observed that when  $a = 0.8$ , the CIDEr score was 127.1, and when  $a = 1.4$ , the CIDEr score decreased to 125.4. The highest CIDEr score of 129.4 was obtained when  $a = 1$ . Based on these findings, we can conclude that the experimental results are optimal when  $a$  is close to 1. Deviating too far from this value in either direction results in a decrease in performance.

## 4.4 Comparison with the State of the Art

**Performance on COCO.** We conducted a comparison of Self-Enhanced Attention with several recent image captioning models, including those based on RNNs such as LSTM,

**Table 4** Performance comparisons of different image captioning models on the MSCOCO Karpathy test split





	Cross-entropy loss										CIDEr score optimization									
	B-1	B-2	B-3	B-4	M	R	C	S	B-1	B-2	B-3	B-4	M	R	C	S				
LSTM[47]	-	-	-	29.6	25.2	52.6	94.0	-	-	-	-	31.9	25.5	54.3	106.3	-				
SCST[2]	-	-	-	30.0	25.9	53.4	99.4	-	-	-	-	34.2	26.7	55.7	114.0	-				
LSTM-A[48]	75.4	-	-	35.2	26.9	55.8	108.8	20.0	78.6	-	-	35.5	27.3	56.8	118.3	20.8				
RNet[41]	76.4	60.4	46.6	35.8	27.4	56.5	112.5	20.5	79.1	63.1	48.4	36.5	27.7	57.3	121.9	21.2				
GCN-LSTM[31]	77.3	-	-	36.2	27.9	57.0	116.3	20.9	80.5	-	-	38.2	28.5	58.3	127.6	22.0				
SGAE[9]	77.6	-	-	36.9	27.7	57.2	116.7	20.9	80.8	-	-	38.4	28.4	58.6	127.8	22.1				
AoANet[6]	77.4	-	-	37.2	28.4	57.5	119.8	21.3	80.2	-	-	38.9	29.2	58.8	129.8	22.4				
X-Transformer[28]	77.3	61.5	47.8	37.0	28.7	57.5	120.0	21.8	80.9	65.8	51.5	39.7	29.5	59.1	132.8	23.4				
Camel[18]	78.0	-	-	38.8	29.4	58.6	125.0	22.2	82.7	-	-	40.9	30.3	60.1	138.9	24.5				
SEA	<b>79.6</b>	<b>64.4</b>	<b>50.9</b>	<b>39.9</b>	<b>29.7</b>	<b>59.2</b>	<b>129.4</b>	<b>22.9</b>	<b>83.0</b>	<b>68.1</b>	<b>53.7</b>	<b>41.5</b>	<b>30.3</b>	<b>60.4</b>	<b>141.2</b>	<b>24.3</b>				

SCST, LSTM-A, AoANet, X-Lan as well as models based on Transformers like ETA, M<sup>2</sup>T, X-Transformer, RFNet. Additionally, we considered visual models based on GCN, such as GCN-LSTM and SGAE, and a training strategy utilizing Mean-Teacher called Camel.

As shown in Table 4, our SEA model consistently outperforms all the compared models on the COCO dataset, both in the cross-entropy stage and the SCST stage. In the cross-entropy stage, our model achieves an impressive CIDEr score of 129.4, while in the SCST stage, the CIDEr score further increases to 141.2. Furthermore, the generated captions from our model, as illustrated in Fig. 3, demonstrate accurate capturing of the content depicted in the images.

**Online evaluation.** To further demonstrate the feasibility and effectiveness of our model, we conducted additional evaluations on the official test split of the dataset where ground-truth annotations are not publicly available. We utilized our model to generate image captions for the Val and Test splits locally and then submitted the results to the online COCO test server [45].

In Table 5, we present the performances of our model with respect to 5 reference captions (c5) and 40 reference captions (c40), compared to the top-performing approaches on the COCO leaderboard. Notably, our method outperforms the current state of the art across all evaluation metrics. We achieved a significant advancement of 0.4 CIDEr points compared

Input image	References	Caption
	<p>A little girl holding a kitten next to a blue fence.            Girl in a tank top holding a kitten in her back yard.            A young girl is holding a small cat.            Girl with a yellow shirt holding a small cat.            A girl smiles as she holds a kitty cat.</p>	a young girl holding a kitten in front of a bike.
	<p>Two giraffes standing next to each other on a grassy field.            Two giraffes stroll past each other near a bush.            Two giraffes standing near trees in a grassy area.            Two giraffes rub their necks together as they stand by the trees in the sunlight.            Two giraffes crossing paths on a green and grassy field.</p>	two giraffes standing next to each other in a field
	<p>A woman rides a bicycle on a road next to the median.            A girl is riding her bike down the street.            A lady riding her bicycle on the side of a street.            A person on a bike riding on a street.            A woman riding a bike down a street next to a divider.</p>	a woman riding a bike down the street.
	<p>An elegant bathroom features a tub, sink, mirror, and decorations.            An old fashion above ground tub is shown with gold feet.            A lovely, vintage-styled bathroom with a great claw-footed tub.            Bathroom with a pedestal sink and claw foot bathtub            A claw foot tub is in a large bathroom near a pedestal sink.</p>	a bathroom with a large tub and a sink.

**Fig. 3** Examples of captions generated by our Self-Enhanced Attention

**Table 5** Leaderboard of the published state-of-the-art image captioning models on the MSCOCO online testing server

	BLEU-1		BLEU-2		BLEU-3		BLEU-4		MEREOR		ROUGE		CIDER	
	c5	c40	c5	c40	c5	c40	c5	c40	c5	c40	c5	c40	c5	c40
SCST[2]	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	116.7
Up-Down[7]	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 [41]	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
SGAE [32]	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
ETA[30]	81.2	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
AoANet[6]	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
M <sup>2</sup> -T[22]	81.6	96.0	66.4	90.8	51.8	82.7	39.7	72.8	29.4	39.0	59.2	74.8	129.3	132.1
X-Transformer[28]	81.9	95.7	66.9	90.5	52.4	82.5	40.3	72.4	29.6	39.2	59.5	75.0	131.1	133.5
GET[39]	81.6	96.1	66.5	90.9	51.9	82.8	39.7	72.9	29.4	38.8	59.1	74.4	130.3	132.5
REStNet[49]	82.1	96.4	67.0	91.3	52.2	83.0	40.0	73.1	29.6	39.1	59.5	74.6	131.9	134.0
Camel[18]	82.6	96.8	67.5	91.9	52.8	83.9	40.5	73.8	29.9	39.4	59.8	74.9	135.1	137.7
SEA	<b>83.1</b>	<b>96.9</b>	<b>67.7</b>	<b>92.1</b>	<b>53.2</b>	<b>84.3</b>	<b>40.3</b>	<b>74.4</b>	<b>30.1</b>	<b>39.7</b>	<b>59.9</b>	<b>75.3</b>	<b>135.5</b>	<b>138.0</b>

to the best performing approach, demonstrating the superiority of our model in generating high-quality image captions.

## 5 Conclusion

In this paper, we introduce a novel self-enhanced attention mechanism to replace the conventional self-attention mechanism in the image Transformer decoder. The effectiveness of our proposed model is validated through several experiments.

First, we evaluate the performance of our model with and without the self-enhanced attention mechanism using different CLIP visual models. The results consistently demonstrate that the inclusion of the self-enhanced attention mechanism significantly improves the model's performance. Notably, the best performance is achieved when using the CLIP-RN50  $\times$  16 model, attaining an impressive CIDEr score of 126.8. Furthermore, with the implementation of the Camel training strategy, the CIDEr score saw a further improvement, increasing from 125.4 to 129.4. Finally, we evaluate the performance of our self-enhanced attention mechanism under the SCST paradigm and submit the generated captions to the online test server. Our model surpasses the current state-of-the-art method, showcasing its superiority in generating high-quality image captions. However, it's worth mentioning that our approach, though successful, may be considered somewhat simplistic and lacks a solid foundation in mathematical theory. We intend to continue investigating this issue in our future work, seeking to uncover the root causes of the problems and design more robust methodologies.

**Author contributions** Qingyu Sun was primarily responsible for manuscript composition, Juan Zhang undertook content validation of the manuscript, while Zhijun Fang and Yongbin Gao provided guiding insights and advice on the paper.

## Declarations

**Conflict of interests** Qingyu Sun, Juan Zhang, Zhijun Fang, and Yongbin Gao declare no conflict of interest. This paper also did not plagiarize the research results of other authors.

**Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

## References

1. Allaouzi I, Ben Ahmed M, Benamrou B, Ouardouz M (2018) Automatic caption generation for medical images. In: Proceedings of the 3rd International Conference on Smart City Applications, pp 1–6
2. 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
3. Xiong Y, Du B, Yan P (2019) Reinforced transformer for medical image captioning. In Machine Learning in Medical Imaging: 10th International Workshop, MLMI 2019, Held in Conjunction with MICCAI 2019, Shenzhen, China, October 13, 2019, Proceedings 10, pp 673–680. Springer International Publishing. [https://doi.org/10.1007/978-3-030-32692-0\\_77](https://doi.org/10.1007/978-3-030-32692-0_77)



4. Ayesha H, Iqbal S, Tariq M, Abrar M, Sanaullah M, Abbas I, Hussain S et al (2021) Automatic medical image interpretation: State of the art and future directions. *Pattern Recogn* 114:107856
5. Yu J, Zhang J, Gao Y (2023) MACFNet: multi-attention complementary fusion network for image denoising. *Appl Intell* 53(13):16747–16761
6. Huang L, Wang W, Chen J, Wei XY (2019) Attention on attention for image captioning. In: *Proceedings of the IEEE/CVF international conference on computer vision*, pp 4634–4643
7. Anderson P, He X, Buehler C, Teney D, Johnson M, Gould S, Zhang L (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
8. Zhang Y, Zhang J, Huang B, Fang Z (2021) Single-image deraining via a recurrent memory unit network. *Knowl-Based Syst* 218:106832
9. Yang X, Tang K, Zhang H, Cai J (2019) Auto-encoding scene graphs for image captioning. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp 10685–10694
10. Aneja J, Deshpande A, Schwing AG (2018) Convolutional image captioning. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp 5561–5570
11. Papineni K, Roukos S, Ward T, Zhu WJ (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
12. ROUGE LC (2004). A package for automatic evaluation of summaries. In: *Proceedings of Workshop on Text Summarization of ACL, Spain*
13. Gu J, Wang G, Cai J, Chen T (2017) An empirical study of language cnn for image captioning. In: *Proceedings of the IEEE international conference on computer vision*, pp 1222–1231
14. Denkowski M, Lavie A (2014) Meteor universal: Language specific translation evaluation for any target language. In: *Proceedings of the ninth workshop on statistical machine translation*, pp 376–380
15. 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
16. Cho J, Yoon S, Kale A, Derroncourt F, Bui T, Bansal M (2022) Fine-grained Image Captioning with CLIP Reward. *Find Assoc Comput Linguistics: NAACL 2022*:517–527
17. Anderson P, Fernando B, Johnson M, Gould S (2016) Spice: Semantic propositional image caption evaluation. In: *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part V 14*, pp 382–398. Springer
18. Barraco M., Stefanini M., Cornia M., Cascianelli S, Baraldi L, Cucchiara R (2022, August) CaMEL: mean teacher learning for image captioning. In *2022 26th International Conference on Pattern Recognition (ICPR)*, pp 4087- 4094.
19. He S, Liao W, Tavakoli HR, Yang M, Rosenhahn B, Pugeault N (2020) Image captioning through image transformer. In: *Proceedings of the Asian conference on computer vision*
20. Huang X, Belongie S (2017) Arbitrary style transfer in real-time with adaptive instance normalization. In *Proceedings of the IEEE international conference on computer vision*, pp 1501–1510
21. Wang H, Wang H, Xu K (2020) Evolutionary recurrent neural network for image captioning. *Neurocomputing* 401:249–256. <https://doi.org/10.1016/j.neucom.2020.03.087>
22. 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
23. Kim Y, Soh JW, Park GY, Cho, NI (2020) Transfer learning from synthetic to real-noise denoising with adaptive instance normalization. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp 3482–3492
24. Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez A N, Polosukhin I et al (2017) Attention is all you need. *Advances in neural information processing systems*, 30.
25. Ling J, Xue H, Song L, Xie R., Gu X (2021) Region-aware adaptive instance normalization for image harmonization. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp 9361–9370
26. Vinyals O, Toshev A, Bengio S, Erhan D (2016) Show and tell: Lessons learned from the 2015 mscoco image captioning challenge. *IEEE Trans Pattern Anal Mach Intell* 39(4):652–663
27. Sharma H, Srivastava S (2022) A Framework for Image Captioning Based on Relation Network and Multilevel Attention Mechanism. *Neural Process Letters*. <https://doi.org/10.1007/s11063-022-11106-y>
28. Pan Y, Yao T, Li Y, Mei T (2020) X-linear attention networks for image captioning. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp 10971–10980
29. 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

30. Li G, Zhu L, Liu P, Yang Y (2019) Entangled transformer for image captioning. In: Proceedings of the IEEE/CVF international conference on computer vision, pp 8928–8937
31. Yao T, Pan Y, Li Y, Mei T (2019) Hierarchy parsing for image captioning. In: Proceedings of the IEEE/CVF international conference on computer vision, pp 2621–2629
32. Ribeiro AH, Tiels K, Aguirre LA, Schön T (2020) Beyond exploding and vanishing gradients: analysing RNN training using attractors and smoothness. In: International Conference on Artificial Intelligence and Statistics, pp 2370–2380. PMLR
33. Zhao H, Jia J, Koltun V (2020) Exploring self-attention for image recognition. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 10076–10085
34. Ye S, Han J, Liu N (2018) Attentive linear transformation for image captioning. *IEEE Trans Image Process* 27(11):5514–5524
35. Sarto S, Cornia M, Baraldi L, Cucchiara R (2022) Retrieval-augmented transformer for image captioning. In: Proceedings of the 19th International Conference on Content-based Multimedia Indexing, pp 1–7. <https://doi.org/10.1145/3549555.3549585>
36. Gao J, Wang S, Wang S, Ma S, Gao W (2019) Self-critical n-step training for image captioning. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 6300–6308
37. Mishra SK, Dhir R, Saha S, Bhattacharyya P, Singh AK (2021) Image captioning in Hindi language using transformer networks. *Computers & Electrical Engineering*, 92, 107114. 10.1016/j.compeleceng.2021.107114
38. Zhu H, Wang R, Zhang X (2021) Image Captioning with Dense Fusion Connection and Improved Stacked Attention Module. *Neural Process Lett* 53:1101–1118. <https://doi.org/10.1007/s11063-021-10431-y>
39. Ji J, Luo Y, Sun X, Chen F, Luo G, Wu Y, Gao Y, Ji R (2021) Improving Image Captioning by Leveraging Intra- and Inter-layer Global Representation in Transformer Network. *Proc AAAI Conf Artif Intell* 35(2):1655–1663. <https://doi.org/10.1609/aaai.v35i2.16258>
40. Tiwary T, Mahapatra RP (2023) An accurate generation of image captions for blind people using extended convolutional atom neural network. *Multimed Tools Appl* 82(3):3801–3830
41. Jiang W, Ma L, Jiang YG, Liu W, Zhang T (2018) Recurrent fusion network for image captioning. In: Proceedings of the European conference on computer vision (ECCV), pp 499–515.
42. Chen PD, Zhang J, Gao YB, Fang ZJ, Hwang JN (2024) A lightweight RGB superposition effect adjustment network for low-light image enhancement and denoising. *Eng Appl Artif Intell* 127:107234
43. Chen X, Fang H, Lin TY, Vedantam R, Gupta S, Dollár P, Zitnick CL (2015) Microsoft coco captions: Data collection and evaluation server. arXiv preprint [arXiv:1504.00325](https://arxiv.org/abs/1504.00325)
44. Lin TY, Maire M, Belongie SJ, Hays J, Perona P, Ramanan D et al (2014) Microsoft COCO: Common Objects in Context. In: Fleet D, Pajdla T, Schiele B, Tuytelaars T (eds) *Computer Vision – ECCV 2014*. ECCV 2014. Lecture Notes in Computer Science, vol 8693. Springer, Cham. [https://doi.org/10.1007/978-3-319-10602-1\\_48](https://doi.org/10.1007/978-3-319-10602-1_48)
45. 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
46. Sennrich R, Haddow B, Birch A (2015) Neural machine translation of rare words with subword units. arXiv preprint [arXiv:1508.07909](https://arxiv.org/abs/1508.07909)
47. 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.
48. 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
49. Zhang X, Sun X, Luo Y, Ji J, Zhou Y, Wu Y, Ji R (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.