MATIC: Memory-Guided Adaptive Transformer for Image Captioning

Gaurav O. Gajhiye and Abhijeet V. Nandedkar

1 Introduction

Image captioning task entails the interpretation of visual contents and its description in natural linguistic manner automatically [6, 13, 23, 24, 27]. In the past few years, the topic has gained popularity in the field of artificial intelligence due to the cross-modal interaction of vision and language modality and abundant research is still being conducted to explore the connectivity between vision-language modeling (e.g. dense captioning, visual question answering, video captioning, and cross-modal retrieval) [1, 9, 12, 25]. The image captioning task was fascinated by sequence learning and machine translation [2] and was primarily addressed by encoder-decoder frameworks. The encoder substantially extracts visual characteristics using variants of convolutional neural network (CNN) and the decoder faithfully constructs the caption using forms of recurrent neural networks (RNN). Further, the attention mechanism [16] was equipped with a CNN-RNN structure to attend to prominent visual regions while generating the word sequence. Despite these advances, the association of objects, attributes, and their intrinsic relationship to describe images remains the topic of intense research.

In CNN-RNN-based structures, the pre-trained CNN methods for visual features were used to capture dominant visual attributes but were incompetent for capturing inherent visual knowledge for captioning. The long-short term memory (LSTM) [10] was commonly utilized as a form of RNN to have a long-term dependency on linguistic patterns and generates the next sample by operating on the hidden state of

G. O. Gajhiye (✉) · A. V. Nandedkar
CVPR Lab, Shri Guru Gobind Singhji Institute of Engineering and Technology (SGGSIE&T), Nanded, India
e-mail: gajbhiyegaurav@sggs.ac.in

A. V. Nandedkar
e-mail: avnandedkar@sggs.ac.in

U. Mudenagudi et al. (eds.), *Proceedings of the Satellite Workshops of ICVGIP 2021*,
Lecture Notes in Electrical Engineering 924,
https://doi.org/10.1007/978-981-19-4136-8_15

the current time step. This regressive nature of LSTM does not allow to parallelize the training procedure.

The novel Transformer [21] architecture has shown the significant potential in addressing the sequence modeling tasks like language generation and translation, as well as multi-modal sequential learning [8]. The standard Transformer is consist of an encoder-decoder model, where the encoder represents the stack of self-attention module followed by feed-forward network and the decoder represents the stack of self and cross attention module followed by feed-forward network. The autoregressive nature of Transformer extends its capability with the stack of attention modules and position-wise embedding of input sequences for parallelism. This motivates to investigate the utility of Transformer for describing the contents of a visual scene by extracting inherent scenery knowledge. Inspired by Cornia et al. [8], this work targets the investigation of memory vectors in a visual encoder to determine the correspondence between objects and attributes using CNN features. In this work, a novel Memory-guided Adaptive Transformer is proposed with a memory-guided encoder for preserving intrinsic visual information received from traditional CNN, while the decoder connects the visual and linguistic features by learning inter-modal association with an adaptive gating mechanism for image description. The overall contributions of the work are as follows:

- Single layer of the memory-guided encoder in conjunction with conventional convolution network is presented for finding the inherent relationship (such as colors, positions, gender, and background) within objects and understand scene attributes by updating memory parameters.
- Multiple layers of the decoder with adaptive multi-headed attention modules, co-relate the visual and linguistic pattern by assigning adaptive weightage to spatial and language attention for predicting the future word sample.
- A novel Memory-guided Adaptive Transformer for Image Captioning (MATIC) is proposed by incorporating a single memory-guided encoder layer with multiple adaptive attention decoder layers, and its performance is validated on Flickr8k [19] and Flickr30k [28] dataset.

2 Related Work

The image captioning task became one of the vital issues in artificial intelligence and has been widely addressed by numerous methodologies in the past few years with advancements in deep learning algorithms. In this section, based on the architectural design, the literature is divided into two subgroups as (i) CNN-RNN-based models and (ii) Transformer-based models.

2.1 CNN-RNN Based Models

In CNN-RNN-based encoder-decoder methods, CNNs were broadly employed as a visual encoder for spatial-regional characteristics and RNNs were adopted as decoders for the generation of word sequences. In earlier study [11, 13, 17, 23], various levels of CNN were used for spatial features and visual regions extraction and trained with a language model consisting of RNN layers to optimise the likelihood probability of word sequences given the image. The notion of attention in machine translation [2] was utilized in image captioning to attend to prominent visual features aligned with each word in sequence [26]. Later, the purpose attention network was enhanced in [27] by combining attending on visual regions and visual semantic attributes with RNN for better caption prediction. To collect more fine-grained information about visual scenes, channel and spatial-wise attention was introduced with CNN [6]. The adaptive mechanism was combined with attention network [16] for providing substantial weightage to visual and linguistic models in order to generate word sequences.

2.2 Transformer Based Models

The transformer model has advanced to the cutting-edge of several essential tasks in the artificial intelligence domain, including image captioning. In Yu et al. [29] CNN-based regional encoder with self-attention has merged with Transformer decoder for transforming visual information into textual captions. The Transformer's decoder section was updated in Zhang et al. [30] to describe the visual contents sequentially, by including an adaptive mechanism in the multi-head attention component leveraging the query vector. Li et al. [14] presented a two-way encoder to process visual and semantic information with EnTangled Attention to generate captions by controlling the flow of visual and semantic knowledge simultaneously. A revolutionary captioning network was developed [8], in which memory vectors were incorporated in the visual encoding layer for acquiring co-relative prior information between image regions, and a mesh-like structure was followed to connect encoder and decoder layer outputs. The scene graphs were built and fused with decoder output using the attention module for sequence generation in Chen et al. [5] to grasp better visual semantics relationship.

3 Methodology

3.1 Overview

This work presents the novel end-to-end attentive architecture of the Memory-guided Adaptive Transformer for Image Captioning (MATIC), which comprises of single-layer encoder and a multi-layered decoder. Figure 1 depicts the overall framework, in which the encoder employs spatial information recovered from CNN and the decoder uses textual features based on FastText embedding to generate caption. The encoder learns inherent relationships within the objects using scenery knowledge of visual features, while the decoder adaptively controls the attentive visual and semantic information by conditioning memory-guided encoder output and embedded textual output.

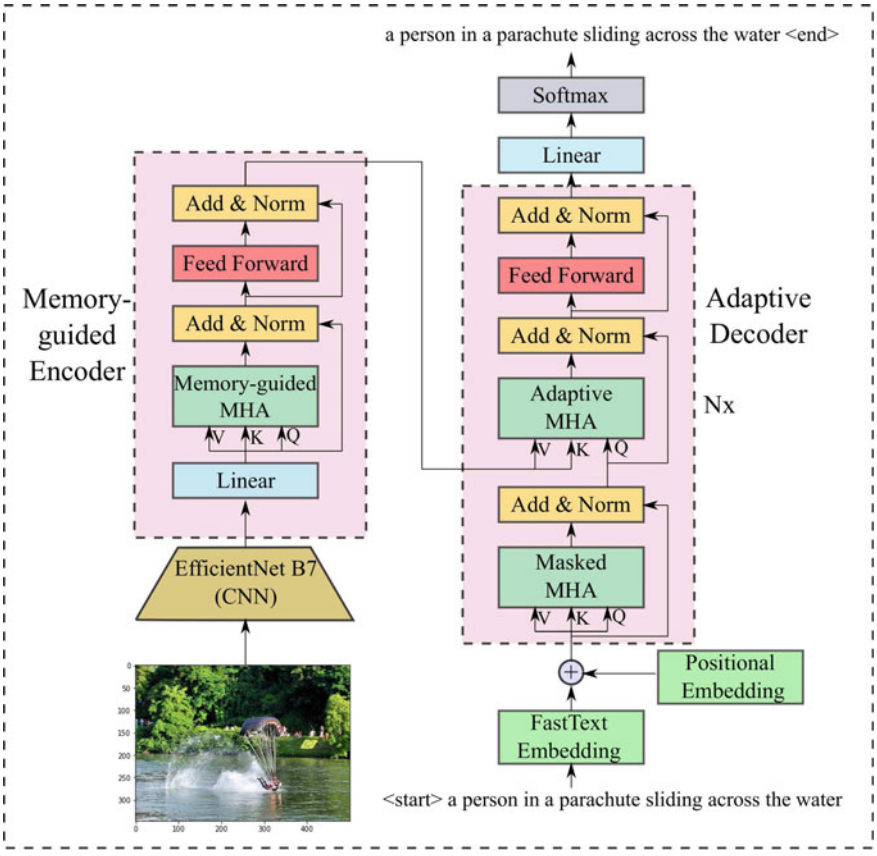

Fig. 1 Memory-guided adaptive transformer for image captioning

3.2 Visual Encoder

The strength of EfficientNet [20] by scaling compound parameters (depth, width and resolution) enhances the efficiency of classification as well as the transfer learning challenge, making it the preferred method for extracting higher-level spatial information from the image. The last convolutional layer employs spatial information by providing the regional feature maps of the image in the form of $V_s = V_1, V_2, .., V_D$, where $V_i \in \mathbb{R}^{H \times W}$ (Here, H, W, D represents Height, Width, and Depth of feature maps). Every feature map is flattened to convert the 3D representation into 2D representation, which allows the visual encoder to determine the distinguish relationship within regional features. The 2D representation of spatial features can be rewrite as $V_s \in \mathbb{R}^{F \times D}$, where F represents flatten dimension ($H * W$). These flattened spatial features are provided to the memory-guided encoder to acquire close relationships within various objects.

3.2.1 Memory-Guided Multi-Head Attention

Generally in Transformer, the multi-head attention (MHA) [21] computes the similarity between query (Q) and key (K) vectors and then maps them with value (V) vector to correlate outputs by the parallel projection of the query, key, and value vectors into distinguish head components. It can be represented as follows:

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

$$MultiHead(Q, K, V) = Concat(H_1, H_2, ..H_h)W^O \tag{2}$$

$$H_i = Attention(W_i^q Q, W_i^k K, W_i^v V) \tag{3}$$

where d_k is the scaling factor, h represent the number of heads, and W^O, W^q, W^k, and W^v are projected weight parameters.

To preserve complete depthwise regional information, spatial features (V_s) from feature maps with higher dimensionality are linearly projected to intermediate state dimensionality of Transformer model (d_{model}) with ReLU activation. These linearly projected features are further inferred as multi-scale inputs *viz.* Query (Q), Key (K) and Value (V) for memory-guided MHA, where $Q, K, V \in \mathbb{R}^{F \times d_{model}}$. In order to acquire the inherent information within the image, the learnable memory elements (m) are appended to the key and value vector. The spatial input vectors (Q, K, V) are linearly transformed with the projection parameter as:

$$Q = W_{d_{model}}^q Q, \; K = W_{d_{model}}^k K, \; V = W_{d_{model}}^v V \tag{4}$$

where $W_{d_{model}}$ represents the projection parameter of Q, K and V. The memory based key (K_m) and value (V_m) vectors are updated by including memory slots (m) of dimension $m \in \mathbb{R}^{m \times d_{model}}$. Thus memory based key and value vectors becomes $K_m, V_m \in \mathbb{R}^{(F+m) \times d_{model}}$:

$$K_m = [K : W_m^k m], \;\; V_m = [V : W_m^v m] \tag{5}$$

Here, [:] defines vertical concatenation operation and memory matrix with (m) rows is generated for keys and values by xavier uniform initializer, which gets updated by trainable weight parameters (W_m^k) and (W_m^v) respectively. Analytically, Memory-guided MHA is computed as given below with $Mem\_MHA \in \mathbb{R}^{F \times d_{model}}$:

$$Mem\_MHA(Q, K, V) = MultiHead(Q, K_m, V_m) \tag{6}$$

3.2.2 Full Encoder

The output of Memory-guided MHA is passed to a position-wise feed-forward network comprising two linear layers with ReLU activation and operates as:

$$FF(x) = W_1(max(0, W_2 x + b)) + c \tag{7}$$

where W_1 and W_2 are outer and internal weight parameters, while b and c are bias terms.

The complete encoder combines Memory-guided MHA and Feed Forward modules by residual additive connection and normalization layer ($Norm$) for yielding encoded output (enc_{out}) as follows:

$$enc_1 = Norm(Mem\_MHA(Q, K, V) + Q) \tag{8}$$

$$enc_{out} = Norm(FF(enc_1) + enc_1) \tag{9}$$

3.3 Linguistic Decoder

Following the standard Transformer, the proposed architecture also utilizes N identical layers of decoder, in which multi-modal MHA is modified by including a conditional gating mechanism for weighting the attentive linguistic and spatial information. To acquire the concrete numerical representation of word sequence, FastText [4] model is trained on captions and used to generate word embedding of respective caption ($WE \in \mathbb{R}^{L \times d_{model}}$). Here, L and d_{model} are representing maximum caption length and dimensionality of embedding respectively. In order to access the relative position of the word sequence, positional embedding is added with word embedding output.

The masked MHA sublayer allows the model to attend to all previous time step linguistic information to generate the current time step sample.

3.3.1 Adaptive Multi Head Attention

Inspired from the work [16], adaptive gating mechanism is incorporated with MHA sub-module of Transformer decoder for sequence modeling. The cross-MHA of decoder layer is updated with encoder's output and attentive previous steps linguistic output. From memory-guided visual encoder, output of feed forward network ($enc_{out} \in \mathbb{R}^{F \times d_{model}}$) is adopted for extracting inherent visual characteristics with relative object information as (V) and (K), while shifted attentive linguistic knowledge from masked MHA of decoder ($dec_1 \in \mathbb{R}^{L \times d_{mdoel}}$) is used as query matrix (Q). The $Attention$ in equation (1) is modified by introducing adaptive gating mechanism ($\hat{\beta}$) as shown in Fig. 2 for conditioning spatial knowledge and linguistic pattern. The adaptive gating parameter ($\hat{\beta}$) works similarly as sentinel gate in Lu et al. [16].

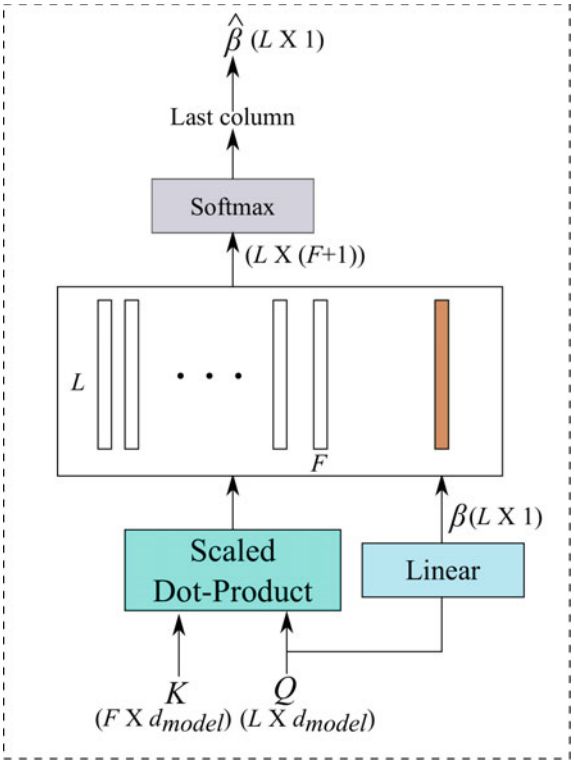

Fig. 2 Adaptive gate

$$Adp\_Attention(Q, K, V) = \hat{\beta} * Q + (1 - \hat{\beta}) * Attention(Q, K, V) \quad (10)$$

Here, $(*)$ represents element-wise multiplication and $(\hat{\beta})$ represents scaler value within range [0, 1], where 0 indicates flow of spatial information and 1 implies flow of linguistic knowledge.

In order to co-relate linguistic and spatial information for generation of next word sample, β parameter is introduced, which is computed by projecting the query matrix (Q) linearly into single dimension $(\beta \in \mathbb{R}^{L\times 1})$ as follows:

$$\beta = tanh(W_\beta(Q)) \quad (11)$$

The dot-product (α) is used to obtained adaptive gate $(\hat{\beta})$ using (β) parameter as follows:

$$\alpha = softmax([\frac{QK^T}{\sqrt{d_k}}] : \beta) \quad (12)$$

$$\hat{\beta} = \alpha[:, -1] \quad (13)$$

Here, $QK^T \in \mathbb{R}^{L\times F}$ and dot-product produces matrix with dimensionality as $\alpha \in \mathbb{R}^{L\times(F+1)}$, from which last column is extracted to retrieve adaptive gate $(\hat{\beta})$. The $(\hat{\beta})$ exhibits the multinomial probability distribution of the query, thus signifying which elements from the query preserve essential linguistic information. Overall, $(\hat{\beta})$ is trained to generate syntactically correct words (e.g. at, on, with, in, through, etc.) by weighting linguistic information, while generating contextual words (e.g. color, gender, position, shape, etc.) by weighting multilevel spatial information.

3.3.2 Full Decoder

The proposed decoder works similarly to a traditional Transformer decoder with autoregressive training properties. The decoder consists of two MHA sub-modules, first module attends on previous textual embedding by hiding future information to generate the current step word sample, while the second module co-relates visual and linguistic patterns. The feed-forward network and residual connections are incorporated to complete the decoder. The FastText embedding of the caption is fed as linguistic input to the decoder, while sinusoidal positional embedding from base transformer [21] is used to co-relate the absolute positioning of each word token. The complete decoder follows the given operations:

$$dec_{emb} = FastText_{emb} + Positional_{emb} \quad (14)$$

$$dec_1 = Norm(Masked\_MHA(dec_{emb}, dec_{emb}, dec_{emb}) + dec_{emb}) \quad (15)$$

$$dec_2 = Norm(Adp\_Attention(dec_1, enc_{out}, enc_{out}) + dec_1) \tag{16}$$

$$dec_{out} = Norm(FF(dec_2) + dec_2) \tag{17}$$

To generate the sequence of words (caption) for the image, the output of 'N' layered-decoder module is linearly transformed to vocabulary size ($vocab$) for retrieving the probability distribution of the next word.

$$P(w) = Softmax(W_{vocab} dec_{out}) \tag{18}$$

Here, W_{vocab} defines the trainable weight parameter for vocabulary size.

3.4 Training

The aim of proposed model is to minimize standard cross-entropy loss ($\mathcal{L}_{XE}$) of word sequence ($y^*_{1:T}$) given spatial features (V_s) of target image as follows:

$$\mathcal{L}_{XE} = -\sum_{t=1}^{L} log(P(y^*_t | y^*_{0:t-1}; V_s; \theta)) \tag{19}$$

Here, L is the maximum word length of the caption, while P defines the softmax probability of t-th word as given in Eq. (18) and θ defines model hyper-parameters.

4 Experiments and Results

4.1 Dataset

To evaluate the performance of proposed model, Flickr8k [19] and Flickr30k [28] datasets are utilized, in which each image is associated with 5 human reference captions. Flickr8k is small-scale captioning dataset with 8000 image-caption pairs, while Flickr30k is a large scale captioning dataset with 31783 image-caption pairs. The distribution of training, validation, and test samples are given in Table 1. The maximum word length for captioning is set as 30 and 50 for Flickr8k and Flickr30k datasets respectively. All the captions are transformed to lower case and least occurred words are excluded (less than 3 occurrences for Flickr8k and less than 5 occurrence for Flickr30k) to build the final vocabulary (3427 for Flickr8k and 7037 for Flickr30K). All the images are resized to (300×300) and encoded by the EfficientNetB7 module.

Table 1 Data distribution for Flickr8k and Flickr30K

Dataset	Train	Validation	Test
Flickr8k	6000	1000	1000
Flickr30k	29783	1000	1000

4.2 Training Details

To train the proposed model, the number of heads is set to 8, embedding and sub-module dimensionality are selected as 512, while memory slots are varied from 10 to 40 with steps of 10 in memory-guided MHA. The internal layer dimension of the feed forward network is set as 2048. The Adam optimizer with warmup strategy and batch size of 128 is utilized to train the model. The warmup step size and epochs are set as 8000 and 15 and 4000 and 20 for Flickr8K and Flickr30K datasets respectively. Specifically, the proposed MATIC model is trained with a single encoder and four decoder layers ($N = 4$), which ensures better quantitative results. All proposed variants are implemented with TensorFlow 2.2 library on TITAN Xp GPU.

4.3 Quantitative and Qualitative Results

In this section, proposed method is compared with state-of-the-arts and respective quantitative results are summarized. To quantify generated captions, natural language generation (NLG) metrics e.g. n-gram Bleu [18], Meteor [3], Rouge [15] and CIDEr [22] scores are computed from MSCOCO captioning API [7]. In order to generate the fine-level precise caption, heuristic beam search algorithm is employed with beam indexing upto 3. The best metrics outcome are extracted using various beam index and reported in the quantitative results. Table 2 and Table 3 represents the comparitive analysis of quantitative results for various methods and proposed MATIC with various memory units (m) on Flickr8k and Flickr30k dataset respectively.

Certain experiments on decoder layers revealed that the proposed MATIC model works better with 4 decoder layers than 6 decoder layers, thus reducing the total trainable parameters and making it a lightweight model. Here, $Base\ XR$ represents the re-implementation of standard Transformer [21] with a single encoder and four decoder layers. Tables 2 and 3 show the quantitative effectiveness of the proposed model for generating captions with 30 memory units for Flickr8k and 40 memory units for Flickr30k dataset.

Table 4 summarises statistical analysis, which shows the amount of trainable parameters and average testing time required by the proposed architecture is approximately equal to that of the Base Transformer. The average testing time is computed by generating captions for 20 test images. With a similar amount of hyper-parameters

Table 2 Comparative analysis of NLG metrics for various methods and proposed MATIC on Flickr8k

Methods	Bleu 1	Bleu 2	Bleu 3	Bleu 4	Meteor	Rouge	CIDEr
DeepVS [13]	57.9	38.3	24.5	16.0	–	–	–
NIC [23]	63.0	41.0	27.0	–	–	–	–
Soft-Att [26]	67.0	44.8	29.9	19.5	18.5	–	–
Hard-Att [26]	67.0	45.7	31.4	21.3	20.3	–	–
g-LSTM [11]	64.7	45.9	31.8	21.2	20.6	–	–
SCA-CNN [6]	68.2	49.6	35.9	**25.8**	22.4	–	–
Base XR	66.4	47.2	33.1	22.8	21.1	54.2	47.1
MATIC (m = 10)	68.2	49.4	34.9	24.2	22.1	55.6	53.6
MATIC (m = 20)	68.3	49.4	35.1	24.4	22.1	55.4	53.3
MATIC (m = 30)	**69.3**	**50.7**	**36.5**	**25.7**	**23.3**	**56.1**	**55.7**
MATIC (m = 40)	67.7	48.7	34.3	23.7	22.9	55.3	51.7

XR represents Transformer

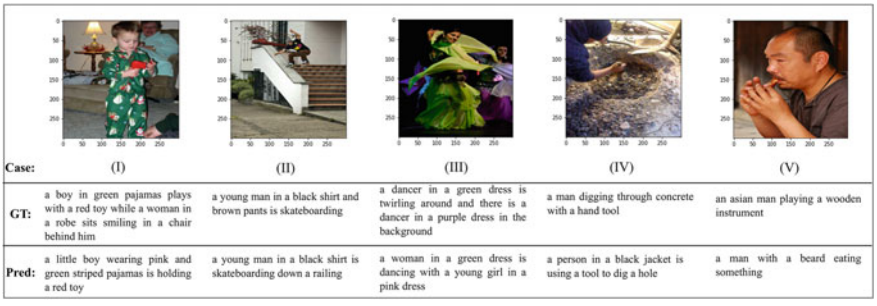

Fig. 3 Generated captions on tricky images from Flickr30K dataset

as of Base Transformer, the MATIC model exhibits a considerable improvement in captioning performance.

To assess the effectiveness of the proposed MATIC model, five tricky test images were chosen from the Flickr30K dataset, and the corresponding generated captions are shown in Fig. 3. The proposed MATIC generates excellent captions for the first four test cases by expressing minute visual contents and relative color-objects-attribute information but misleads in the fifth test instance by generating the incorrect action-based caption. The generated captions by the proposed method demonstrate the application of memory-guided encoder to capture the color, gender, and position of objects, while the adequacy of the adaptive decoder describes all minute details of the image in a semantically convenient manner.

Table 3 Comparative analysis of NLG metrics for various methods and proposed MATIC on Flickr30k

Methods	Bleu 1	Bleu 2	Bleu 3	Bleu 4	Meteor	Rouge	CIDEr
DeepVS [13]	57.3	36.9	24.0	15.7	–	–	24.7
NIC [23]	66.3	42.3	27.7	18.3	–	–	–
mRNN [17]	60.3	41.0	28.0	19.0	–	–	–
Soft-Att [26]	66.7	43.4	28.8	19.1	18.5	–	–
Hard-Att [26]	66.9	43.9	29.6	19.9	18.5	–	–
g-LSTM [11]	64.6	44.6	30.5	20.6	17.9	–	–
Sem-Att [27]	64.7	46.0	32.4	23.0	18.9	–	–
SCA-CNN [6]	66.2	46.8	32.5	22.3	19.5	–	–
Adapt-Att [16]	67.7	49.4	35.4	25.1	20.4	–	53.1
Scene-Graph XR [5]	66.9	49.4	35.4	24.8	20.3	–	**53.3**
Adapt XR [30]	67.0	49.6	35.5	25.2	20.4	–	53.0
Base XR	66.5	47.5	33.3	23.2	20.8	53.9	48.9
MATIC (m = 10)	68.1	48.8	34.6	24.3	20.8	54.0	46.7
MATIC (m = 20)	67.2	48.6	35.0	25.0	20.7	54.0	47.6
MATIC (m = 30)	68.8	49.4	35.4	24.8	20.7	54.2	49.7
MATIC (m = 40)	**69.5**	**50.7**	**36.5**	**26.0**	**20.9**	**54.9**	51.7

#XR represents Transformer

Table 4 Statistical analysis of Base Transformer and proposed MATIC on both datasets

	Flickr8k		Flickr30k	
	Base XR	MATIC	Base XR	MATIC
No. of parameters (in Millions)	24.78	25.35	28.48	29.05
Avg. test time (in Seconds)	3.59	3.68	4.63	4.67

5 Conclusion

In this work, a novel Memory-guided Adaptive Transformer for Image Captioning (MATIC) is demonstrated by merging a memory-guided encoder with an adaptive decoder, and its efficacy for generating natural captions is proven on Flickr8k and Flickr30k datasets. Memory-guided encoder is used in conjunction with a conventional CNN network to acquire innate perceptual scenery knowledge, and an Adaptive decoder is employed to align vision-language modality by conditionally weighting visual and semantic information for the generation of word sequence as the caption. In comparison to a typical Transformer with six encoder and six decoder layers, the proposed MATIC has a single memory-based encoder and four adaptive-attention-based decoder layers, which aids in the reduction of overall trainable parameters. As a result, it may be served as a lightweight model and embedded in a device for various captioning applications such as virtual aid, visually impaired individual assisting tools, scene interpretation, and many more. The proposed MATIC has outperformed state-of-the-arts in both quantitative and qualitative findings with 30 memory units for Flickr8k and 40 memory units for the Flickr30K dataset. The proposed MATIC demonstrates the effectiveness of a memory-guided encoder by understanding implicit scenery knowledge aligned with a multi-headed adaptive decoder to describe visual contents in a faithful linguistic manner.

References

1. 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. IEEE, pp 6077–6086
2. Bahdanau D, Cho K, Bengio Y (2015) Neural machine translation by jointly learning to align and translate. CoRR. arXiv:abs/1409.0473
3. Banerjee S, Lavie A (2005) METEOR: an automatic metric for MT evaluation with improved correlation with human judgments. In: Proceedings of the ACL workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization, pp 65–72
4. Bojanowski P, Grave E, Joulin A, Mikolov T (2017) Enriching word vectors with subword information. Trans Assoc Comput Linguist 5:135–146
5. Chen H, Wang Y, Yang X, Li J (2021) Captioning transformer with scene graph guiding. In: 2021 IEEE international conference on image processing (ICIP). IEEE, pp 2538–2542
6. Chen L, Zhang H, Xiao J, Nie L, Shao J, Liu W, Chua TS (2017) Sca-cnn: spatial and channel-wise attention in convolutional networks for image captioning. In: Proceedings of the IEEE conference on computer vision and pattern recognition. IEEE, pp 5659–5667
7. Chen X, Fang H, Lin TY, Vedantam R, Gupta S, Dollár P, Zitnick CL (2015) Microsoft coco captions: data collection and evaluation server. arXiv:1504.00325
8. 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. IEEE, pp 10578–10587
9. Gao L, Guo Z, Zhang H, Xu X, Shen HT (2017) Video captioning with attention-based LSTM and semantic consistency. IEEE Trans Multimed 19(9):2045–2055
10. Hochreiter S, Schmidhuber J (1997) Long short-term memory. Neural Comput 9(8):1735–1780

11. Jia X, Gavves E, Fernando B, Tuytelaars T (2015) Guiding the long-short term memory model for image caption generation. In: Proceedings of the IEEE international conference on computer vision. IEEE, pp 2407–2415
12. Johnson J, Karpathy A, Fei-Fei L (2016) Densecap: fully convolutional localization networks for dense captioning. In: Proceedings of the IEEE conference on computer vision and pattern Recognition. IEEE, pp 4565–4574
13. 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. IEEE, pp. 3128–3137
14. 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. IEEE, pp 8928–8937
15. Lin CY (2004) Rouge: a package for automatic evaluation of summaries. In: Text summarization branches out, pp 74–81
16. 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. IEEE, pp 375–383
17. Mao J, Xu W, Yang Y, Wang J, Huang Z, Yuille A (2015) Deep captioning with multimodal recurrent neural networks (m-rnn)
18. 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 on association for computational linguistics. Association for Computational Linguistics, pp 311–318
19. Rashtchian C, Young P, Hodosh M, Hockenmaier J (2010) Collecting image annotations using Amazon's Mechanical Turk. In: Proceedings of the NAACL HLT 2010 workshop on creating speech and language data with Amazon's Mechanical Turk, pp 139–147
20. Tan M, Le Q (2019) Efficientnet: rethinking model scaling for convolutional neural networks. In: International conference on machine learning. PMLR, pp 6105–6114
21. Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, Kaiser LU, Polosukhin I (2017) Attention is all you need. In: Guyon I, Luxburg UV, Bengio S, Wallach H, Fergus R, Vishwanathan S, Garnett R (eds) Advances in neural information processing systems, vol 30. Curran Associates, Inc. https://proceedings.neurips.cc/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf
22. 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. IEEE, pp 4566–4575
23. 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. IEEE, pp 3156–3164
24. Vinyals O, Toshev A, Bengio S, Erhan D (2016) Show and tell: Lessons learned from the 2015 mscoco image captioning challenge. IEEE transactions on pattern analysis and machine intelligence 39(4):652–663
25. Wang B, Yang Y, Xu X, Hanjalic A, Shen HT (2017) Adversarial cross-modal retrieval. In: Proceedings of the 25th ACM international conference on multimedia, pp 154–162
26. Xu K, Ba J, Kiros R, Cho K, Courville A, Salakhudinov R, Zemel R, Bengio Y (2015) Show, attend and tell: neural image caption generation with visual attention. In: International conference on machine learning. PMLR, pp 2048–2057
27. You Q, Jin H, Wang Z, Fang C, Luo J (2016) Image captioning with semantic attention. In: Proceedings of the IEEE conference on computer vision and pattern recognition. IEEE, pp 4651–4659
28. Young P, Lai A, Hodosh M, Hockenmaier J (2014) From image descriptions to visual denotations: new similarity metrics for semantic inference over event descriptions. Trans Assoc Comput Linguist 2:67–78
29. Yu J, Li J, Yu Z, Huang Q (2019) Multimodal transformer with multi-view visual representation for image captioning. IEEE Trans Circuits Syst Video Technol 30(12):4467–4480

30. Zhang W, Nie W, Li X, Yu Y (2019) Image caption generation with adaptive transformer. In: 2019 34rd Youth academic annual conference of Chinese association of automation (YAC). IEEE, pp 521–526