symmetry

MDPI

Article

Full-Memory Transformer for Image Captioning

Tongwei Lu [1,2,*], Jiarong Wang [1,2] and Fen Min [1,2]

[1] School of Computer Science and Engineering, Wuhan Institute of Technology, Wuhan 430205, China
[2] Hubei Key Laboratory of Intelligent Robot Wuhan Institute of Technology, Wuhan 430205, China
* Correspondence: lutongwei@wit.edu.cn

Abstract: The Transformer-based approach represents the state-of-the-art in image captioning. However, existing studies have shown Transformer has a problem that irrelevant tokens with overlapping neighbors incorrectly attend to each other with relatively large attention scores. We believe that this limitation is due to the incompleteness of the Self-Attention Network (SAN) and Feed-Forward Network (FFN). To solve this problem, we present the Full-Memory Transformer method for image captioning. The method improves the performance of both image encoding and language decoding. In the image encoding step, we propose the Full-LN symmetric structure, which enables stable training and better model generalization performance by symmetrically embedding Layer Normalization on both sides of the SAN and FFN. In the language decoding step, we propose the Memory Attention Network (MAN), which extends the traditional attention mechanism to determine the correlation between attention results and input sequences, guiding the model to focus on the words that need to be attended to. Our method is evaluated on the MS COCO dataset and achieves good performance, improving the result in terms of BLEU-4 from 38.4 to 39.3.

Keywords: transformer; attention; image captioning; symmetric

Citation: Lu, T.; Wang, J.; Min, F. Full-Memory Transformer for Image Captioning. *Symmetry* **2023**, *15*, 190. https://doi.org/10.3390/sym15010190

Academic Editors: Dumitru Baleanu, Jeng-Shyang Pan and Jan Awrejcewicz

Received: 18 November 2022
Revised: 21 December 2022
Accepted: 31 December 2022
Published: 9 January 2023

1. Introduction

The image captioning task is designed to generate accurate natural language captions for images and includes the steps of acquiring an image, analyzing its visual content, and generating text captions to illustrate salient targets and behaviors in the image. Image captioning is a very complex task, facing three challenges. Firstly, the input and output are heterogeneous media forms, the network structure of encoding and decoding is also heterogeneous, and there is a cross-modal semantic gap, that is, there is polysemy and uncertainty in the representation of image content by unimodal image visual features. In addition, the information content of images is enormous, with a large amount of explicit and implicit visual semantic information, and it is necessary to identify the most important objects and scenes in a targeted manner, to establish accurate links between visual features and the generated text, and to describe them in a focused manner. In the end, the generated image captions must not only conform to the grammatical rules of natural language, that is to say, be formally well articulated, but also semantically coherent. In recent years, with the rapid development of computer vision and natural language processing [1,2], substantial progress has been made in image captioning tasks [3–5]. It is worth noting that an increasing number of image captioning models are using Transformer [6] as their network framework.

The Transformer is a deep neural network containing only the attention module and consists of two main structures: the Self-Attention Network (SAN) and the Feed-Forward Network (FFN). However, we find that Transformer is insufficient in its ability to capture local information. With the Self-Attention mechanism, we noted a finding that related words or adjacent words tend to receive larger attention scores, which is what we would like to see. However, we still found special cases. As shown in Figure 1, "a white dog catching for a frisbee". Although "catching" and "white" are two unrelated words, they

received high attention scores due to their presence as adjacent words before and after "dog". This high attention score for irrelevant words is not something we want to see. We need to direct the Transformer to focus on the words that need attention and reduce the attention scores of these irrelevant words.

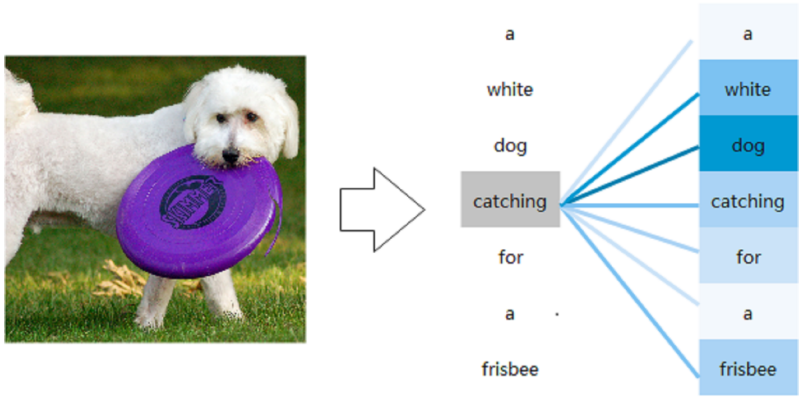


Figure 1. "white" and "catching" is the neighbors of "dog". The darker the color the greater the attention score.

To address this issue, we present a Full-Memory Transformer, which extends the encoder and decoder of a traditional Transformer. For the encoder, we propose a Full-LN structure, which extends the use of Layer Normalization in traditional Transformers. Especially, LN is widely used in two structures, known as Post-LN and Pre-LN. Post-LN was the structure used when the Transformer was proposed, but Post-LN suffers from the problem of slow convergence. Therefore, the Pre-LN structure was proposed to improve the convergence speed of the Transformer. In some tasks, the Pre-LN can achieve comparable performance to Post-LN but with only about 1/3 of the iteration speed. However, few researchers have verified which of the above two structures is better suited for image captioning tasks, or by a better structure. In this paper, we combine Pre-LN with Post-LN to design a more ideal Full-LN, adding LN above and below each sublayer. To prevent the network from becoming redundant, only one layer of LN is added between the two sub-layers. For the decoder, we propose the Memory Attention Network (MAN), which extends the traditional attention mechanism to determine the correlation between attention results and input sequences, guiding the model to focus on the words that need to be attended to. Specifically, its key vector and value vector are served by the attention results, and its query vector is maintained by the memory vector of the input sequence, called the memory block. The memory block is the weight of the current input and the previous.

Full-LN and MAN are applied to image coding and language decoding respectively. For the encoder, Full-LN is applied to achieve better generalization performance and stable training. For the decoder, it applies MAN to filter out irrelevant/misleading attention results and retain only useful attention results. We evaluate the impact of applying Full-LN and MAN to the encoder and decoder respectively. The results show that Full-LN and MAN modules are valid. Our proposed Full-memory Transformer achieves 39.3 BLEU [7] and 22.3 SPICE [8] scores in offline test segmentation of MS COCO datasets. The main contributions of this paper are as follows:

(1) We introduce the Full-LN structure and apply it to the encoder structure of the standard Transformer. With this approach, our model reduces the loss in image feature transfer and generates more accurate captions.
(2) We use the memory block to maintain information about the input sequence in the decoding phase and as a query vector for the standard self-attentive mechanism while using the model's attention results as keys and values to form the MAN module. In this way, our MAN model can be improved to some extent in the decoding stage, so

that the model focuses on the words that need to be attended to improve the accuracy of the generated subtitles.

(3) We validated our method on the widely used MS COCO dataset and our model outperformed other methods on BLEU-2, BLEU-3, METEOR, ROUGE, and SPICE under Cross-Entropy Loss. Under CIDEr Score Optimization, our model outperforms other methods on BLEU-2, BLEU-3, and BLEU-4 metrics.

2. Ralated Works

2.1. Image Captioning

Early methods of image captioning consisted of two main categories. One type of approach is the template-based image captioning approach [9,10], which first captures objects, actions, and scenes in an image and then fills them into a fixed sentence template to generate an annotation of the image. This type of method is simple to use, but the generated sentences lack a certain variety and differ too much from the presentation style of the manually annotated sentences. The other is a retrieval-based approach [11,12], which ranks the similarity of the target image to images in a database of manually annotated images and retrieves the best representation. However, the text generated by this method relies heavily on the manual annotations in the image database and is unable to generate novel statements, which lacks flexibility. With the success of deep learning in recent years, the encoder-decoder-based model [13–18] has solved the constraint that the input and output sequences must be of equal length, and it has been used extensively in tasks such as machine translation and machine writing in the field of natural language, with very good results. Researchers have applied it to image captioning tasks, using an encoder to extract features from the input image and then decoding it to generate a final text description [19]. In terms of training, although the original approach was based on Cross-Entropy training in time, significant achievements have been made with the introduction of reinforcement learning [5,20–22]. For image coding, grid features from CNNs were initially used [4,23,24] and currently image region features extracted by object detectors are used [25,26]. To further improve the coding of objects and their relations, SGAE [27] proposed the use of graph convolutional neural networks to integrate semantics in the image coding phase. In addition, some image captioning models [28–31] investigate the diversity and controllability of caption generation.

2.2. Transformer

Lately, a deep neural network containing only attention has been proposed with state-of-the-art results in natural language understanding and computer vision. Similarly, several recent approaches have investigated the application of the Transformer model [6] to image captioning with good results. The Transformer is a deep neural network containing only the attention module and consists of two main structures, the SAN and the FFN.

SAN is a type of attention mechanism and an important part of the Transformer. The purpose of the attention mechanism is to focus on part of the details based on our goal, rather than analyzing based on the big picture. So the core of the attention mechanism is how to determine the part we want to focus on based on the goal, and how to analyze it further after finding this part of the detail. For the image captioning task, SAN aims to obtain the degree of correlation between the current word and other words, and SAN does this by calculating the similarity of word vectors. In general, the closer the meanings of two-word vectors are, the smaller the distance angle between them and the larger the product. The weights are obtained by normalizing the similarity. Multiplying the weights with the word vectors and finally adding them together gives the attention score of the current word to the other words, i.e., the attention level.

The FFN is a one-way propagation neural network that can be grouped according to the order in which information is received. The neurons in each layer receive the output of the neurons in the previous layer and output to the neurons in the next layer. The FFN can be thought of as a function that achieves a complex mapping from input space to output

space through the multiple compounding of simple non-linear functions. Unlike a SAN, it is only concerned with the signal being input from one end and output from the other.

Layer Normalization (LN) is one of the components of a Transformer. In stochastic optimization theory, the learning rate is often set to a constant or decayed to ensure convergence of the algorithm. However, neither setting the learning rate to a constant nor decaying the learning rate allows the Transformer to converge well. The optimization of the Transformer structure is very difficult in two aspects: (1) the learning rate is hyperparameter-sensitive in the warm-up phase. (2) Slow convergence of the optimization process. To address these problems, Xiong [32] et al. proposed two Layer Normalization structures and compared them. The traditional Add and Norm in the Transformer architecture is called Post-LN, and for Post-LN, another structure, Pre-LN, is proposed, where Layer Normalization is added before the residual join. They conclude that the warm-up phase is no longer necessary when using the Pre-LN structure and that the Pre-LN structure can substantially improve the convergence speed of the Transformer. In machine translation tasks, the Pre-LN structure without a warm-up can converge about a factor of 1 faster than Post-LN, while on BERT, Pre-LN achieves the same performance as Post-LN on downstream tasks at about 1/3 of the latter's iteration speed, and with better final results.The Post-LN is adopted by various state-of-the-art models including BERT [33], XLNet [34], Roberta [35], ALBERT [36], Transformer-XL [37], and ETC [38]. Another design is Pre-LN which is used by some well-known extra-large models such as GPT-2 [39] and Megatron [40].

More and more researchers are applying the Transformer to the field of image captioning. Herdade et al. [41] proposed an object relationship transformer to incorporate spatial relationship information into image features extracted by target detection by geometrically focusing on such methods, thus improving the accuracy of image captioning. Osolo et al. [42] proposed a Transformer-based model that reduces losses in image feature transfer by using the Fast Fourier Transform to further decompose input features and extract more intrinsically salient information, resulting in more detailed and accurate captions. Jia et al. [43] argue that it is difficult for SAN to capture the semantic association between candidate objects and queryable objects when the appearance features between them are not obvious. Therefore, they propose a semantic enhancement module, constructed from the geometric and appearance features of objects, which can enhance the semantic association of objects with weak appearance features. While the above methods make different improvements to the Transformer and also improve the accuracy of the generated caption, in our work we propose the Full-Memory Transformer, which improves the model performance from the encoder and decoder respectively. For the encoder, we design a novel structure Full-LN based on Pre-LN and Post-LN, which not only achieves the same loss reduction during image transfer as AFCT [42], but is also simpler to implement, requiring only a few lines of code. For the decoder, we propose MAN, an extension of the standard Self-Attentive mechanism. The model is guided to focus on the words that should be attended to by maintaining the state and attention results of the current sequence. As a result, the captions generated by our model tend to be more detail-oriented and more accurate.

3. Materials and Methods

3.1. Model Architecture

The architecture of our approach is the Encoder-Decoder Architecture, which was first applied to machine translation and other sequence conversion tasks. The Encoder-Decoder Architecture has more advantages than traditional image captioning architectures such as the Injecting Architecture and the Merging Architecture. The Encoder-Decoder Architecture can use variable length sequences as input and output and is therefore also suitable for image captioning tasks. As shown in Figure 2, our architecture uses a Transformer encoder to encode the visual features and a Transformer decoder to generate the captions. Our model consists of an encoder for feature extraction and a decoder for user text generation. The encoder is responsible for extracting the key features in the image from the input image

and, to some extent, processing their relationship with each other, and then passing the extracted features to the decoder. After this, the decoder reads the information from the output of each encoding layer and uses the obtained feature information to progressively generate words and eventually a sentence.

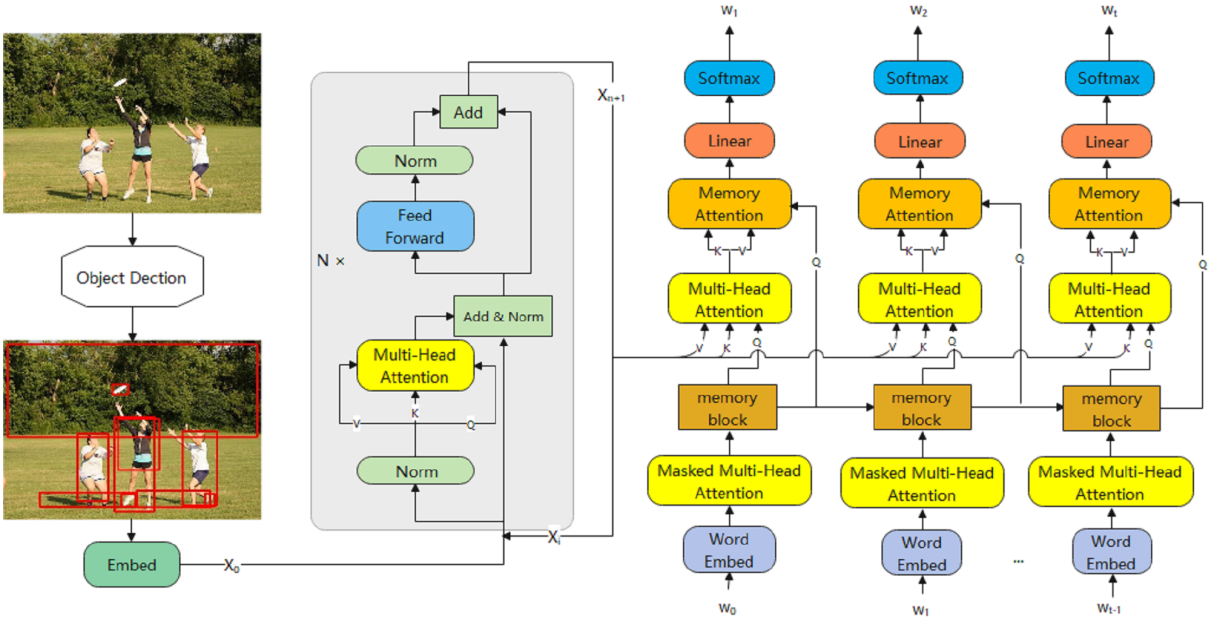


Figure 2. The framework of our proposed method.

Both the encoder and the decoder consist of two main structures, the SAN and the FFN. The purpose of the SAN module is to help the current node focus not only on the current word itself but also on the surrounding words to get the context going better. The Formula (1) is as follows:

$$Attention(Q, K, V) = Softmax(\frac{QK^{\top}}{\sqrt{d_k}})V. \tag{1}$$

where Q, K, and V are obtained from the input sequence by matrix transformation. Where Q, K is used to calculate the similarity of different words, V can be considered as the feature representation of a word, multiplied and summed with the attention score obtained from Q, K, and then normalized by the softmax function to obtain the final result.

Much simpler than a SAN, the FFN module consists of linear layers stacked on top of each other, unable to perceive information other than itself and only able to focus on itself. As shown in Formula (2):

$$FNN(X) = \sigma(Xw_1 + b_1)w_2 + b_2 \tag{2}$$

where σ is an activation function, in which case RELU is usually used.

3.2. Problem Description

As demonstrated in our architecture, the Transformer takes on a very central role in our model. However, in the image captioning task, there are still areas where the Transformer needs to be improved. We have noticed a problem where unrelated and non-adjacent words may receive larger attention scores. We know that the purpose of the self-attention mechanism is to identify the part of the target we want to focus on based on the goal and to further analyze this part of the detail once it has been found. Once irrelevant words receive a larger attention score, then our model suffers a performance loss in two ways. (1) The attention score sums to one, and a larger score for errors affects the model's ability

to focus on the words that really need attention. (2) Too much attention to unimportant parts is unnecessary. Therefore, in the following sections, we will describe how our model improves on this problem.

3.3. Features Selection

As shown in Figure 3, we compare two different methods of image feature extraction. One is a CNN-based feature extraction method, which extracts regular rectangular boxes and assigns the same weight to different regions. One is based on the target detection algorithm (Faster-RCNN) extracting visual region features, which extracts irregular rectangular boxes. The features extracted using the Object Detection algorithm are more compatible with human visual features than the regular matrix boxes extracted by the standard CNN. In our work, we use the target-based detection algorithm (Faster-RCNN) to extract the regional features of an image in order to obtain a better representation of the image features.

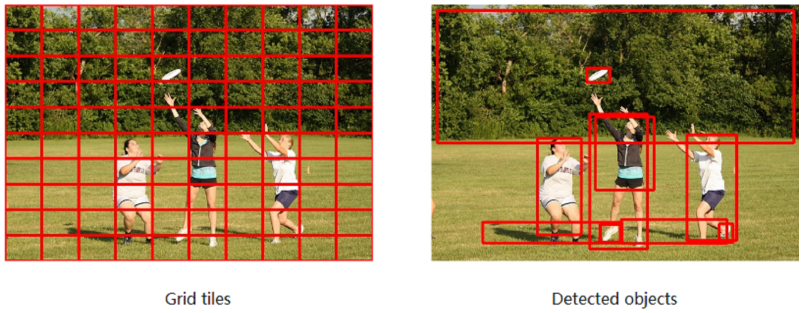


Figure 3. Two feature extraction methods.

3.4. Image Encoding

Existing Transformer networks have two canonical designs that differ only in the way they organize their modules. They are Post-LN and Pre-LN respectively. Post-LN was the structure used when the Transformer was proposed, but Post-LN suffers from the problem of slow convergence. Therefore, the Pre-LN structure was proposed to improve the convergence speed of the Transformer, and in some downstream tasks the Pre-LN can achieve comparable performance to Post-LN but with only about 1/3 of the iteration speed. However, few researchers have verified which of the above two structures is better suited, or better structure, for image captioning tasks. In this work, we combine two schemes, as shown in Figure 4. We regard SAN and FFN as two fragile items, to fully use them, a layer of Layer Normalization should be added to the upper and lower layers of each structure. The two structures are adjacent, so only one layer of Layer Normalization is needed in the middle.

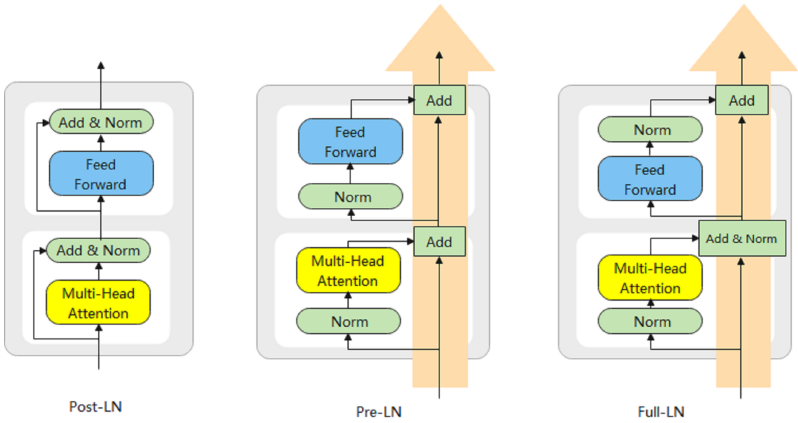


Figure 4. The three LN architectures.

The task of the image encoder is to convert the image region features into the features available to the decoder. Specifically, for the image I, we first extract a set of regional feature vectors $V = v_1, v_2, \ldots, v_k$. An image encoder is a module that converts the input set of spatial image region feature V into a series of intermediate states and enhances them through the context information between the intermediate states. Given a set of image region feature V extracted from the input image. we make the Layer Normalization of image region feature V according to the structure of Full-LN. As shown in Formula (3):

$$X = LayerNorm(V) \tag{3}$$

The permutation invariant encoding of X can then be obtained through the Self-Attention operation used in Transformer [6]. In this case, Q, K, and V are obtained by linear projection input characteristics. As shown in Formula (4):

$$S = Attention(W_Q X, W_K X, W_V X) \tag{4}$$

where,W_Q, W_K, and W_V are matrices of learnable weights.

As we mentioned above, we have embedded Layer Normalization before and after the SAN and *FFN* respectively.

$$Z = LayerNorm(X + S) \tag{5}$$

$$F = FFN(Z) \tag{6}$$

$$O = LayerNorm(Z + F) \tag{7}$$

Finally, we will add the LN symmetrically on both sides of the *FFN*. Equation (5) represents a residual join of the input picture features and the self-attentive results, which are then normalized. Equation (6) represents the propagation of the current results through the feed-forward neural network. As mentioned above, Equation (7) is part of the Full-LN structure and exists symmetrically with Equation (5), again using residual concatenation followed by normalization.

3.5. Language Decoding

The purpose of the SAN module is to help the current node focus not only on the current word itself but also on the surrounding words to get the context going better. Much simpler than a SAN, the FFN module consists of linear layers stacked on top of each other, unable to perceive information other than itself and only able to focus on itself. However, the module outputs a weighted average for each query, regardless of whether or how Q and K/V are related. Even if there is no associated vector, take note that the module still generates a weighted average vector, which may contain information that is irrelevant or even deceptive. Specifically, irrelevant vectors with overlapping neighbors are more likely to get higher attention scores, thus influencing the word that needs real attention. Therefore, we maintain a memory block concerning the input sequence. As shown in Formula (8):

$$M = Y_t + \theta * Y_{t-1} \tag{8}$$

where M stands for the memory block, Y_t is the input of the current, and Y_{t-1} is the input of the previous. λ is the default parameter.

On this basis, we implement the Memory Attention Network, which follows the standard Transformer Self-Attention mechanism. Its query vector is acted by memory

block, and its key vector and value vector are acted by attention results. As shown in Formula (9):

$$MAN = Attention(W_Q M, W_K X, W_V X) \tag{9}$$

where, W_Q, W_K, and W_V are matrices of learnable weights. X is the output of the decoder Cross-Attention Network, and M is the result of the memory block.

As shown in Figure 5. Our decoder generates the predicted words in the current stage based on the words generated in the previous stage and the visual features enhanced by the encoder. Specifically, after location embedding, the words generated in the previous stage will be input to the masked Multi-head Attention module as an input feature. Different from multiple attention in the coding stage, the cover mechanism is used here mainly because there is future information in the ground truth, which should not exist in the actual generation of text description statements. Therefore, the cover mechanism is used to ensure the consistency of the training and testing process. The specific approach is to use a matrix with the upper triangle as 1 and the other positions as 0 for the mask in the calculation of attention weight. Further, following the standard implementation of Transformer decoders, the encoder's output is transformed into a set of attention vectors K and V, while the output of mask Mutil-head Attention is taken as Q. Self-Attention operations are performed between them. This Self-Attention layer is usually called the encoder-decoder Attention layer. The goal is to make the decoder reference K, and V information more focused on the relevant parts of the input sequence. Then, the function of the MAN we proposed here is to make the attention result pay more attention to the part of the input sequence, and avoid overlapping irrelevant vectors of neighbors, to obtain higher attention scores. Following the practice of a standard Transformer, the result of the MAN will be connected to the Feed-Forward Network using Layer Normalization and residual error. Finally, the softmax will generate the word of the current time step according to the probability.

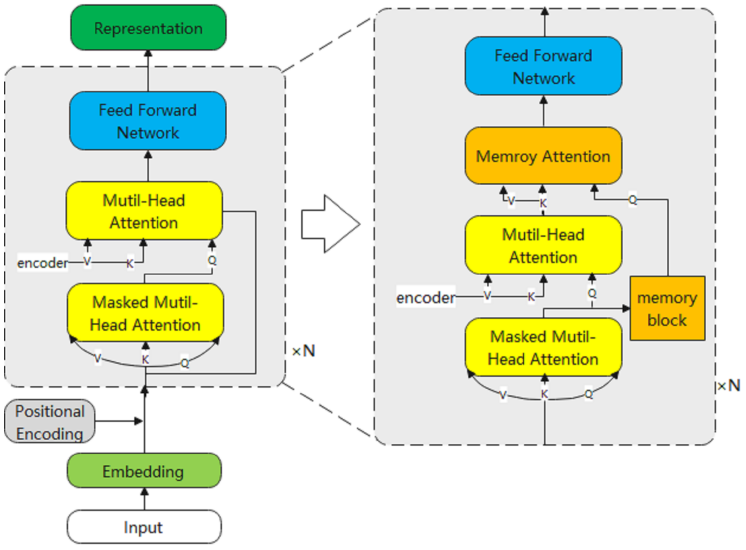


Figure 5. On the left is the standard decoder, and on the right is the decoder with MAN added.

3.6. Training

As most image description models do, the model is first trained using Cross-Entropy Loss (CE) and then trained using Self-Critical (SCST) [29] methods. The minimum Cross-Entropy Loss is (see Formula (10)):

$$L_{CE}(\theta) = -\sum_{t=1}^{T} \log(P_\theta(y_t^* | y_1^*, \ldots, y_t^*)) \tag{10}$$

After using the Cross-Entropy Loss training model, the reinforcement learning training model is carried out according to SCST. We consider the model as the 'agent' and the text and images as the 'environment'. The "agent" policy is the parameter of the encoder and decoder in this paper. When the image captioning is generated, the generated sequence is regarded as the "state" of the current time when the next word unit is generated. Each time the agent takes an action to generate a new word, it will return the CIDEr-D score reward according to the state and action taken. The goal of using the SCST training model is to minimize negative expectations. The corresponding formula is (see Formula (11)):

$$L_{SCST} = -E_{Y_S - P_\theta}[r(Y_s)] \tag{11}$$

where, $Y_s = (y_{1,s}, y_{2,s}, \ldots, y_{t,s}, y_{T,s}), y_{t,s}$ is the word sampled from the model at time t and is CIDEr-D fraction function. The loss expressed by Formula (12) can be further approximated as:

$$\nabla_\theta L_{SCST} \approx -(r(Y_s) - r(\overline{Y}))\nabla_\theta \log P_\theta(Y_s) \tag{12}$$

where, $Y_s = (y_{1,s}, y_{2,s}, \ldots, y_{T,s})$ is obtained by sampling according to P_θ using Monte-Carlo idea, $\overline{Y} = (\overline{y_1}, \overline{y_2}, \ldots, \overline{y_T})$ is based on the greedy decoding of the model.

4. Results and Discussion

4.1. Datasets

We have validated the effectiveness of our method on the popular MS COCO dataset. The entire MS COCO dataset contains 123,287 images, including 82,783 training images, 40,504 validation images, and 40,775 testing images. Each image corresponds to five human–annotated sentences. In this paper, we have mainly used the well-known "karpathy" partitioning method to validate our model. In the case of the "karpathy" segmentation method, we will have 113,287 images for training, 5000 images for validation, and 5000 images for testing. In addition, we additionally perform pre-processing operations on the sentences involved in the training, firstly by converting all the upper case letters in their sentences to lower case letters, and then by deleting the words that occur less than 5 times, resulting in a vocabulary of 10,369 words.

4.2. Hardware Description

Our experimental equipment is as follows: CPU is Intel Core i9-9900K, released in 2018, CPU frequency is 3.1 GHz, 8 cores and 16 threads, maximum supported memory is 128 GB, memory type is DDR4 2666 MHz, maximum memory bandwidth is 41.6 GB/s. GPU is 2 × 2080 graphics cards, released in 2018, core frequency is 1515/1800 MHz, memory type GDDR6, memory frequency 14,000 MHz, memory bit width 256 bit, and 2944 CUDA cores. The operating system is Ubuntu 16.04 LTS under Linux, the deep learning framework is Pytorch 1.5, python version 3.6, and CUDA version 10.1.

4.3. Experimental Settings and Training Details

We report results using the MS COCO [44] captioning evaluation toolkit [44] that reports the widely-used automatic evaluation metrics SPICE [8], CIDEr [45], BLEU [7], METEOR [46], and ROUGE [47]. BLEU [7] was originally an algorithm used to evaluate the quality of texts obtained by a machine translation from one natural language to another. This is in line with the requirement for image description algorithms to evaluate the generated text, that is, to score the differences between the generated utterance to be evaluated and the manually annotated utterance, with a score output between 0 and 1. This criterion has now become one of the most widely used computational criteria for image description algorithms. ROUGE [47] was originally used as an evaluation criterion for assessing automatic summarization and machine translation in natural language processing. It involves multiple experts in the field of natural language processing giving a professional description of the specified data and then comparing the automatically generated summary or translation with it. The quality of automatic summaries or translations is evaluated by

comparing the number of overlaps between the two such as n-grammar, word sequences, and word pairs. The higher the ROUGE score, the better the performance. METEOR [46] was also originally used as a criterion for evaluating machine-translation output. The algorithm is based on a harmonic mean of accuracy and recall across the corpus. In short, it compares the overlap of monads between the utterance to be evaluated and the reference utterance and matches the monads according to semantics, stem form, and precision. Compared to the BLEU standard, this standard is closer to human judgment when evaluating because it introduces external knowledge. Unlike the above criteria, CIDEr [45] is specifically designed to evaluate image description algorithms by calculating the TF-IDF weights of each n-tuple to obtain the similarity between the statement to be evaluated and the reference statement, thereby evaluating the effectiveness of the image description.

4.4. Assumptions

For better performance, following standard practice, During the training, we trained Full-Memory Transformer to train 30 stages under Cross-Entropy Loss, with a small batch of 10, using ADAM [23] optimizer, and the learning rate was initialized to 3×10^{-4}. We increased the planned sampling probability by 0.05 for every 5 epochs [25]. We used SCST to optimize CIDEr-D scores for another 24 periods, with an initial learning rate of 5×10^{-6}, dropout of 0.3, and beam size equal to 5.

4.5. Performance Comparison

In Table 1, we compare the performance of our method with several recent image captioning methods.

Table 1. Performance of our model and other methods on MS COCO datasets, where B@N, M, R, C, and S are short for BLEU@N, METEOR, ROUGE-L, CIDEr-D, and SPICE scores.

-	Cross-Entropy Loss								CIDEr Score Optimization							
Matrix	**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	-	-	-	29.6	25.2	52.6	94.0	-	-	-	-	31.9	25.5	54.3	106.3	-
SCST	-	-	-	30.0	25.9	53.4	99.4	-	-	-	-	34.2	26.7	55.7	114.0	-
LSTM-A	75.4	-	-	35.2	26.9	55.8	108.8	20.0	78.6	-	-	35.5	27.3	56.8	118.3	20.8
RFNET	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
Up-Down	77.2	-	-	36.2	27.0	56.4	113.5	20.3	79.8	-	-	36.3	27.7	56.9	120.1	21.4
GCN-LSTM	77.3	-	-	36.8	27.9	57.0	116.3	20.9	80.5	-	-	38.2	28.5	58.3	127.6	22.0
SGAE	**77.6**	-	-	**36.9**	27.7	57.2	**116.7**	20.9	**80.8**	-	-	38.4	28.4	58.6	127.8	22.1
ORT	-	-	-	-	-	-	-	-	80.5	-	-	38.6	28.7	58.4	128.3	**22.6**
AFCT	-	-	-	-	-	-	-	-	80.5	-	-	38.7	**29.2**	58.4	**130.1**	22.5
SAET	-	-	-	-	-	-	-	-	-	-	-	39.1	28.9	**58.7**	129.6	**22.6**
OURS	76.6	**60.5**	**46.8**	36.3	**28.3**	**57.6**	116.6	**21.3**	80.6	**64.9**	**50.9**	**39.3**	28.4	58.5	126.5	22.3

These models, including the LSTM, were designed to address the long-term dependency problem that exists in RNNs and has achieved relatively good results in areas such as sequence modeling and machine translation. We then compared SCST [5], which uses reinforcement learning to address the exposure bias and non-differentiability problems of existing evaluation metrics. Comparing this approach, our method performs better on all evaluation metrics. In addition, LSTM-A is an extension of LSTM which improves LSTM by emphasizing semantic properties in the decoding phase. We compare this with the UpDown [3] approach, which proposes two different attention mechanisms, a bottom-up attention mechanism using Faster-RCNN [44] to extract regions of interest in the image and a top-down attention mechanism that assigns different word weights in the decoding

stage. In addition, we compared it with RFNet [33], which uses a recursive fusion network to merge different CNN features. Our method outperforms it on different evaluation metrics. We also compared GCN-LSTM, which exploits pairwise relationships between image regions via a graphical CNN. It can be seen that it performs better on BLEU-4, while on other evaluation metrics our approach is better. SGAE [27], which instead uses autoencoding scene graphs. ORT [41], which incorporates spatial relationship information into the image features extracted by target detection by geometrically focusing on this method, thus improving the accuracy of image captioning. By comparison, our method outperforms it in BLEU–, BLEU–, and ROUGE metrics. AFCT [42] produces more detailed and accurate captions by using the Fast Fourier Transform to further decompose the input features and extract more intrinsically salient information, reducing the loss in image feature transfer. Our method outperforms it in the metrics BLEU-1, BLEU-4, and ROUGE. It scored 29.2 in METEOR, higher than other models. SAET [43] which proposes a semantic enhancement module, constructed from geometric and appearance features of objects, capable of enhancing the semantic association of objects with weak appearance features. Our approach outperforms it in terms of BLEU-4 metrics. In conclusion, our model achieves the best performance on BLEU-2, BLEU-3, METEOR, ROUGE, and SPICE for the results of Cross-Entropy Loss training. For the results optimized by CIDEr scores, our model achieves the best performance on BLEU-2, BLEU-3, and BLEU-4.

4.6. Qualitative Analysis

Figure 6 shows some examples, including images and captioning generated by our Full-Memory Transformer, the original Transformer as the baseline model, and three tags generated by humans.

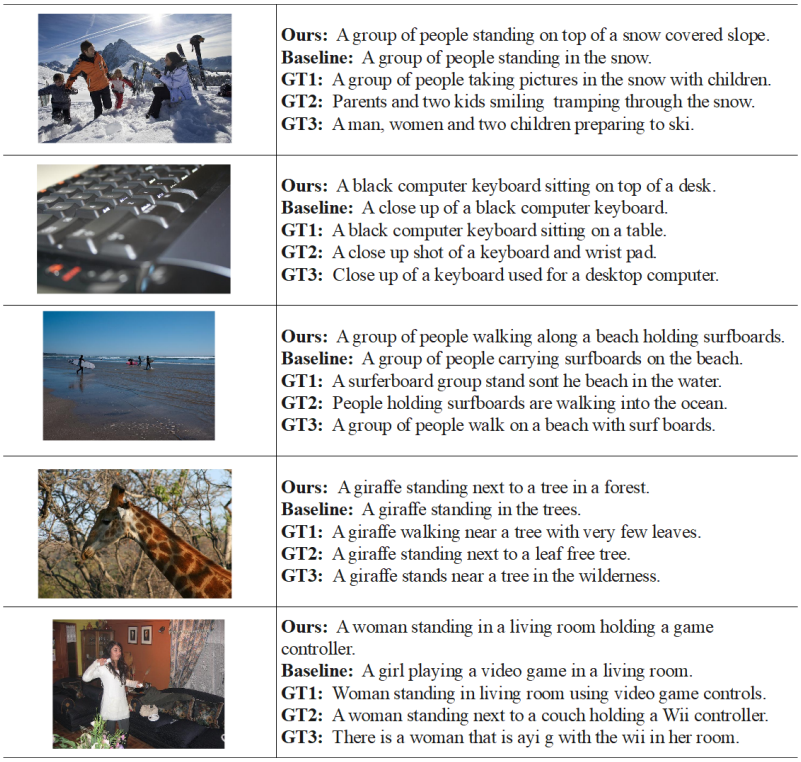


Figure 6. Examples of captions generated by Full-Memory Transformer and a baseline model as well as the corresponding ground truths.

From these examples, we found that the headings generated by the baseline model, while being linguistically logical, failed to discover the connections and details of the content in the image. In contrast, our model generates higher-quality titles. Especially, our model generates titles that are usually more specific and detail-oriented. In the first example, the baseline model can only get information about a group of people standing on a snowy mountain, whereas our model takes into account the fact that people are standing on a snowy hillside, and our model captures even better detail than the actual human labels. In the second example, the baseline model only found a black keyboard, while our model took into account the more specific information that the black keyboard was on the table. In the third example, the baseline model found the giraffe next to the tree, whereas our model took into account the overall environment. More specifically, the tree was in the forest. As the generated subtitles show, our model captures details better than the traditional Transformer. Refer to Figures A1 and A2 in Appendix A for more information.

As shown in Figure 7, we show a plot of our results for 30 epochs using the Cross-Entropy Loss function. As the training losses for the first 20 epochs show, the overall loss drop is significant, with a total drop of about 3.2. This means that at this stage our model is still learning. In the last 10 epochs, we note that the overall drop in loss is about 0.1. This indicates that the distribution learned by our model has reached a bottleneck under Cross-Entropy Loss.

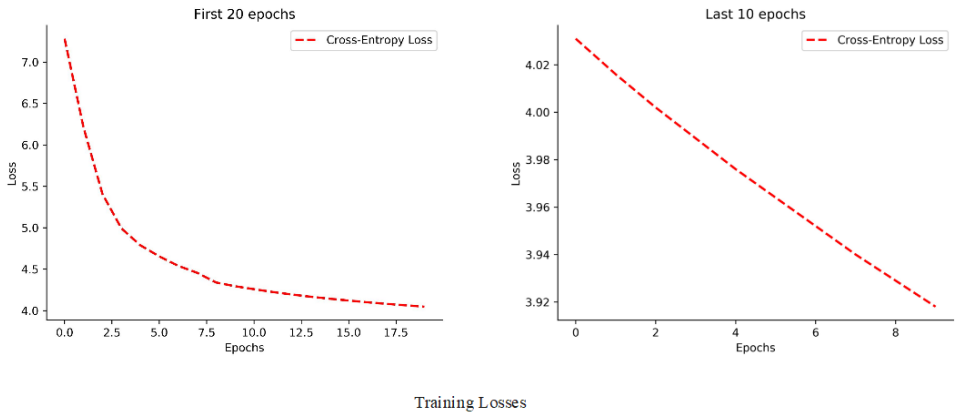


Figure 7. Cross-Entropy Loss for the first 20 epochs and the last 10 epochs.

4.7. Ablation Analysis

To begin with, we compared the performance impact of different layers of encoders and decoders in the Transformer architecture. The standard Transformer uses a six-layer encoder and a six-layer decoder. We note that the sentences in the image captioning task are simpler compared to other language comprehension tasks, so we reduced the number of encoder and decoder layers from 6 to 3 respectively to see the results. As Table 2 shows, we can see that the standard Transformer structure with six layers only achieves a CIDEr score of 110.9, which is significantly lower than the other three cases and does not give particularly good results in other metrics. However, the three-layer Encoder and Decoder structures obtained good results on different metrics, and based on this finding, subsequent experiments were conducted in the three-layer Encoder Decoder state. Next, we verified the effectiveness of the encoder and decoder. Full-LN is an encoder component of our model, which extends the use of Layer Normalization. MAN is a component of our model decoder. It enhances the correlation between attention results and input sequences by extending the traditional attention mechanism. At last, we verify the results of adding both Full-LN encoder and MAN decoder models.

Table 2. Comparison based on the number of Transformer layers.

-	Cross-Entropy Loss							
Matrix	**B@1**	**B@2**	**B@3**	**B@4**	**M**	**R**	**C**	**S**
Transformer 6	73.9	57.6	44.3	34.2	27.7	56.5	110.9	20.4
Transformer 3	**74.4**	**57.9**	**44.7**	**34.6**	**27.8**	**56.6**	**112.0**	**20.6**
Transformer 3,6	74.2	57.8	44.6	**34.6**	**27.8**	**56.6**	111.8	**20.6**
Transformer 6,6	74.1	57.7	44.6	**34.6**	27.7	**56.6**	111.4	20.5

Effect of Full-LN encoder. To evaluate the effect of Full-LN applied to encoders, we compared the original architecture Post-LN and Pre-LN. As shown in Table 3, Full-LN brings positive effects. Compared with the original Post-LN, the encoder structure of Full-LN has improved significantly in all indexes, especially in CIDEr [45] index by 3.4.

Table 3. Comparison based on Transformer encoder.

Encoder	Cross-Entropy Loss							
Matrix	**B@1**	**B@2**	**B@3**	**B@4**	**M**	**R**	**C**	**S**
Post-LN	74.4	57.9	44.7	34.6	27.8	56.6	112.0	20.6
Pre-LN	74.8	58.5	45.0	34.7	28.1	56.9	113.7	**21.6**
Full-LN	**75.3**	**59.0**	**45.6**	**35.3**	**28.3**	**57.3**	**115.4**	21.4

Effect of the decoder with MAN. Following previous practice, we choose to use the simple non-linear fusion mechanism. From Table 4, to better apply MAN to the decoder, we compare the effect of the model with MAN when $\lambda = 0.0$, $\lambda = 0.1$, and $\lambda = 0.3$. We draw two conclusions. First, whether $\lambda = 0.0$, $\lambda = 0.1$, or $\lambda = 0.3$, our model has a better performance compared to the base model. Then, the experimental results surface, when $\lambda = 0.1$, and our MAN performs better in the decoder. In the end, the encoder based on Full-LN and the decoder based on MAN are integrated. Compared with the base model, the effect of our model is significantly improved.

Table 4. Comparison based on Transformer decoder.

Decoder	Cross-Entropy Loss							
Matrix	**B@1**	**B@2**	**B@3**	**B@4**	**M**	**R**	**C**	**S**
$\lambda = 0.0$	75.0	58.6	45.2	35.1	27.7	57.0	114.7	21.0
$\lambda = 0.1$	75.2	58.8	45.3	34.8	27.9	57.2	114.2	21.2
$\lambda = 0.3$	74.9	58.1	45.1	34.9	27.6	56.8	114.0	20.8
Full-LN + MAN	**76.6**	**60.5**	**46.8**	**36.3**	**28.3**	**57.6**	**116.6**	**21.3**

We examine our approach through the generated image captions. The methods we compare include Post-LN, Pre-LN, Full-LN, and Full-LN + MAN. Post-LN is the structure of the traditional Transformer encoder. Pre-LN improves on the difficulties of optimizing Post-LN structures by increasing the speed of model convergence without loss of performance. Full-LN is our proposed structure combining Post-LN and Pre-LN to enable the model to generate more accurate subtitles. Full-LN + MAN means using Full-LN in the encoder and adding MAN to the decoder. MAN can better guide the model to focus on the words that really need attention, making the generated captions more accurate and detailed. As shown in Figure 8. In (a) we find this by comparison. After using the Full-LN structure, our model notices the detail that the field is a baseball field. With the inclusion of MAN, our model accurately identifies the "baseball players" rather than the other three captions with a vague representation of "people". In (b), our model, after using the Full-LN encoder and MAN decoder, notices that the food is cake and the bowl is green. In (c), our Full-LN

encoder and MAN decoder identify that human behavior is through a grocery shop. In (d), our Full-LN encoder and MAN decoder notice that people are not only standing but looking at their phones. In summary, we can find that our Full-LN structure can obtain a part of the detailed features of the image. In particular, with the addition of MAN, our model is better in terms of detail capture and caption accuracy.

(a)	(b)	(c)	(d)
Post-LN: a group of people standing on a field.	**Post-LN:** a bowl of food on a counter.	**Post-LN:** a couple of people standing in front of a store.	**Post-LN:** a group of people standing next to each other.
Pre-LN: a group of people standing on top of a field.	**Pre-LN:** a couple of a bowl of food on a counter.	**Pre-LN:** a couple of people standing in front of a store.	**Pre-LN:** a group of people standing next to each other.
Full-LN: a group of men standing on top of a baseball field.	**Full-LN:** a couple of a bowl of food on a counter.	**Full-LN:** a couple of people standing in front of a store.	**Full-LN:** a group of people standing next to each other.
Full-LN + MAN: a couple of baseball players standing on top of a field.	**Full-LN + MAN:** a cake sitting on top of a green plate next to a bowl of fruit.	**Full-LN + MAN:** a group of people walking through a grocery store.	**Full-LN + MAN:** a group of people standing around looking at their phones.

Figure 8. Image captioning results for our method and related methods on the MS COCO dataset.

5. Conclusions

In this paper, we present the Full-Memory Transformer, which includes an encoder with Full-LN and a decoder with MAN. Our encoder extends the use of Layer Normalization and provides better generalization performance by comparing it with existing LN structures. Our decoder is based on traditional architecture by adding a layer of self-attention mechanism to guide the model to focus on the words that need attention and thus generate better sentences. The experimental results indicate that our method achieves excellent performance on the MS COCO dataset. We validate the components of our model through ablation studies, and our model generates sentences that pay more attention to detail than the baseline model. In addition, our future work can be further investigated in the following aspects. Firstly, we will make further improvements to the existing network. Current Transformer-based image captioning models are generally associated with a complex network structure and a large number of parameters. Our model is no exception, and we plan to modify the model in the future so that its performance is no less than that of the existing model with a small number of parameters. Secondly, more and more short videos are entering people's lives, and we will try to study video captioning tasks in the future, where the understanding of videos faces more challenges than pictures. Finally, we plan to do further research and exploration of our approach in industrial scenarios.

Author Contributions: Conceptualization: T.L. and J.W.; methodology: T.L. and J.W.; software: J.W.; validation: J.W.; formal analysis: J.W.; investigation: T.L. and J.W.; resources: T.L.; data curation: T.L. and F.M.; writing—original draft preparation: J.W.; writing—review and editing: J.W., T.L. and F.M.; visualization: J.W.; supervision: J.W., T.L. and F.M.; project administration: T.L.; funding acquisition: T.L. All authors have read and agreed to the published version of the manuscript.

Funding: This work is supported by the Hubei Technology Innovation Project (2019AAA045), Graduate Innovative Fund of Wuhan Institute of Technology (No.CX2021244), and the National Natural Science Foundation of China (62072350).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The dataset is available at: https://cocodataset.org/, (accessed on 1 January 2021).

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

Self-Attention Network (SAN), Feed-Forward Network (FFN), Memory Attention Network (MAN), Layer Normalization (LN), Cross-Entropy Loss (CE), Self-Critical (SCST), BLEU@N (B@N), METEOR (M), ROUGE-L (R), CIDEr-D (C), SPICE (S).

Appendix A. Additional Material

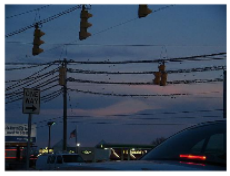
	**Base Transformer:** a young man riding a skateboard down a street. **Full-Memory Transformer:** a person riding a skateboard on a city street.	**GT1:** A group of people standing on top of a snow covered slope. **GT2:** A child in a helmet and safety pads riding a skateboard. **GT3:** A young kid in elbow and knee pads riding a skateboard. **GT4:** There is a young skateboarder riding his board. **GT5:** Young person on the street skateboarding wearing a helmet.
	**Base Transformer:** a train on the tracks at a train station. **Full-Memory Transformer:** a passenger train that is pulling into a station.	**GT1:** A yellow white and green train traveling down train tracks. **GT2:** A train under some wires at a train station. **GT3:** a railway with train parked on a track. **GT4:** Yellow commuter train at multi track station in urban setting. **GT5:** a train is moving along at the train stop.
	**Base Transformer:** cars are stopped at an intersection with traffic lights. **Full-Memory Transformer:** a city street filled with traffic and traffic lights.	**GT1:** Power lines lined with hundreds of birds at twilight. **GT2:** Traffic lights are hanging from power lines with birds on them. **GT3:** A great many birds sit on power lines. **GT4:** Birds cover the wires over a busy intersections. **GT5:** An evening scene shows power lines and stop lights.
	**Base Transformer:** a train that is sitting on the tracks. **Full-Memory Transformer:** a green and yellow train at a train station.	**GT1:** A train on a track pulling into a station dock. **GT2:** Passenger train stopped at a station with no people on it. **GT3:** The front of a commuter train parked by a platform. **GT4:** A yellow and green train passes through a rural area. **GT5:** A yellow and green train going down a track near a platform.
	**Base Transformer:** a yellow train car sitting in a train station. **Full-Memory Transformer:** a yellow and black train traveling down train tracks.	**GT1:** A train going down the train tracks in a building. **GT2:** an old passenger train one car looks a lot like a stage coach. **GT3:** The passenger train is traveling on the tracks. **GT4:** Old black train sitting inside a display area with open windows. **GT5:** Antique trains are parked in a terminal with flags.
	**Base Transformer:** a white clock tower with flags flying in the. **Full-Memory Transformer:** a large white clock tower with flags on top.	**GT1:** A large white building with a clock in it is surrounded by palm trees and red flags. **GT2:** A clock tower in a building surrounded by banners and trees. **GT3:** A white clock tower stands in front of some palm trees. **GT4:** Strung flags fly in front of a stone clock turret. **GT5:** An ornate clock tower surrounded by many small red flags.

Figure A1. Captions for our model and basic model and three real labels.

	**Base Transformer:** a bathroom with three sinks and a large mirror. **Full-Memory Transformer:** a bathroom with two sinks and a large mirror.	**GT1:** A large empty bathroom with a walk in shower tub. **GT2:** A large bathroom that is very well kept. **GT3:** A bathroom that has two sinks and a shower. **GT4:** THE VIEW OF A BATHROOM WITH WASHBASINS AND A LARGE MIRROR. **GT5:** A large white bathroom with white cabinets and double sinks
	**Base Transformer:** a group of people standing and looking at a cell phone. **Full-Memory Transformer:** a group of people standing next to each other.	**GT1:** A group of people trying out the new Nintendo Wii U. **GT2:** A group of young men interacting with their cell phones. **GT3:** a group of people standing near each other playing with small devices. **GT4:** The boys stand next to each other using the devices. **GT5:** A group of young men holding video game controllers.
	**Base Transformer:** a green truck parked on a dirt road. **Full-Memory Transformer:** a truck that is sitting in the grass.	**GT1:** a green truck parked in a dirt field with green shrubbery. **GT2:** A rusty green truck is parked among some weeds. **GT3:** An old fashioned green truck that is parked in a field. **GT4:** a kitty cat all curled up on it's bed. **GT5:** A rusted truck sits abandoned in some underbrush.
	**Base Transformer:** a man holding a tennis racket on a tennis court. **Full-Memory Transformer:** a man holding a tennis racquet on a tennis court.	**GT1:** A tennis player prepares to serve a tennis ball. **GT2:** a tennis player in all white playing on a court. **GT3:** A tennis player is reaching up with one arm and has a racquet in the other hand. **GT4:** The tennis player throws the ball up to serve. **GT5:** Spectators watching a man swinging at a tennis ball.
	**Base Transformer:** two slices of pizza on a plate. **Full-Memory Transformer:** a pizza sitting on top of a pan covered in cheese.	**GT1:** A pizza with sauce, spinach and cheese on a pan. **GT2:** A pizza on an upside down plate on a table. **GT3:** A SQUARE PIZZA WITH EXTRA SAUCE ON TOP. **GT4:** A large type pizza with cheese, spinach, and sauce is on a silver plate. **GT5:** a pizza with some extra sauce on the top of it.
	**Base Transformer:** a baseball player throwing a ball on a field. **Full-Memory Transformer:** a baseball player pitching a ball on top of a field.	**GT1:** A couple of baseball players in uniform standing in a field. **GT2:** A baseball player contorts his body as he throws a ball. **GT3:** A man in a pitching pose with a glove near another man with a glove. **GT4:** A baseball player is running to catch a ball. **GT5:** A pitcher has just thrown a baseball and another player is in the background.

Figure A2. Captions for our model and basic model and three real labels.

References

1. Hu, Z.; Yang, Z.; Liang, X.; Salakhutdinov, R.; Xing, E.P. Toward controlled generation of text. In Proceedings of the International Conference on Machine Learning, San Juan, PR, USA, 29 June–2 July 2000; pp. 1587–1596.
2. Johnson, J.; Karpathy, A.; Fei-Fei, L. Densecap: Fully convolutional localization networks for dense captioning. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, San Juan, PR, USA, 17–19 June 1997; pp. 4565–4574.
3. Anderson, P.; He, X.; Buehler, C.; Teney, D.; Johnson, M.; Gould, S.; Zhang, L. 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, San Juan, PR, USA, 17–19 June 1997; pp. 6077–6086.
4. Xu, K.; Ba, J.; Kiros, R.; Cho, K.; Courville, A.; Salakhudinov, R.; Zemel, R.; Bengio, Y. Show, attend and tell: Neural image caption generation with visual attention. In Proceedings of the International Conference on Machine Learning, Lille, France, 6–11 July 2015; pp. 2048–2057.
5. Rennie, S.J.; Marcheret, E.; Mroueh, Y.; Ross, J.; Goel, V. Self-critical sequence training for image captioning. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, San Juan, PR, USA, 17–19 June 1997; pp. 7008–7024.
6. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. *Adv. Neural Inf. Process. Syst.* **2017**, *30*. Available online: https://proceedings.neurips.cc/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html (accessed on 17 November 2022).

7. Papineni, K.; Roukos, S.; Ward, T.; Zhu, W.J. Bleu: A method for automatic evaluation of machine translation. In Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics, Philadelphia, PA, USA, 7–12 July 2002; pp. 311–318.
8. Anderson, P.; Fernando, B.; Johnson, M.; Gould, S. Spice: Semantic propositional image caption evaluation. In Proceedings of the European Conference on Computer Vision, Florence, Italy, 7–13 October 2012; pp. 382–398.
9. Yao, B.Z.; Yang, X.; Lin, L.; Lee, M.W.; Zhu, S.C. I2t: Image parsing to text description. *Proc. IEEE* **2010**, *98*, 1485–1508. [CrossRef]
10. Karpathy, A.; Fei-Fei, L. Deep visual-semantic alignments for generating image descriptions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, San Juan, PR, USA, 17–19 June 1997; pp. 3128–3137.
11. Mitchell, M.; Dodge, J.; Goyal, A.; Yamaguchi, K.; Stratos, K.; Han, X.; Mensch, A.; Berg, A.; Berg, T.; Daumé III, H. Midge: Generating image descriptions from computer vision detections. In Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics, Avignon, France, 23–27 April 2012; pp. 747–756.
12. Devlin, J.; Cheng, H.; Fang, H.; Gupta, S.; Deng, L.; He, X.; Zweig, G.; Mitchell, M. Language models for image captioning: The quirks and what works. *arXiv* **2015**, arXiv:1505.01809.
13. Liu, F.; Ren, X.; Liu, Y.; Wang, H.; Sun, X. simnet: Stepwise image-topic merging network for generating detailed and comprehensive image captions. *arXiv* **2018**, arXiv:1808.08732.
14. Xu, Y.; Wu, B.; Shen, F.; Fan, Y.; Zhang, Y.; Shen, H.T.; Liu, W. Exact adversarial attack to image captioning via structured output learning with latent variables. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 4135–4144.
15. Wang, W.; Chen, Z.; Hu, H. Hierarchical attention network for image captioning. In Proceedings of the AAAI Conference on Artificial Intelligence, Honolulu, HI, USA, 27 January–1 February 2019; Volume 33, pp. 8957–8964.
16. Huang, L.; Wang, W.; Xia, Y.; Chen, J. Adaptively aligned image captioning via adaptive attention time. *Adv. Neural Inf. Process. Syst.* **2019**, *32*. Available online: https://proceedings.neurips.cc/paper/2019/file/fecc3a370a23d13b1cf91ac3c1e1ca92-Paper.pdf (accessed on 17 November 2022).
17. Ramanishka, V.; Das, A.; Zhang, J.; Saenko, K. Top-down visual saliency guided by captions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, San Juan, PR, USA, 17–19 June 1997; pp. 7206–7215.
18. Dai, B.; Fidler, S.; Urtasun, R.; Lin, D. Towards diverse and natural image descriptions via a conditional gan. In Proceedings of the IEEE International Conference on Computer Vision, San Juan, PR, USA, 17–19 June 1997; pp. 2970–2979.
19. Gao, Y.; Beijbom, O.; Zhang, N.; Darrell, T. Compact bilinear pooling. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, San Juan, PR, USA, 17–19 June 1997; pp. 317–326.
20. Ranzato, M.; Chopra, S.; Auli, M.; Zaremba, W. Sequence level training with recurrent neural networks. *arXiv* **2015**, arXiv:1511.06732.
21. Chen, C.; Mu, S.; Xiao, W.; Ye, Z.; Wu, L.; Ju, Q. Improving image captioning with conditional generative adversarial nets. In Proceedings of the AAAI Conference on Artificial Intelligence, Honolulu, HI, USA, 27 January–1 February 2019; Volume 33, pp. 8142–8150.
22. Gu, J.; Cai, J.; Wang, G.; Chen, T. Stack-captioning: Coarse-to-fine learning for image captioning. In Proceedings of the AAAI Conference on Artificial Intelligence, Honolulu, HI, USA, 27 January–1 February 2018; Volume 32.
23. Lu, J.; Xiong, C.; Parikh, D.; Socher, R. 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, San Juan, PR, USA, 17–19 June 1997; pp. 375–383.
24. You, Q.; Jin, H.; Wang, Z.; Fang, C.; Luo, J. Image captioning with semantic attention. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, San Juan, PR, USA, 17–19 June 1997; pp. 4651–4659.
25. Pedersoli, M.; Lucas, T.; Schmid, C.; Verbeek, J. Areas of attention for image captioning. In Proceedings of the IEEE International Conference on Computer Vision, San Juan, PR, USA, 17–19 June 1997; pp. 1242–1250.
26. Lu, J.; Yang, J.; Batra, D.; Parikh, D. Neural baby talk. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, San Juan, PR, USA, 17–19 June 2018; pp. 7219–7228.
27. Yang, X.; Tang, K.; Zhang, H.; Cai, J. Auto-encoding scene graphs for image captioning. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 10685–10694.
28. Chen, S.; Jin, Q.; Wang, P.; Wu, Q. Say as you wish: Fine-grained control of image caption generation with abstract scene graphs. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 9962–9971.
29. Mathews, A.; Xie, L.; He, X. Semstyle: Learning to generate stylised image captions using unaligned text. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, San Juan, PR, USA, 17–19 June 1997; pp. 8591–8600.
30. Cornia, M.; Baraldi, L.; Cucchiara, R. Show, control and tell: A framework for generating controllable and grounded captions. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, San Juan, PR, USA, 17–19 June 1997; pp. 8307–8316.
31. Chunseong Park, C.; Kim, B.; Kim, G. Attend to you: Personalized image captioning with context sequence memory networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, San Juan, PR, USA, 17–19 June 1997; pp. 895–903.

32. Xiong, R.; Yang, Y.; He, D.; Zheng, K.; Zheng, S.; Xing, C.; Zhang, H.; Lan, Y.; Wang, L.; Liu, T. On layer normalization in the transformer architecture. In Proceedings of the International Conference on Machine Learning, San Juan, PR, USA, 29 June–2 July 2000; pp. 10524–10533.
33. Jiang, W.; Ma, L.; Jiang, Y.G.; Liu, W.; Zhang, T. Recurrent fusion network for image captioning. In Proceedings of the European conference on computer vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 499–515.
34. Yang, Z.; Dai, Z.; Yang, Y.; Carbonell, J.; Salakhutdinov, R.R.; Le, Q.V. Xlnet: Generalized autoregressive pretraining for language understanding. *Adv. Neural Inf. Process. Syst.* **2019**, *32*. Available online: https://proceedings.neurips.cc/paper/2019/hash/dc6a7e655d7e5840e66733e9ee67cc69-Abstract.html (accessed on 17 November 2022).
35. Zhuang, L.; Wayne, L.; Ya, S.; Jun, Z. A robustly optimized BERT pre-training approach with post-training. In Proceedings of the 20th Chinese National Conference on Computational Linguistics, Hohhot, China, 13–15 August 2021; pp. 1218–1227.
36. Devlin, J.; Chang, M.W.; Lee, K.; Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv* **2018**, arXiv:1810.04805.
37. Dai, Z.; Yang, Z.; Yang, Y.; Carbonell, J.; Le, Q.V.; Salakhutdinov, R. Transformer-xl: Attentive language models beyond a fixed-length context. *arXiv* **2019**, arXiv:1901.02860.
38. Ravula, A.; Alberti, C.; Ainslie, J.; Yang, L.; Pham, P.M.; Wang, Q.; Ontanon, S.; Sanghai, S.K.; Cvicek, V.; Fisher, Z. ETC: Encoding long and structured inputs in transformers. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), Online, 16–20 November 2020. Available online: https://aclanthology.org/volumes/2020.emnlp-demos/ (accessed on 17 November 2022).
39. Radford, A.; Wu, J.; Child, R.; Luan, D.; Amodei, D.; Sutskever, I. Language models are unsupervised multitask learners. *OpenAI blog* **2019**, *1*, 9.
40. Kwiatkowski, T.; Palomaki, J.; Redfield, O.; Collins, M.; Parikh, A.; Alberti, C.; Epstein, D.; Polosukhin, I.; Devlin, J.; Lee, K.; et al. Natural questions: A benchmark for question answering research. *Trans. Assoc. Comput. Linguist.* **2019**, *7*, 453–466. [CrossRef]
41. Herdade, S.; Kappeler, A.; Boakye, K.; Soares, J. Image captioning: Transforming objects into words. *Adv. Neural Inf. Process. Syst.* **2019**, *32*. Available online: https://proceedings.neurips.cc/paper/2019/hash/680390c55bbd9ce416d1d69a9ab4760d-Abstract.html (accessed on 17 November 2022).
42. Osolo, R.I.; Yang, Z.; Long, J. An Attentive Fourier-Augmented Image-Captioning Transformer. *Appl. Sci.* **2021**, *11*, 8354. [CrossRef]
43. Jia, X.; Wang, Y.; Peng, Y.; Chen, S. Semantic association enhancement transformer with relative position for image captioning. *Multimed. Tools Appl.* **2022**, *15*, 1–19. [CrossRef]
44. Lin, T.Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; Zitnick, C.L. Microsoft coco: Common objects in context. In Proceedings of the European Conference on Computer Vision, Florence, Italy, 7–13 October 2012; Springer: Berlin/Heidelberg, Germany, 2012; pp. 740–755.
45. Vedantam, R.; Lawrence Zitnick, C.; Parikh, D. Cider: Consensus-based image description evaluation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, San Juan, PR, USA, 17–19 June 1997; pp. 4566–4575.
46. Banerjee, S.; Lavie, A. 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, Ann Arbor, MI, USA, 29 June 2005; pp. 65–72.
47. Lin, C.Y. Rouge: A package for automatic evaluation of summaries. In Proceedings of the Text Summarization Branches Out, Barcelona, Spain, 25–26 July 2004; pp. 74–81.