Meshed-Memory Transformer for Image Captioning

Marcella Cornia* Matteo Stefanini* Lorenzo Baraldi* Rita Cucchiara
University of Modena and Reggio Emilia
{name.surname}@unimore.it

Abstract

Transformer-based architectures represent the state of the art in sequence modeling tasks like machine translation and language understanding. Their applicability to multi-modal contexts like image captioning, however, is still largely under-explored. With the aim of filling this gap, we present $\mathcal{M}^2$ – a Meshed Transformer with Memory for Image Captioning. The architecture improves both the image encoding and the language generation steps: it learns a multi-level representation of the relationships between image regions integrating learned a priori knowledge, and uses a mesh-like connectivity at decoding stage to exploit low- and high-level features. Experimentally, we investigate the performance of the $\mathcal{M}^2$ Transformer and different fully-attentive models in comparison with recurrent ones. When tested on COCO, our proposal achieves a new state of the art in single-model and ensemble configurations on the "Karpathy" test split and on the online test server. We also assess its performances when describing objects unseen in the training set. Trained models and code for reproducing the experiments are publicly available at: https://github.com/aimagelab/meshed-memory-transformer.

1. Introduction

Image captioning is the task of describing the visual content of an image in natural language. As such, it requires an algorithm to understand and model the relationships between visual and textual elements, and to generate a sequence of output words. This has usually been tackled via Recurrent Neural Network models [40, 15, 43, 42, 7], in which the sequential nature of language is modeled with the recurrent relations of either RNNs or LSTMs. Additive attention or graph-like structures [46] are often added to the recurrence [43, 12] in order to model the relationships between image regions, words, and eventually tags [20].

This schema has remained the dominant approach in the last few years, with the exception of the investigation of Convolutional language models [5], which however did not become a leading choice. The recent advent of fully-attentive models, in which the recurrent relation is abandoned in favour of the use of self-attention, offers unique opportunities in terms of set and sequence modeling performances, as testified by the Transformer [37] and BERT [8] models and their applications to retrieval [33] and video understanding [35]. Also, this setting offers novel architectural modeling capabilities, as for the first time the attention operator is used in a multi-layer and extensible fashion. Nevertheless, the multi-modal nature of image captioning demands for specific architectures, different from those employed for the understanding of a single modality.

Following this premise, we investigate the design of a novel fully-attentive approach for image captioning. Our architecture takes inspiration from the Transformer model [37] for machine translation and incorporates two key novelties with respect to all previous image captioning algorithms: (*i*) image regions and their relationships are

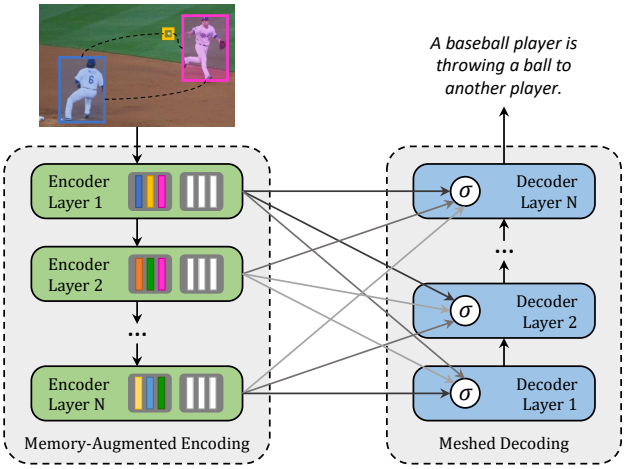


Figure 1: Our image captioning approach encodes relationships between image regions exploiting learned a priori knowledge. Multi-level encodings of image regions are connected to a language decoder through a meshed and learnable connectivity.

*Equal contribution.

encoded in a multi-level fashion, in which low-level and high-level relations are taken into account. When modeling these relationships, our model can learn and encode a priori knowledge by using persistent *memory vectors*. (*ii*) The generation of the sentence, done with a multi-layer architecture, exploits both low- and high-level visual relationships instead of having just a single input from the visual modality. This is achieved through a learned gating mechanism, which weights multi-level contributions at each stage. As this creates a mesh connectivity schema between encoder and decoder layers, we name our model *Meshed-Memory Transformer* – $\mathcal{M}^2$ Transformer for short. Figure 1 depicts a schema of the architecture.

Experimentally, we explore different fully-attentive baselines and recent proposals, gaining insights on the performance of fully-attentive models in image captioning. Our $\mathcal{M}^2$ Transformer, when tested on the COCO benchmark, achieves a new state of the art on the "Karpathy" test set, on both single-model and ensemble configurations. Most importantly, it surpasses existing proposals on the online test server, *ranking first among published algorithms*.

Contributions. To sum up, our contributions are as follows:

- We propose a novel fully-attentive image captioning algorithm. Our model encapsulates a multi-layer encoder for image regions and a multi-layer decoder which generates the output sentence. To exploit both low-level and high-level contributions, encoding and decoding layers are connected in a mesh-like structure, weighted through a learnable gating mechanism;
- In our visual encoder, relationships between image regions are encoded in a multi-level fashion exploiting learned a priori knowledge, which is modeled via persistent memory vectors;
- We show that the $\mathcal{M}^2$ Transformer surpasses all previous proposals for image captioning, achieving a new state of the art on the online COCO evaluation server;
- As a complementary contribution, we conduct experiments to compare different fully-attentive architectures on image captioning and validate the performance of our model on novel object captioning, using the recently proposed nocaps dataset. Finally, to improve reproducibility and foster new research in the field, we will publicly release the source code and trained models of all experiments.

2. Related work

A broad collection of methods have been proposed in the field of image captioning in the last few years. Earlier captioning approaches were based on the generation of simple templates, filled by the output of an object detector or attribute predictor [32, 45]. With the advent of Deep Neural Networks, most captioning techniques have employed RNNs as language models and used the output of one or more layers of a CNN to encode visual information and condition language generation [41, 31, 9, 14]. On the training side, while initial methods were based on a time-wise cross-entropy training, a notable achievement has been made with the introduction of Reinforcement Learning, which enabled the use of non-differentiable caption metrics as optimization objectives [31, 29, 23]. On the image encoding side, instead, single-layer attention mechanisms have been adopted to incorporate spatial knowledge, initially from a grid of CNN features [43, 24, 48], and then using image regions extracted with an object detector [4, 27, 25]. To further improve the encoding of objects and their relationships, Yao *et al*. [46] have proposed to use a graph convolution neural network in the image encoding phase to integrate semantic and spatial relationships between objects. On the same line, Yang *et al*. [44] used a multi-modal graph convolution network to modulate scene graphs into visual representations.

Despite their wide adoption, RNN-based models suffer from their limited representation power and sequential nature. After the emergence of Convolutional language models, which have been explored for captioning as well [5], new fully-attentive paradigms [37, 8, 34] have been proposed and achieved state-of-the-art results in machine translation and language understanding tasks. Likewise, some recent approaches have investigated the application of the Transformer model [37] to the image captioning task.

In a nutshell, the Transformer comprises an encoder made of a stack of self-attention and feed-forward layers, and a decoder which uses self-attention on words and cross-attention over the output of the last encoder layer. Herdade *et al*. [11] used the Transformer architecture for image captioning and incorporated geometric relations between detected input objects. In particular, they computed an additional geometric weight between object pairs which is used to scale attention weights. Li *et al*. [20] used the Transformer in a model that exploits visual information and additional semantic knowledge given by an external tagger. On a related line, Huang *et al*. [12] introduced an extension of the attention operator in which the final attended information is weighted by a gate guided by the context. In their approach, a Transformer-like encoder was paired with an LSTM decoder. While the aforementioned approaches have exploited the original Transformer architecture, in this paper we devise a novel fully-attentive model that improves the design of both the image encoder and the language decoder, introducing two novel attention operators and a different design of the connectivity between encoder and decoder.

3. Meshed-Memory Transformer

Our model can be conceptually divided into an encoder and a decoder module, both made of stacks of attentive layers. While the encoder is in charge of processing regions from the input image and devising relationships between

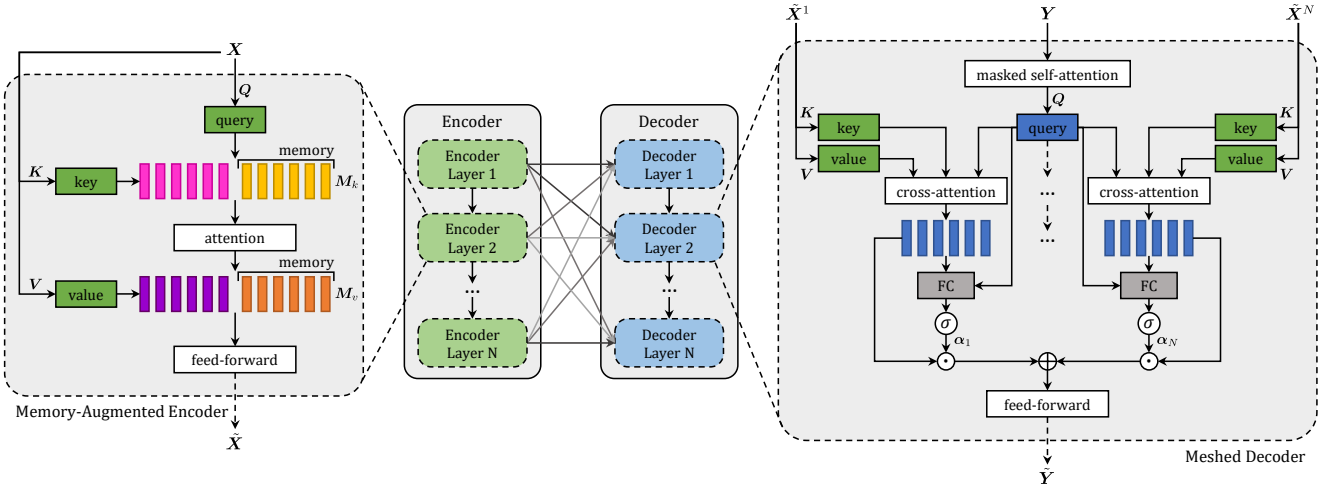


Figure 2: Architecture of the $\mathcal{M}^2$ Transformer. Our model is composed of a stack of memory-augmented encoding layers, which encodes multi-level visual relationships with a priori knowledge, and a stack of decoder layers, in charge of generating textual tokens. For the sake of clarity, AddNorm operations are not shown. Best seen in color.

them, the decoder reads from the output of each encoding layer to generate the output caption word by word. All intra-modality and cross-modality interactions between word and image-level features are modeled via scaled dot-product attention, without using recurrence. Attention operates on three sets of vectors, namely a set of queries $\boldsymbol{Q}$, keys $\boldsymbol{K}$ and values $\boldsymbol{V}$, and takes a weighted sum of value vectors according to a similarity distribution between query and key vectors. In the case of scaled dot-product attention, the operator can be formally defined as

$$\text{Attention}(\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V}) = \text{softmax}\left(\frac{\boldsymbol{Q}\boldsymbol{K}^T}{\sqrt{d}}\right)\boldsymbol{V}, \quad (1)$$

where $\boldsymbol{Q}$ is a matrix of n_q query vectors, $\boldsymbol{K}$ and $\boldsymbol{V}$ both contain n_k keys and values, all with the same dimensionality, and d is a scaling factor.

3.1. Memory-Augmented Encoder

Given a set of image regions $\boldsymbol{X}$ extracted from an input image, attention can be used to obtain a permutation invariant encoding of $\boldsymbol{X}$ through the self-attention operations used in the Transformer [37]. In this case, queries, keys, and values are obtained by linearly projecting the input features, and the operator can be defined as

$$\mathcal{S}(\boldsymbol{X}) = \text{Attention}(W_q\boldsymbol{X}, W_k\boldsymbol{X}, W_v\boldsymbol{X}), \quad (2)$$

where W_q, W_k, W_v are matrices of learnable weights. The output of the self-attention operator is a new set of elements $\mathcal{S}(\boldsymbol{X})$, with the same cardinality as $\boldsymbol{X}$, in which each element of $\boldsymbol{X}$ is replaced with a weighted sum of the values, *i.e.* of linear projections of the input (Eq. 1).

Noticeably, attentive weights depend solely on the pairwise similarities between linear projections of the input set itself. Therefore, the self-attention operator can be seen as a way of encoding pairwise relationships inside the input set. When using image regions (or features derived from image regions) as the input set, $\mathcal{S}(\cdot)$ can naturally encode the pairwise relationships between regions that are needed to understand the input image before describing it[1].

This peculiarity in the definition of self-attention has, however, a significant limitation. Because everything depends solely on pairwise similarities, self-attention cannot model a priori knowledge on relationships between image regions. For example, given one region encoding a man and a region encoding a basketball ball, it would be difficult to infer the concept of *player* or *game* without any a priori knowledge. Again, given regions encoding eggs and toasts, the knowledge that the picture depicts a *breakfast* could be easily inferred using a priori knowledge on relationships.

Memory-Augmented Attention. To overcome this limitation of self-attention, we propose a memory-augmented attention operator. In our proposal, the set of keys and values used for self-attention is extended with additional "slots" which can encode a priori information. To stress that a priori information should not depend on the input set $\boldsymbol{X}$, the additional keys and values are implemented as plain learnable vectors which can be directly updated via SGD. Formally, the operator is defined as:

$$\begin{aligned} \mathcal{M}_{\text{mem}}(\boldsymbol{X}) &= \text{Attention}(W_q\boldsymbol{X}, \boldsymbol{K}, \boldsymbol{V}) \\ \boldsymbol{K} &= [W_k\boldsymbol{X}, \boldsymbol{M}_k] \\ \boldsymbol{V} &= [W_v\boldsymbol{X}, \boldsymbol{M}_v], \end{aligned} \quad (3)$$

where $\boldsymbol{M}_k$ and $\boldsymbol{M}_v$ are learnable matrices with n_m rows, and $[\cdot,\cdot]$ indicates concatenation. Intuitively, by adding

[1] Taking another perspective, self-attention is also conceptually equivalent to an attentive encoding of graph nodes [39].

learnable keys and values, through attention it will be possible to retrieve learned knowledge which is not already embedded in $\boldsymbol{X}$. At the same time, our formulation leaves the set of queries unaltered.

Just like the self-attention operator, our memory-augmented attention can be applied in a multi-head fashion. In this case, the memory-augmented attention operation is repeated h times, using different projection matrices W_q, W_k, W_v and different learnable memory slots $\boldsymbol{M}_k, \boldsymbol{M}_v$ for each head. Then, we concatenate the results from different heads and apply a linear projection.

Encoding layer. We embed our memory-augmented operator into a Transformer-like layer: the output of the memory-augmented attention is applied to a position-wise feed-forward layer composed of two affine transformations with a single non-linearity, which are independently applied to each element of the set. Formally,

$$\mathcal{F}(\boldsymbol{X})_i = U\sigma(V\boldsymbol{X}_i + b) + c, \tag{4}$$

where $\boldsymbol{X}_i$ indicates the i-th vector of the input set, and $\mathcal{F}(\boldsymbol{X})_i$ the i-th vector of the output. Also, $\sigma(\cdot)$ is the ReLU activation function, V and U are learnable weight matrices, b and c are bias terms.

Each of these sub-components (memory-augmented attention and position-wise feed-forward) is then encapsulated within a residual connection and a layer norm operation. The complete definition of an encoding layer can be finally written as:

$$\begin{aligned} \boldsymbol{Z} &= \mathsf{AddNorm}(\mathcal{M}_{\text{mem}}(\boldsymbol{X})) \\ \tilde{\boldsymbol{X}} &= \mathsf{AddNorm}(\mathcal{F}(\boldsymbol{Z})), \end{aligned} \tag{5}$$

where AddNorm indicates the composition of a residual connection and of a layer normalization.

Full encoder. Given the aforementioned structure, multiple encoding layers are stacked in sequence, so that the i-th layer consumes the output set computed by layer $i-1$. This amounts to creating multi-level encodings of the relationships between image regions, in which higher encoding layers can exploit and refine relationships already identified by previous layers, eventually using a priori knowledge. A stack of N encoding layers will therefore produce a multi-level output $\tilde{\mathcal{X}} = (\tilde{\boldsymbol{X}}^1, ..., \tilde{\boldsymbol{X}}^N)$, obtained from the outputs of each encoding layer.

3.2. Meshed Decoder

Our decoder is conditioned on both previously generated words and region encodings, and is in charge of generating the next tokens of the output caption. Here, we exploit the aforementioned multi-level representation of the input image while still building a multi-layer structure. To this aim, we devise a meshed attention operator which, unlike the cross-attention operator of the Transformer, can take advantage of all encoding layers during the generation of the sentence.

Meshed Cross-Attention. Given an input sequence of vectors $\boldsymbol{Y}$, and outputs from all encoding layers $\tilde{\mathcal{X}}$, the Meshed Attention operator connects $\boldsymbol{Y}$ to all elements in $\tilde{\mathcal{X}}$ through gated cross-attentions. Instead of attending only the last encoding layer, we perform a cross-attention with all encoding layers. These multi-level contributions are then summed together after being modulated. Formally, our meshed attention operator is defined as

$$\mathcal{M}_{\text{mesh}}(\tilde{\mathcal{X}}, \boldsymbol{Y}) = \sum_{i=1}^{N} \boldsymbol{\alpha}_i \odot \mathcal{C}(\tilde{\boldsymbol{X}}^i, \boldsymbol{Y}), \tag{6}$$

where $\mathcal{C}(\cdot,\cdot)$ stands for the encoder-decoder cross-attention, computed using queries from the decoder and keys and values from the encoder:

$$\mathcal{C}(\tilde{\boldsymbol{X}}^i, \boldsymbol{Y}) = \mathsf{Attention}(W_q\boldsymbol{Y}, W_k\tilde{\boldsymbol{X}}^i, W_v\tilde{\boldsymbol{X}}^i), \tag{7}$$

and $\boldsymbol{\alpha}_i$ is a matrix of weights having the same size as the cross-attention results. Weights in $\boldsymbol{\alpha}_i$ modulate both the single contribution of each encoding layer, and the relative importance between different layers. These are computed by measuring the relevance between the result of the cross-attention computed with each encoding layer and the input query, as follows:

$$\boldsymbol{\alpha}_i = \sigma\left(W_i\left[\boldsymbol{Y}, \mathcal{C}(\tilde{\boldsymbol{X}}^i, \boldsymbol{Y})\right] + b_i\right), \tag{8}$$

where $[\cdot,\cdot]$ indicates concatenation, σ is the sigmoid activation, W_i is a $2d \times d$ weight matrix, and b_i is a learnable bias vector.

Architecture of decoding layers. As for encoding layers, we apply our meshed attention in a multi-head fashion. As the prediction of a word should only depend on previously predicted words, the decoder layer comprises a masked self-attention operation which connects queries derived from the t-th element of its input sequence $\boldsymbol{Y}$ with keys and values obtained from the left-hand subsequence, *i.e.* $\boldsymbol{Y}_{\leq t}$. Also, the decoder layer contains a position-wise feed-forward layer (as in Eq. 4), and all components are encapsulated within AddNorm operations. The final structure of the decoder layer can be written as:

$$\begin{aligned} \boldsymbol{Z} &= \mathsf{AddNorm}(\mathcal{M}_{\text{mesh}}(\tilde{\mathcal{X}}, \mathsf{AddNorm}(\mathcal{S}_{\text{mask}}(\boldsymbol{Y})))) \\ \tilde{\boldsymbol{Y}} &= \mathsf{AddNorm}(\mathcal{F}(\boldsymbol{Z})), \end{aligned} \tag{9}$$

where $\boldsymbol{Y}$ is the input sequence of vectors and $\mathcal{S}_{\text{mask}}$ indicates a masked self-attention over time. Finally, our decoder stacks together multiple decoder layers, helping to refine both the understanding of the textual input and the generation of next tokens. Overall, the decoder takes as input word

vectors, and the t-th element of its output sequence encodes the prediction of a word at time $t+1$, conditioned on $\boldsymbol{Y}_{\leq t}$. After taking a linear projection and a softmax operation, this encodes a probability over words in the dictionary.

3.3. Training details

Following a standard practice in image captioning [29, 31, 4], we pre-train our model with a word-level cross-entropy loss (XE) and finetune the sequence generation using reinforcement learning. When training with XE, the model is trained to predict the next token given previous ground-truth words; in this case, the input sequence for the decoder is immediately available and the computation of the entire output sequence can be done in a single pass, parallelizing all operations over time.

When training with reinforcement learning, we employ a variant of the self-critical sequence training approach [31] on sequences sampled using beam search [4]: to decode, we sample the top-k words from the decoder probability distribution at each timestep, and always maintain the top-k sequences with highest probability. As sequence decoding is iterative in this step, the aforementioned parallelism over time cannot be exploited. However, intermediate keys and values used to compute the output token at time t can be reused in the next iterations.

Following previous works [4], we use the CIDEr-D score as reward, as it well correlates with human judgment [38]. We baseline the reward using the mean of the rewards rather than greedy decoding as done in previous methods [31, 4], as we found it to slightly improve the final performance. The final gradient expression for one sample is thus:

$$\nabla_\theta L(\theta) = -\frac{1}{k}\sum_{i=1}^{k}\left((r(\boldsymbol{w}^i) - b)\nabla_\theta \log p(\boldsymbol{w}^i)\right) \qquad (10)$$

where $\boldsymbol{w}^i$ is the i-th sentence in the beam, $r(\cdot)$ is the reward function, and $b = \left(\sum_i r(\boldsymbol{w}^i)\right)/k$ is the baseline, computed as the mean of the rewards obtained by the sampled sequences. At prediction time, we decode again using beam search, and keep the sequence with highest predicted probability among those in the last beam.

4. Experiments

4.1. Datasets

We first evaluate our model on the COCO dataset [22], which is the most commonly used test-bed for image captioning. Then, we assess the captioning of novel objects by testing on the recently proposed nocaps dataset [1].

COCO. The dataset contains more than $120\,000$ images, each of them annotated with 5 different captions. We follow the splits provided by Karpathy *et al*. [15], where $5\,000$ images are used for validation, $5\,000$ for testing and the rest for training. We also evaluate the model on the COCO online test server, composed of $40\,775$ images for which annotations are not made publicly available.

nocaps. The dataset consists of $15\,100$ images taken from the Open Images [19] validation and test sets, each annotated with 11 human-generated captions. Images are divided into validation and test splits, respectively composed of $4\,500$ and $10\,600$ elements. Images can be further grouped into three subsets depending on the nearness to COCO, namely in-domain, near-domain, and out-of-domain images. Under this setting, we use COCO as training data and evaluate our results on the nocaps test server.

4.2. Experimental settings

Metrics. Following the standard evaluation protocol, we employ the full set of captioning metrics: BLEU [26], METEOR [6], ROUGE [21], CIDEr [38], and SPICE [2].

Implementation details. To represent image regions, we use Faster R-CNN [30] with ResNet-101 [10] finetuned on the Visual Genome dataset [18, 4], thus obtaining a 2048-dimensional feature vector for each region. To represent words, we use one-hot vectors and linearly project them to the input dimensionality of the model d. We also employ sinusoidal positional encodings [37] to represent word positions inside the sequence and sum the two embeddings before the first decoding layer.

In our model, we set the dimensionality d of each layer to 512, the number of heads to 8, and the number of memory vectors to 40. We employ dropout with keep probability 0.9 after each attention and feed-forward layer. In our meshed attention operator (Eq. 6), we normalize the output with a scaling factor of $\sqrt{N}$. Pre-training with XE is done following the learning rate scheduling strategy of [37] with a warmup equal to $10\,000$ iterations. Then, during CIDEr-D optimization, we use a fixed learning rate of 5×10^{-6}. We train all models using the Adam optimizer [17], a batch size of 50, and a beam size equal to 5.

Novel object captioning. To train the model on the nocaps dataset, instead of using one-hot vectors, we represent words with GloVe word embeddings [28]. Two fully-connected layers are added to convert between the GloVe dimensionality and d before the first decoding layer and after the last decoding layer. Before the final softmax, we multiply with the transpose of the word embeddings. All other implementation details are kept unchanged.

Additional details on model architecture and training can be found in the supplementary material.

4.3. Ablation study

Performance of the Transformer. In previous works, the Transformer model has been applied to captioning only in its original configuration with six layers, with the structure of connections that has been successful for uni-modal sce-

	B-1	B-4	M	R	C	S
Transformer (w/ 6 layers as in [37])	79.1	36.2	27.7	56.9	121.8	20.9
Transformer (w/ 3 layers)	79.6	36.5	27.8	57.0	123.6	21.1
Transformer (w/ AoA [12])	80.3	38.8	29.0	58.4	129.1	**22.7**
$\mathcal{M}^2$ Transformer$^{\text{1-to-1}}$ (w/o mem.)	80.5	38.2	28.9	58.2	128.4	22.2
$\mathcal{M}^2$ Transformer$^{\text{1-to-1}}$	80.3	38.2	28.9	58.2	129.2	22.5
$\mathcal{M}^2$ Transformer (w/o mem.)	80.4	38.3	29.0	58.2	129.4	22.6
$\mathcal{M}^2$ Transformer (w/ softmax)	80.3	38.4	29.1	58.3	130.3	22.5
$\mathcal{M}^2$ **Transformer**	**80.8**	**39.1**	**29.2**	**58.6**	**131.2**	22.6

Table 1: Ablation study and comparison with Transformer-based alternatives. All results are reported after the REINFORCE optimization stage.

narios like machine translation. As we speculate that captioning requires specific architectures, we compare variations of the original Transformer with our approach.

Firstly, we investigate the impact of the number of encoding and decoding layers on captioning performance. As it can be seen in Table 1, the original Transformer (six layers) achieves 121.8 CIDEr, slightly superior to the Up-Down approach [4] which uses a two-layer recurrent language model with additive attention and includes a global feature vector (120.1 CIDEr). Varying the number of layers, we observe a significant increase in performance when using three encoding and three decoding layers, which leads to 123.6 CIDEr. We hypothesize that this is due to the reduced training set size and to the lower semantic complexities of sentences in captioning with respect to those of language understanding tasks. Following this finding, all subsequent experiments will use three layers.

Attention on Attention baseline. We also evaluate a recent proposal that can be straightforwardly applied to the Transformer as an alternative to standard dot-product attention. Specifically, we evaluate the addition of the "Attention on Attention" (AoA) approach [12] to the attentive layers, both in the encoder and in the decoder. Noticeably, in [12] this has been done with a Recurrent language model with attention, but the approach is sufficiently general to be applied to any attention stage. In this case, the result of dot-product attention is concatenated with the initial query and fed to two fully connected layers to obtain an information vector and a sigmoidal attention gate, then the two vectors are multiplied together. The final result is used as an alternative to the standard dot-product attention. This addition to a standard Transformer with three layers leads to 129.1 CIDEr (Table 1), thus underlying the usefulness of the approach also in Transformer-based models.

Meshed Connectivity. We then evaluate the role of the meshed connections between encoder and decoder layers. In Table 1, we firstly introduce a reduced version of our approach in which the i-th decoder layer is only connected to the corresponding i-th encoder layer (1-to-1), instead of being connected to all encoders. Using this 1-to-1 connectivity schema already brings an improvement with respect to using the output of the last encoder layer as in the standard Transformer (123.6 CIDEr vs 129.2 CIDEr), thus confirming that exploiting a multi-level encoding of image regions is beneficial. When we instead use our meshed connectivity schema, that exploits relationships encoded at all levels and weights them with a sigmoid gating, we observe a further performance improvement, from 129.2 CIDEr to 131.2 CIDEr. This amounts to a total improvement of 7.6 points with respect to the standard Transformer. Also, the result of our full model is superior to that obtained using the AoA.

As an alternative to the sigmoid gating approach for weighting the contributions from different encoder layers (Eq. 6), we also test with a softmax gating schema. In this case, the element-wise sigmoid applied to each encoder is replaced with a softmax operation over the rows of $\boldsymbol{\alpha}_i$. Using this alternative brings to a reduction of around 1 CIDEr point, underlying that it is beneficial to exploit the full potentiality of a weighted sum of the contributions from all encoding layers, rather than forcing a peaky distribution in which one layer is given more importance than the others.

Role of persistent memory. We evaluate the role of memory vectors in both the 1-to-1 configuration and in the fi-

	B-1	B-4	M	R	C	S
SCST [31]	-	34.2	26.7	55.7	114.0	-
Up-Down [4]	79.8	36.3	27.7	56.9	120.1	21.4
RFNet [13]	79.1	36.5	27.7	57.3	121.9	21.2
Up-Down+HIP [47]	-	38.2	28.4	58.3	127.2	21.9
GCN-LSTM [46]	80.5	38.2	28.5	58.3	127.6	22.0
SGAE [44]	**80.8**	38.4	28.4	58.6	127.8	22.1
ORT [11]	80.5	38.6	28.7	58.4	128.3	**22.6**
AoANet [12]	80.2	38.9	**29.2**	**58.8**	129.8	22.4
$\mathcal{M}^2$ **Transformer**	**80.8**	**39.1**	**29.2**	58.6	**131.2**	**22.6**

Table 2: Comparison with the state of the art on the "Karpathy" test split, in single-model setting.

	B-1	B-4	M	R	C	S
Ensemble/Fusion of 2 models						
GCN-LSTM [46]	80.9	38.3	28.6	58.5	128.7	22.1
SGAE [44]	81.0	39.0	28.4	58.9	129.1	22.2
ETA [20]	81.5	**39.9**	28.9	59.0	127.6	22.6
GCN-LSTM+HIP [47]	-	39.1	28.9	**59.2**	130.6	22.3
$\mathcal{M}^2$ **Transformer**	**81.6**	39.8	**29.5**	**59.2**	**133.2**	**23.1**
Ensemble/Fusion of 4 models						
SCST [31]	-	35.4	27.1	56.6	117.5	-
RFNet [13]	80.4	37.9	28.3	58.3	125.7	21.7
AoANet [12]	81.6	40.2	29.3	59.4	132.0	22.8
$\mathcal{M}^2$ **Transformer**	**82.0**	**40.5**	**29.7**	**59.5**	**134.5**	**23.5**

Table 3: Comparison with the state of the art on the "Karpathy" test split, using an ensemble of models.

	BLEU-1		BLEU-2		BLEU-3		BLEU-4		METEOR		ROUGE		CIDEr	
	c5	c40	c5	c40	c5	c40	c5	c40	c5	c40	c5	c40	c5	c40
SCST [31]	78.1	93.7	61.9	86.0	47.0	75.9	35.2	64.5	27.0	35.5	56.3	70.7	114.7	116.7
Up-Down [4]	80.2	95.2	64.1	88.8	49.1	79.4	36.9	68.5	27.6	36.7	57.1	72.4	117.9	120.5
RDN [16]	80.2	95.3	-	-	-	-	37.3	69.5	28.1	37.8	57.4	73.3	121.2	125.2
RFNet [13]	80.4	95.0	64.9	89.3	50.1	80.1	38.0	69.2	28.2	37.2	58.2	73.1	122.9	125.1
GCN-LSTM [46]	80.8	95.9	65.5	89.3	50.8	80.3	38.7	69.7	28.5	37.6	58.5	73.4	125.3	126.5
SGAE [44]	81.0	95.3	65.6	89.5	50.7	80.4	38.5	69.7	28.2	37.2	58.6	73.6	123.8	126.5
ETA [20]	81.2	95.0	65.5	89.0	50.9	80.4	38.9	70.2	28.6	38.0	58.6	73.9	122.1	124.4
AoANet [12]	81.0	95.0	65.8	89.6	51.4	81.3	39.4	71.2	29.1	38.5	58.9	74.5	126.9	129.6
GCN-LSTM+HIP [47]	**81.6**	95.9	66.2	90.4	51.5	81.6	39.3	71.0	28.8	38.1	59.0	74.1	127.9	130.2
$\mathcal{M}^2$ **Transformer**	**81.6**	**96.0**	**66.4**	**90.8**	**51.8**	**82.7**	**39.7**	**72.8**	**29.4**	**39.0**	**59.2**	**74.8**	**129.3**	**132.1**

Table 4: Leaderboard of various methods on the online MS-COCO test server.

nal configuration with meshed connections. As it can be seen from Table 1, removing memory vectors brings to a reduction in performance of around 1 CIDEr point in both connectivity settings, thus confirming the usefulness of exploiting a priori learned knowledge when encoding image regions. Further experiments on the number of memory vectors can be found in the supplementary material.

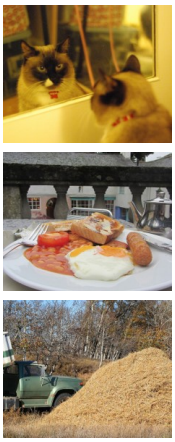
4.4. Comparison with state of the art

We compare the performances of our approach with those of several recent proposals for image captioning. The models we compare to include SCST [31] and Up-Down [4], which respectively use attention over the grid of features and attention over regions. Also, we compare to RFNet [13], which uses a recurrent fusion network to merge different CNN features; GCN-LSTM [46], which exploits pairwise relationships between image regions through a Graph CNN; SGAE [44], which instead uses auto-encoding scene graphs. Further, we compare with the original AoANet [12] approach, which uses attention on attention for encoding image regions and an LSTM language model. Finally, we compare with ORT [11], which uses a plain Transformer and weights attention scores in the region encoder with pairwise distances between detections.

We evaluate our approach on the COCO "Karpathy" test split, using both single model and ensemble configurations, and on the online COCO evaluation server.

Single model. In Table 2 we report the performance of our method in comparison with the aforementioned competitors, using captions predicted from a single model and optimization on the CIDEr-D score. As it can be observed, our method surpasses all other approaches in terms of BLEU-4, METEOR and CIDEr, while being competitive on BLEU-1 and SPICE with the best performer, and slightly worse on ROUGE with respect to AoANet [12]. In particular, it advances the current state of the art on CIDEr by 1.4 points.

Ensemble model. Following the common practice [31, 12] of building an ensemble of models, we also report the performances of our approach when averaging the output probability distributions of multiple and independently trained instances of our model. In Table 3, we use ensembles of two and four models, trained from different random seeds. Noticeably, when using four models our approach achieves the best performance according to all metrics, with an increase of 2.5 CIDEr points with respect to the current state of the art [12].



GT: A cat looking at his reflection in the mirror.
Transformer: A cat sitting in a window sill looking out.
$\mathcal{M}^2$ **Transformer:** A cat looking at its reflection in a mirror.

GT: A plate of food including eggs and toast on a table next to a stone railing.
Transformer: A group of food on a plate.
$\mathcal{M}^2$ **Transformer:** A plate of breakfast food with eggs and toast.

GT: A truck parked near a tall pile of hay.
Transformer: A truck is parked in the grass in a field.
$\mathcal{M}^2$ **Transformer:** A green truck parked next to a pile of hay.

Figure 3: Examples of captions generated by our approach and the original Transformer model, as well as the corresponding ground-truths.

Online Evaluation. Finally, we also report the performance of our method on the online COCO test server[2]. In this case, we use the ensemble of four models previously described, trained on the "Karpathy" training split. The evaluation is done on the COCO test split, for which ground-truth annotations are not publicly available. Results are reported in Table 4, in comparison with the top-performing approaches of the leaderboard. For fairness of comparison, they also used an ensemble configuration. As it can be seen, our method surpasses the current state of the art on all metrics, achieving an advancement of 1.4 CIDEr points with respect to the best performer.

[2] https://competitions.codalab.org/competitions/3221

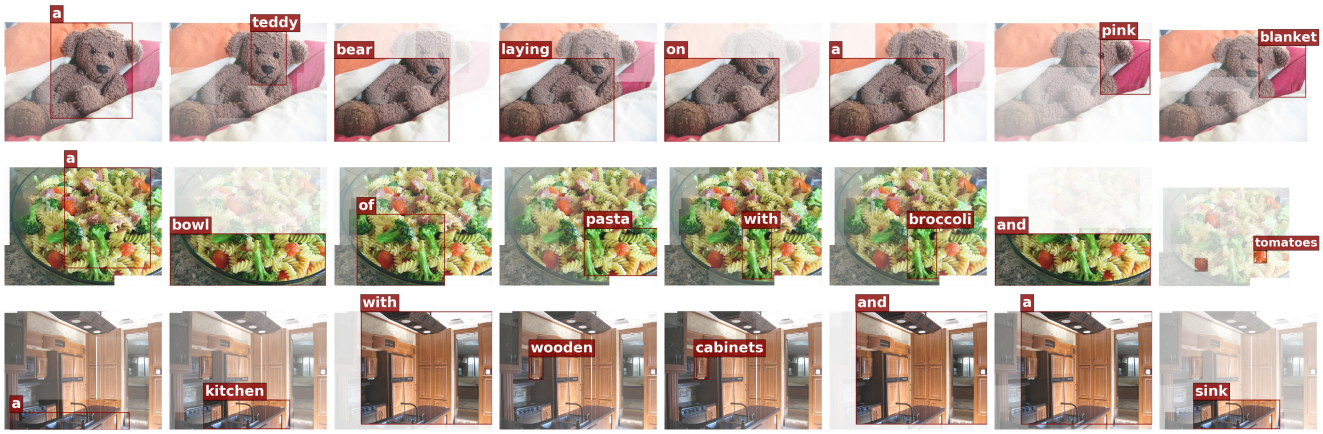


Figure 4: Visualization of attention states for three sample captions. For each generated word, we show the attended image regions, outlining the region with the maximum output attribution in red.

	In-Domain		Out-of-Domain		Overall	
	CIDEr	SPICE	CIDEr	SPICE	CIDEr	SPICE
NBT + CBS [1]	62.1	10.1	62.4	8.9	60.2	9.5
Up-Down + CBS [1]	80.0	12.0	66.4	9.7	73.1	11.1
Transformer	78.0	11.0	29.7	7.8	54.7	9.8
$\mathcal{M}^2$ **Transformer**	**85.7**	**12.1**	38.9	8.9	64.5	11.1
Transformer + CBS	74.3	11.0	62.5	9.2	66.9	10.3
$\mathcal{M}^2$ **Transformer + CBS**	81.2	12.0	**69.4**	**10.0**	**75.0**	**11.4**

Table 5: Performances on nocaps validation set, for in-domain and out-of-domain captioning.

4.5. Describing novel objects

We also assess the performance of our approach when dealing with images containing object categories that are not seen in the training set. We compare with Up-Down [4] and Neural Baby Talk [25], when using GloVe word embeddings and Constrained Beam Search (CBS) [3] to address the generation of out-of-vocabulary words and constrain the presence of categories detected by an object detector. To compare with our model, we use a simplified implementation of the procedure described in [1] to extract constraints, without using word phrases. Results are shown in Table 5: as it can be seen, the original Transformer is significantly less performing than Up-Down on both in-domain and out-of-domain categories, while our approach can properly deal with novel categories, surpassing the Up-Down baseline in both in-domain and out-of-domain images. As expected, the use of CBS significantly enhances the performances, in particular on out-of-domain captioning.

4.6. Qualitative results and visualization

Figure 3 proposes qualitative results generated by our model and the original Transformer. On average, our model is able to generate more accurate and descriptive captions, integrating fine-grained details and object relations.

Finally, to better understand the effectiveness of our $\mathcal{M}^2$ Transformer, we investigate the contribution of detected regions to the model output. Differently from recurrent-based captioning models, in which attention weights over regions can be easily extracted, in our model the contribution of one region with respect to the output is given by more complex non-linear dependencies. Therefore, we revert to attribution methods: specifically, we employ the Integrated Gradients approach [36], which approximates the integral of gradients with respect to the given input. Results are presented in Figure 4, where we observe that our approach correctly grounds image regions to words, also in presence of object details and small detections. More visualizations are included in the supplementary material.

5. Conclusion

We presented $\mathcal{M}^2$ Transformer, a novel Transformer-based architecture for image captioning. Our model incorporates a region encoding approach that exploits a priori knowledge through memory vectors and a meshed connectivity between encoding and decoding modules. Noticeably, this connectivity pattern is unprecedented for other fully-attentive architectures. Experimental results demonstrated that our approach achieves a new state of the art on COCO, ranking first in the on-line leaderboard. Finally, we validated the components of our model through ablation studies, and its performances when describing novel objects.

Acknowledgment

This work was partially supported by the "IDEHA - Innovation for Data Elaboration in Heritage Areas" project (PON ARS01_00421), funded by the Italian Ministry of Education (MIUR). We also acknowledge the NVIDIA AI Technology Center, EMEA, for its support and access to computing resources.

References

[1] Harsh Agrawal, Karan Desai, Xinlei Chen, Rishabh Jain, Dhruv Batra, Devi Parikh, Stefan Lee, and Peter Anderson. nocaps: novel object captioning at scale. In *Proceedings of the International Conference on Computer Vision*, 2019.

[2] Peter Anderson, Basura Fernando, Mark Johnson, and Stephen Gould. SPICE: Semantic Propositional Image Caption Evaluation. In *Proceedings of the European Conference on Computer Vision*, 2016.

[3] Peter Anderson, Basura Fernando, Mark Johnson, and Stephen Gould. Guided open vocabulary image captioning with constrained beam search. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 2017.

[4] Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang. 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*, 2018.

[5] Jyoti Aneja, Aditya Deshpande, and Alexander G Schwing. Convolutional image captioning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[6] Satanjeev Banerjee and Alon Lavie. 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*, 2005.

[7] Marcella Cornia, Lorenzo Baraldi, and Rita Cucchiara. Show, Control and Tell: A Framework for Generating Controllable and Grounded Captions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019.

[8] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[9] Jeffrey Donahue, Lisa Anne Hendricks, Sergio Guadarrama, Marcus Rohrbach, Subhashini Venugopalan, Kate Saenko, and Trevor Darrell. Long-term recurrent convolutional networks for visual recognition and description. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015.

[10] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

[11] Simao Herdade, Armin Kappeler, Kofi Boakye, and Joao Soares. Image Captioning: Transforming Objects into Words. In *Advances in Neural Information Processing Systems*, 2019.

[12] Lun Huang, Wenmin Wang, Jie Chen, and Xiao-Yong Wei. Attention on Attention for Image Captioning. In *Proceedings of the International Conference on Computer Vision*, 2019.

[13] Wenhao Jiang, Lin Ma, Yu-Gang Jiang, Wei Liu, and Tong Zhang. Recurrent Fusion Network for Image Captioning. In *Proceedings of the European Conference on Computer Vision*, 2018.

[14] Justin Johnson, Andrej Karpathy, and Li Fei-Fei. DenseCap: Fully convolutional Localization Networks for Dense Captioning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

[15] Andrej Karpathy and Li Fei-Fei. Deep visual-semantic alignments for generating image descriptions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015.

[16] Lei Ke, Wenjie Pei, Ruiyu Li, Xiaoyong Shen, and Yu-Wing Tai. Reflective Decoding Network for Image Captioning. In *Proceedings of the International Conference on Computer Vision*, 2019.

[17] Diederik P Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization. In *Proceedings of the International Conference on Learning Representations*, 2015.

[18] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, Michael Bernstein, and Li Fei-Fei. Visual Genome: Connecting Language and Vision Using Crowdsourced Dense Image Annotations. *International Journal of Computer Vision*, 123(1):32–73, 2017.

[19] Alina Kuznetsova, Hassan Rom, Neil Alldrin, Jasper Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Malloci, Tom Duerig, et al. The Open Images Dataset V4: Unified image classification, object detection, and visual relationship detection at scale. *arXiv preprint arXiv:1811.00982*, 2018.

[20] Guang Li, Linchao Zhu, Ping Liu, and Yi Yang. Entangled Transformer for Image Captioning. In *Proceedings of the International Conference on Computer Vision*, 2019.

[21] Chin-Yew Lin. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out: Proceedings of the ACL Workshop*, volume 8, 2004.

[22] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft COCO: Common Objects in Context. In *Proceedings of the European Conference on Computer Vision*, 2014.

[23] Siqi Liu, Zhenhai Zhu, Ning Ye, Sergio Guadarrama, and Kevin Murphy. Improved Image Captioning via Policy Gradient Optimization of SPIDEr. In *Proceedings of the International Conference on Computer Vision*, 2017.

[24] Jiasen Lu, Caiming Xiong, Devi Parikh, and Richard Socher. 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*, 2017.

[25] Jiasen Lu, Jianwei Yang, Dhruv Batra, and Devi Parikh. Neural Baby Talk. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[26] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, 2002.

[27] Marco Pedersoli, Thomas Lucas, Cordelia Schmid, and Jakob Verbeek. Areas of attention for image captioning. In

Proceedings of the International Conference on Computer Vision, 2017.

[28] Jeffrey Pennington, Richard Socher, and Christopher Manning. GloVe: Global vectors for word representation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 2014.

[29] Marc'Aurelio Ranzato, Sumit Chopra, Michael Auli, and Wojciech Zaremba. Sequence level training with recurrent neural networks. In *Proceedings of the International Conference on Learning Representations*, 2015.

[30] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. In *Advances in Neural Information Processing Systems*, 2015.

[31] Steven J Rennie, Etienne Marcheret, Youssef Mroueh, Jarret Ross, and Vaibhava Goel. Self-critical sequence training for image captioning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[32] Richard Socher and Li Fei-Fei. Connecting modalities: Semi-supervised segmentation and annotation of images using unaligned text corpora. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2010.

[33] Yale Song and Mohammad Soleymani. Polysemous visual-semantic embedding for cross-modal retrieval. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019.

[34] Sainbayar Sukhbaatar, Edouard Grave, Guillaume Lample, Herve Jegou, and Armand Joulin. Augmenting Self-attention with Persistent Memory. *arXiv preprint arXiv:1907.01470*, 2019.

[35] Chen Sun, Austin Myers, Carl Vondrick, Kevin Murphy, and Cordelia Schmid. Videobert: A joint model for video and language representation learning. In *Proceedings of the International Conference on Computer Vision*, 2019.

[36] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In *Proceedings of the International Conference on Machine Learning*, 2017.

[37] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, 2017.

[38] Ramakrishna Vedantam, C Lawrence Zitnick, and Devi Parikh. CIDEr: Consensus-based Image Description Evaluation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015.

[39] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. Graph Attention Networks. In *Proceedings of the International Conference on Learning Representations*, 2018.

[40] Oriol Vinyals, Alexander Toshev, Samy Bengio, and Dumitru Erhan. Show and tell: A neural image caption generator. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015.

[41] Oriol Vinyals, Alexander Toshev, Samy Bengio, and Dumitru Erhan. Show and tell: Lessons learned from the 2015 mscoco image captioning challenge. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(4):652–663, 2016.

[42] Oriol Vinyals, Alexander Toshev, Samy Bengio, and Dumitru Erhan. Show and Tell: Lessons Learned from the 2015 MSCOCO Image Captioning Challenge. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(4):652–663, 2017.

[43] Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhutdinov, Richard S Zemel, and Yoshua Bengio. Show, attend and tell: Neural image caption generation with visual attention. In *Proceedings of the International Conference on Machine Learning*, 2015.

[44] Xu Yang, Kaihua Tang, Hanwang Zhang, and Jianfei Cai. Auto-Encoding Scene Graphs for Image Captioning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019.

[45] Benjamin Z Yao, Xiong Yang, Liang Lin, Mun Wai Lee, and Song-Chun Zhu. I2t: Image parsing to text description. *Proceedings of the IEEE*, 98(8):1485–1508, 2010.

[46] Ting Yao, Yingwei Pan, Yehao Li, and Tao Mei. Exploring Visual Relationship for Image Captioning. In *Proceedings of the European Conference on Computer Vision*, 2018.

[47] Ting Yao, Yingwei Pan, Yehao Li, and Tao Mei. Hierarchy Parsing for Image Captioning. In *Proceedings of the International Conference on Computer Vision*, 2019.

[48] Quanzeng You, Hailin Jin, Zhaowen Wang, Chen Fang, and Jiebo Luo. Image captioning with semantic attention. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016.