This is the peer reviewd version of the followng article:

CaMEL: Mean Teacher Learning for Image Captioning / Barraco, Manuele; Stefanini, Matteo; Cornia, Marcella; Cascianelli, Silvia; Baraldi, Lorenzo; Cucchiara, Rita. - 2022-:(2022), pp. 4087-4094. (Intervento presentato al convegno 26th International Conference on Pattern Recognition, ICPR 2022 tenutosi a Montréal Québec nel August 21-25, 2022) [10.1109/ICPR56361.2022.9955644].

Institute of Electrical and Electronics Engineers Inc.
Terms of use:

The terms and conditions for the reuse of this version of the manuscript are specified in the publishing policy. For all terms of use and more information see the publisher's website.

15/05/2024 05:15

(Article begins on next page)

CaMEL: Mean Teacher Learning for Image Captioning

Manuele Barraco, Matteo Stefanini, Marcella Cornia, Silvia Cascianelli, Lorenzo Baraldi, Rita Cucchiara
University of Modena and Reggio Emilia
Email: {name.surname}@unimore.it

***Abstract*—Describing images in natural language is a fundamental step towards the automatic modeling of connections between the visual and textual modalities. In this paper we present CaMEL, a novel Transformer-based architecture for image captioning. Our proposed approach leverages the interaction of two interconnected language models that learn from each other during the training phase. The interplay between the two language models follows a mean teacher learning paradigm with knowledge distillation. Experimentally, we assess the effectiveness of the proposed solution on the COCO dataset and in conjunction with different visual feature extractors. When comparing with existing proposals, we demonstrate that our model provides state-of-the-art caption quality with a significantly reduced number of parameters. According to the CIDEr metric, we obtain a new state of the art on COCO when training without using external data. The source code and trained models will be made publicly available at: https://github.com/aimagelab/camel.**

I. Introduction

Image captioning has received a relevant interest in the last few years, as describing images in natural language is a fundamental step to model the interconnections between the visual and textual modalities [1]. Early works on this research line were based on the usage of convolutional neural networks [2], [3], [4], [5] – usually pre-trained to solve classification tasks – or object detectors [6], [7], [8] as visual feature extractors, and recurrent neural networks as auto-regressive language models [2], [3], [6], [9], [10]. Nowadays, captioning approaches have significantly evolved towards the usage of Transformer-based language models [11], [12], [13], [14], [15], while the visual feature extraction stage is rapidly evolving towards the use of grid-like features extracted with multi-modal architectures trained on large-scale data with language supervision [16], [17], [18]. In parallel, while many captioning approaches have been trained on middle-size datasets like COCO, the literature is now investigating the usage of large-scale noisy datasets as well [17], [19], [20], [21], [22].

Regardless of these architectural and structural improvements, the training methodology has remained almost unaltered. Indeed, most of the existing approaches for image captioning are based on the usage of a single language model, trained to reproduce the ground-truth caption through a cross-entropy loss and later, in a fine-tuning stage, through the REINFORCE algorithm [9], [23]. In this paper, we take a different path and investigate the development of a training strategy that is based on the interplay between two distinct language models. In particular, we draw inspiration from the Mean Teacher Learning approach [24] – which has been successfully employed to learn visual representation in a self-supervised manner [25] – and propose a schema in which two language models learn and interact together at training time. One of the two language models is employed as teacher, while the other is employed as a student in a knowledge distillation relationship [26], [27], [28]. Parameters update, on the teacher model, is carried out by averaging successive states of the student, through an exponential moving average. In this way, the teacher slowly "follows" the student state through time. We devise and compare strategies to apply this interplay paradigm in both the cross-entropy training stage and during the fine-tuning with reinforcement learning.

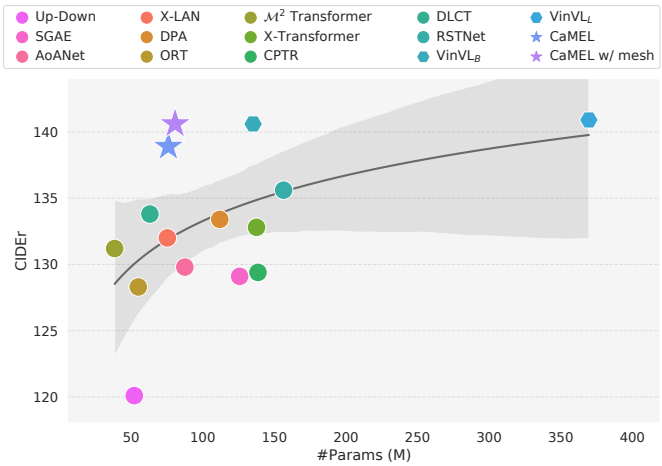

Fig. 1. Comparison of two different versions of our approach (marked with stars) and existing approaches (marked with bullets, and hexagons if exploiting vision-and-language pre-training) in terms of number of parameters and caption quality. Our method features state-of-the-art caption quality in terms of CIDEr with a significantly reduced number of parameters.

Noticeably, at test time one of the two language models can be discarded, so that the number of parameters is kept on pair with traditional models that employ a single language model at training time. We name our model CaMEL – short for Captioner with Mean tEacher Learning. As shown in Fig. 1, our model outperforms existing approaches in terms of caption quality, while being significantly less demanding in terms of number of parameters. We assess the performances of the proposed training strategy on the COCO dataset [29], employing different knowledge distillation strategies, and in comparison with other state-of-the-art approaches that have been trained on the same dataset. We also compare our results

on the COCO online test server. Results demonstrate the goodness of the proposed solution, which attains a new state of the art on COCO when training without using external data.

II. Related Work

Early deep learning-based image captioning approaches entail encoding the visual information by using a CNN-based encoder, and then generating the textual output by using RNN-based language models conditioned on the encoding [2], [3], [9], [30].

Visual encoding. To better capture spatial information and structure, the encoding component has been modified to introduce attention mechanisms, which rely on grids of CNN features in early works [4], [31], [32], and on image regions containing visual entities in following works [6], [33]. An alternative strategy to model spatial and semantic relations between the objects in the image is employing graph convolution neural networks [34], [35], [36]. Finally, a more recent trend entails applying Transformer-like architectures as visual encoders [15], [37], being those also applicable to image patches directly [38], [39], [40]. In our case, we follow the recent paradigm of employing features extracted from large-scale multi-modal architectures [16], [17] like CLIP [18].

Language model. Despite RNN-based language models have been the standard strategy for generating the caption, convolutional language models [41] and fully-attentive language models [14], [42], [43], [44], [45] based on the Transformer paradigm [46] have been explored for image captioning, also motivated by the success of these approaches on Natural Language Processing tasks such as machine translation and language understandings [46], [47], [48]. Moreover, the introduction of Transformer-based language models has brought to the development of effective variants or modifications of the self-attention operator [7], [11], [12], [13], [49], [50], [8] and has enabled vision-and-language early-fusion [19], [22], [51], based on BERT-like architectures [47].

Training strategies. The training strategy for image captioning architectures usually follows the time-wise cross-entropy paradigm. This was later combined with a fine-tuning phase based on the application of the REINFORCE algorithm, to allow using as optimization objectives captioning metrics directly [9], [23], overcoming the issue of their non-differentiability and boosting the final performance. As a strategy to improve both training phases, in [52] it is proposed to exploit a teacher model trained on image attributes to generate additional supervision signals for the captioning model. These are in the form of soft-labels, which the captioning model has to align with in the cross-entropy phase, and re-weighting of the caption words to guide the fine-tuning phase. Additional improvement to the performance of recent self-attention-based image captioning approaches is due to the use of large-scale vision-and-language pre-training [17], [19], [20], [22], [51], which can be done on noisy image-text pairs, also exploiting pre-training losses different from cross-entropy, such as the masked token loss [19], [20]. Different from previous methods, our approach is based on the interplay of two different language models that are trained with the mean teacher learning paradigm and knowledge distillation, without relying on large-scale pre-training.

III. Approach

A. Preliminaries

Most captioning approaches rely on a single language model, which is conditioned on input images and is trained to reproduce ground-truth sentences. Formally, given a dataset of image-caption pairs $\mathcal{D} = \{(\boldsymbol{v}_i, \boldsymbol{t}_i)\}_i$, the language model aims at learning the probability distribution of the next word in a sequence, conditioned on the input image, *i.e.*

$$p(\boldsymbol{w}_\tau | \boldsymbol{w}_{k<\tau}, \boldsymbol{v}), \tag{1}$$

where $\boldsymbol{v}$ is an input image, τ indicates time, and $\{\boldsymbol{w}_\tau\}_\tau$ is the sequence of words comprising the generated caption. The model is trained according to a time-wise cross-entropy (XE) loss over the entire dataset, as follows:

$$\mathcal{L}(\theta) = -\mathbb{E}_{\boldsymbol{x}\sim\mathcal{D}} \sum_\tau \log p(\boldsymbol{w}_\tau | \boldsymbol{w}_{k<\tau}, \boldsymbol{v}, \theta), \tag{2}$$

where θ indicates the set of parameters of the model.

After a training stage with cross-entropy, sequence generation is usually fine-tuned using reinforcement learning. When training with XE, indeed, the model is trained to predict the next token given previous ground-truth words; in reinforcement learning the model is asked to generate an entire sequence and receives a reward that is proportional to the similarity of the generated caption with respect to the ground-truth. A standard practice is to employ a variant of the self-critical sequence training (SCST) approach [9] on sequences sampled using beam search [6]: to decode, the top-k words are sampled from the language model probability distribution at each timestep, and a beam with the top-k sequences with the highest probability is maintained during the generation.

Following previous works [6], the usual practice is to use the CIDEr-D score as reward, as it well correlates with human judgment [53]. In our case, we baseline the reward using the mean of the rewards [13]. The final gradient expression for the SCST training is, therefore

$$\nabla_\theta \mathcal{L}(\theta) = -\frac{1}{k} \sum_{i=1}^{k} \left((r(\boldsymbol{w}^i) - b) \nabla_\theta \log p(\boldsymbol{w}^i) \right), \tag{3}$$

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.

B. CaMEL 🐫

In CaMEL, instead of training a single language model, we rely on the interplay of two different language models – an *online* and a *target* language model, that interact and learn from each other during the training phase, both during the XE pre-training and during the SCST fine-tuning. At test time, each of the two language models can be used, alone, for captioning input images.

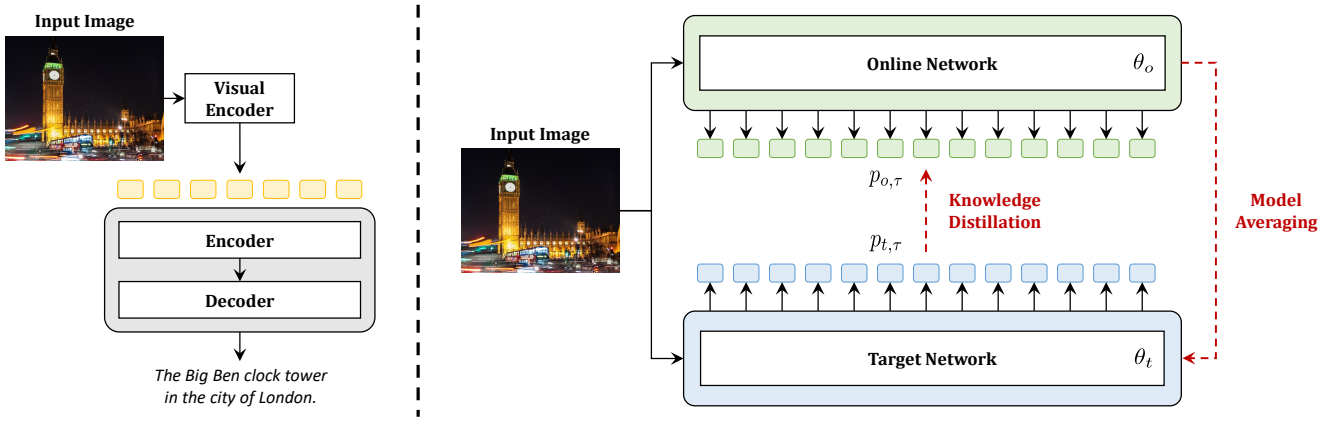

Fig. 2. Overview of our approach and of the interplay between the online and target language models.

The interaction between the online and target language models at training time is two-fold. The online language model is trained, either via XE or SCST, with respect to ground-truth captions. In addition, it performs knowledge distillation with the target language model. The target language model, in turn, updates its weights according to an exponential moving average of the online model weights. An overview of our approach is given in Fig. 2.

Knowledge distillation. The target network provides regression targets to train the online network. This is done through knowledge distillation – treating the online language model as a student network and the target model as a teacher.

Given a visual input $\boldsymbol{v}$ and a conditioning partial sentence $\boldsymbol{w}$, at each timestep τ both networks provide output logits over a vocabulary of N tokens, denoted as $p_{t,\tau}$ and $p_{o,\tau}$ for the target model and the online model, respectively. Given the teacher, which is kept fixed, we learn to match these distributions by minimizing a mean squared error loss with respect to the parameters of the online network, *i.e.*

$$\min_{\theta_o} \sum_{\tau} (p_{t,\tau} - p_{o,\tau})^2, \tag{4}$$

where θ_o indicates the set of parameters of the online network.

Model averaging. The parameters of the target language model are updated as an exponential moving average [25] of the parameters of the online network θ_o. Formally, the parameters of the target model are given by

$$\theta_t \leftarrow \lambda\theta_t + (1-\lambda)\theta_o, \tag{5}$$

where θ_t indicates the set of parameters of the target network and $\lambda \in [0, 1]$ is a target decay rate. In practice, we keep λ fixed during the entire training process. This strategy results in the target network keeping a weighted average of successive states from the online network, thus performing a form of model ensembling [24].

Objective. During XE pre-training, the final objective we employ to train the online network is a combination of the standard XE loss, which is computed with respect to ground-truth captions, and of the knowledge distillation loss with respect to the target logits:

$$\mathcal{L} = \mathcal{L}_{XE} + \lambda_{KD} \cdot \mathcal{L}_{KD}, \tag{6}$$

where λ_{KD} is a weighting hyperparameter. After each SGD update of the online network, the target network is updated through the model averaging. Algorithm 1 provides the PyTorch pseudo-code of the training loop during the XE stage.

Algorithm 1 CaMEL PyTorch pseudocode

```
# gt, go: target and online networks
# l_KD: knowledge distillation weight
# l: network momentum
# v, c: training (image, caption) pair
for v, c in dataloader:
    t = gt(v, c)                          # target output
    o = go(v, c)                          # online output

    loss = XE(softmax(o, dim=-1), c) + l_KD * MSE(t, o)
    loss.backward()                       # backpropagate

    update(go)                            # SGD
    gt.params = l * gt.params + (1 - l) * go.params


def MSE(t, s):
    t = t.detach()                        # stop gradient
    return (t - s).square().mean()
```

Extension to the SCST stage. The same training methodology is also applied during SCST fine-tuning. In this case, given that both language models generate a beam of k captions, there are k^2 pairs of sequences on which the MSE loss can be potentially applied. In the following, we experiment by matching the top-1 caption in each beam, according to the probability assigned by the models themselves or to the score assigned by an external image-text model. Further, we also experiment when matching each caption in the online beam with a caption in the target beam, and then applying the MSE loss on each pair.

Network architecture. CaMEL follows an encoder-decoder Transformer [46] architecture, where the encoder processes visual features via bi-directional attention and the decoder generates captions in an auto-regressive manner. Following [13],

TABLE I

RESULTS ON THE COCO KARPATHY-TEST SPLIT WITH DIFFERENT VISUAL ENCODERS WHEN TRAINING WITH CROSS-ENTROPY LOSS.

				Online Network						**Target Network**					
	CaMEL	λ_{KD}	w/ mesh	B-1	B-4	M	R	C	S	B-1	B-4	M	R	C	S
Faster R-CNN [6], [54]		-		-	-	-	-	-	-	75.4	35.8	27.9	**56.4**	113.9	20.7
	✓	0.01		75.1	35.0	27.4	55.9	112.2	20.4	**75.8**	36.0	27.9	**56.4**	114.7	20.6
	✓	0.1		75.2	35.1	27.6	55.7	111.9	20.4	75.7	**36.1**	**28.0**	56.3	**114.8**	**20.8**
CLIP-RN50 [18]		-		-	-	-	-	-	-	75.4	35.4	27.5	56.1	112.8	20.6
	✓	0.01		74.1	34.1	27.6	55.4	110.4	20.5	75.0	35.1	27.8	56.0	112.9	20.7
	✓	0.1		75.5	35.4	27.6	56.3	113.7	20.6	**75.7**	**36.2**	**28.1**	**56.7**	**115.9**	**20.8**
CLIP-ViT-B16 [18]		-		-	-	-	-	-	-	77.2	37.1	28.7	57.5	122.0	21.7
	✓	0.01		77.0	36.8	28.6	57.5	119.6	21.5	77.5	37.9	**29.1**	**58.1**	**122.5**	**21.9**
	✓	0.1		76.6	36.3	28.3	56.9	117.9	21.2	**77.8**	**38.0**	**29.1**	58.0	**122.5**	21.8
CLIP-RN50×16 [18]		-		-	-	-	-	-	-	77.6	37.6	28.7	57.6	121.0	21.7
	✓	0.01		78.0	37.4	28.7	57.8	121.4	21.9	78.4	38.7	29.3	**58.5**	124.7	**22.3**
	✓	0.05		77.2	37.4	29.0	57.9	121.4	21.9	78.0	38.4	**29.4**	**58.5**	125.4	22.2
	✓	0.1		77.7	37.8	29.0	58.0	122.3	22.0	78.3	**39.1**	**29.4**	**58.5**	**125.7**	22.2
	✓	0.5		77.9	37.6	28.5	57.7	121.1	21.6	**78.5**	39.0	29.3	**58.5**	125.0	22.2
	✓	1.0		77.7	36.7	28.3	57.3	120.0	21.7	**78.5**	38.9	29.3	**58.5**	125.1	**22.3**
CLIP-RN50×16 [18]		-	✓	-	-	-	-	-	-	77.5	37.6	29.0	58.0	122.6	21.9
	✓	0.01	✓	77.5	37.8	29.0	58.0	123.1	21.9	78.0	**38.8**	**29.4**	**58.6**	**125.0**	**22.2**
	✓	0.05	✓	78.0	37.5	28.7	57.9	121.0	21.6	**78.2**	38.5	29.3	58.4	124.4	22.1
	✓	0.1	✓	77.3	36.8	28.5	57.3	120.1	21.7	78.0	38.5	29.3	58.4	124.2	**22.2**
	✓	0.5	✓	77.4	36.9	28.5	57.3	119.7	21.7	77.9	38.6	29.3	58.3	124.2	22.1
	✓	1.0	✓	77.0	36.7	28.4	57.2	119.7	21.5	77.7	38.3	29.3	58.2	123.7	22.0

[50], our encoder incorporates additional memory slots, enhancing its ability to encode knowledge and relations learned from visual data. Specifically, we expand the set of keys and values in self-attention layers with extra and independent learnable vectors, which can encode a priori knowledge retrieved through attention. Our decoder is composed of a stack of decoder layers, each performing a right-masked self-attention and a cross-attention followed by a position-wise feed-forward network.

We also test the usage of a mesh-like connectivity between the encoder and the decoder, following [13]. In this case, the mesh mechanism further connects each encoder and decoder layer in a mesh-like structure, augmenting its ability to deal with low- or high-level features. The architecture is the same for both online and target models, while using independent parameters updated with different strategies during training.

IV. EXPERIMENTS

A. Experimental Settings

Dataset. Following the dominant paradigm in literature, we train and evaluate our model on the COCO dataset [29]. As such, we do not rely on large-scale image-text datasets [17]. COCO is composed of more than $120,000$ images, each of them associated with 5 human-collected captions. We follow the splits defined in [2], using $5,000$ images for both validation and testing and the rest for training. We also evaluate our model on the COCO online test server, which includes $40,775$ images for which annotated captions are not publicly available.

Metrics. According to the standard evaluation protocol, we employ the complete set of captioning metrics: BLEU [55], METEOR [56], ROUGE [57], CIDEr [53], and SPICE [58].

Implementation details. To represent words, we use Byte Pair Encoding (BPE) [59] with a vocabulary size of $49,408$, which is then linearly projected to the input dimensionality of the model. We use standard sinusoidal positional encodings [46] to represent word positions. All models comprise three layers in the visual encoder and three layers in the decoder, each with a dimensionality of 512, a feed-forward dimensionality of 2048, and a number of heads equal to 8. We apply dropout at the output of each sub-layer, with a dropout probability equal to 0.1. The number of memory slots is set to 40.

In all experiments, we employ Adam [60] as optimizer and a beam size equal to 5. During pre-training with XE loss, we use a batch size of 50, following the typical Transformer learning rate scheduling strategy [46] with a warmup equal to $10,000$ iterations. During the SCST finetuning stage, we use a batch size of 30 and a fixed learning rate equal to 5×10^{-6}. The MSE loss is computed by considering only valid tokens and masking the rest. In according with [24], the target network momentum λ is set to 0.999.

B. Ablation Study

Visual features assessment. We firstly discuss the role of visual features, by comparing traditional detection-based features with grid-based features extracted from modern multi-modal models. In particular, we consider object detection features extracted from a Faster R-CNN model [6] pre-trained on the Visual Genome dataset [61], and grid-based features extracted from CLIP [18], which has been trained with language supervision. Since CLIP visual encoders can be either based on ViT-like or CNN-like architectures, we either employ the output of the last Transformer layer, removing the CLS token,

CaMEL: *A man hitting a tennis ball with a tennis racket.*
CaMEL w/ mesh: *A man jumping to hit a tennis ball with a tennis racket.*

CaMEL: *A vase of flowers on a table.*
CaMEL w/ mesh: *A vase of roses on a table with a bird.*

CaMEL: *A woman riding a skateboard on the street.*
CaMEL w/ mesh: *A woman riding a skateboard on the sidewalk in front of a restaurant.*

CaMEL: *A dog running with a frisbee in its mouth.*
CaMEL w/ mesh: *A brown dog carrying a purple frisbee in its mouth.*

CaMEL: *A bowl of cereal with bananas.*
CaMEL w/ mesh: *A bowl of cereal with bananas and a spoon.*

CaMEL: *A brown couch with a couch and a remote control.*
CaMEL w/ mesh: *A living room with a couch and a remote control.*

Fig. 3. Qualitative results on sample images from the COCO test set.

TABLE II
RESULTS ON THE COCO KARPATHY-TEST SPLIT WITH DIFFERENT KNOWLEDGE DISTILLATION STRATEGIES DURING CIDER OPTIMIZATION.

	B-1	B-4	M	R	C	S
$\text{CaMEL}_{\text{best}}$ – CLIP Text Embeddings	82.6	40.7	**30.3**	**60.1**	138.2	24.3
$\text{CaMEL}_{\text{all}}$ – Hungarian Matching	82.7	40.8	30.2	60.0	138.4	24.0
$\text{CaMEL}_{\text{best}}$ – Hungarian Matching	82.4	40.4	30.1	59.8	138.6	23.8
$\text{CaMEL}_{\text{all}}$	**82.9**	**41.0**	30.2	**60.1**	138.5	24.0
$\text{CaMEL}_{\text{best}}$	82.7	40.9	**30.3**	**60.1**	**138.9**	**24.5**

or extract the grid of features produced immediately after the last convolutional layer.

As it can be seen from Table I, Faster R-CNN features can be surpassed by modern multi-modal architectures, which brings a significant advantage in terms of all captioning metrics. While CLIP-RN50 performs worse than Faster R-CNN features when training a single language model without CaMEL and mesh-like connectivity, ViT-based and larger CNN-based models achieve better performance. The best performance is reached by the CLIP-RN50×16 variant, which employs an EfficientNet-style architecture scaling. Overall, this brings an improvement of 7.1 CIDEr points with respect to traditional detection-based features (from 113.9 to 121.0) when training a single language model without the CaMEL technique and without mesh-like connectivity.

Role of CaMEL. We then assess the impact of the proposed training technique during the XE pre-training stage, by also testing with different weights for the knowledge distillation loss, which we indicate with λ_{KD}. Results are reported in Table I. As it can be seen, CaMEL improves the performance when used with all the previously mentioned features, by a considerable margin. This confirms the appropriateness of using a mean teacher learning paradigm in image captioning. When using the RN50×16 encoder, for instance, applying CaMEL with a distillation weight of 0.1 brings an improvement of 4.7 CIDEr points (121.0 vs 125.7). Finally, the CaMEL training strategy improves the performance also when using a mesh-like connectivity, from 122.6 to 125.0 CIDEr points when using λ_{KD} equal to 0.01.

Overall, during XE pre-training CLIP features bring a 6.2% relative improvement with respect to traditional detection-based features, and CaMEL introduces, in the best feature configuration, a relative advancement of 3.9% with respect to a typical single model training. In the following experiments

TABLE III
COMPARISON WITH THE STATE OF THE ART ON THE KARPATHY-TEST SPLIT.

	B-1	B-4	M	R	C	S
Up-Down [6]	79.8	36.3	27.7	56.9	120.1	21.4
ORT [11]	80.5	38.6	28.7	58.4	128.3	22.6
GCN-LSTM [34]	80.9	38.3	28.6	58.5	128.7	22.1
SGAE [35]	81.0	39.0	28.4	58.9	129.1	22.2
CPTR [37]	81.7	40.0	29.1	59.4	129.4	-
MT [36]	80.8	38.9	28.8	58.7	129.6	22.3
AoANet [7]	80.2	38.9	29.2	58.8	129.8	22.4
$\mathcal{M}^2$ Transformer [13]	80.8	39.1	29.2	58.6	131.2	22.6
X-LAN [12]	80.8	39.5	29.5	59.2	132.0	23.4
TCTS [52]	81.2	40.1	29.5	59.3	132.3	23.5
X-Transformer [12]	80.9	39.7	29.5	59.1	132.8	23.4
DPA [8]	80.3	40.5	29.6	59.2	133.4	23.3
DLCT [14]	81.4	39.8	29.5	59.1	133.8	23.0
RSTNet [45]	81.8	40.1	29.8	59.5	135.6	23.3
CaMEL	82.7	40.9	**30.3**	**60.1**	138.9	**24.5**
CaMEL w/ mesh	**82.8**	**41.3**	30.2	**60.1**	**140.6**	23.9
VinVL_B [20]	*82.0*	*40.9*	*30.9*	*60.7*	*140.6*	*25.1*
VinVL_L [20]	*82.0*	*41.0*	*31.1*	*60.9*	*140.9*	*25.2*

we employ the CLIP-RN50×16 variant as image encoder and the target network as language model.

Evaluation of different SCST strategies. Turning to the evaluation of the SCST fine-tuning stage, in Table II we compare the performance of different knowledge distillation strategies. In particular, we experiment the $\text{CaMEL}_{\text{best}}$ version, in which we pair the two logits sequences with the highest probability from both models to compute the KD loss, and the $\text{CaMEL}_{\text{all}}$ version where we use all the sequences generated by the beam search algorithm, pairing them in sorted order of log probability. Further, we explore the use of CLIP text embeddings in the MSE loss. In $\text{CaMEL}_{\text{best}}$ with CLIP text embeddings, we select the most probable caption in each of the two beams, and then apply the MSE loss between the CLS tokens given by the CLIP text encoder.

Moreover, we investigate the usage of the Hungarian matching algorithm [62] to couple captions in the online and target beams, using their CLIP embedding similarity as distance function. We then compute the MSE loss on the original logits between all pairs – in $\text{CaMEL}_{\text{all}}$ version – or only the most probable caption from the target model and its most similar one from the online model – in $\text{CaMEL}_{\text{best}}$ version. The best version, based on all metrics except BLEU scores, is $\text{CaMEL}_{\text{best}}$ without additional algorithms, while for the BLEU

TABLE IV
LEADERBOARD OF VARIOUS METHODS ON THE ONLINE COCO TEST SERVER. THE † MARKER INDICATES ENSEMBLE CONFIGURATIONS.

	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
Up-Down [6]†	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
SGAE [35]†	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
TCTS [52]	80.5	94.8	65.3	89.1	50.9	80.5	39.0	70.3	29.0	38.4	58.9	74.0	125.3	127.2
CPTR [37]	81.8	95.0	66.5	89.4	51.8	80.9	39.5	70.8	29.1	38.3	59.2	74.4	125.4	127.3
AoANet [7]†	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
$\mathcal{M}^2$ Transformer [13]†	81.6	96.0	66.4	90.8	51.8	82.7	39.7	72.8	29.4	39.0	59.2	74.8	129.3	132.1
X-Transformer [12]†	81.9	95.7	66.9	90.5	52.4	82.5	40.3	72.4	29.6	39.2	59.5	75.0	131.1	133.5
DPA [8]†	81.8	96.3	66.5	91.2	51.9	83.2	39.8	73.3	29.6	39.3	59.4	75.1	130.4	133.7
RSTNet [45]†	82.1	96.4	67.0	91.3	52.2	83.0	40.0	73.1	29.6	39.1	59.5	74.6	131.9	134.0
DLCT [14]†	82.4	96.6	67.4	91.7	52.8	83.8	40.6	74.0	29.8	39.6	59.8	75.3	133.3	135.4
CaMEL	82.2	96.6	67.2	91.7	52.5	83.8	40.2	73.7	30.0	39.6	59.6	75.2	133.7	136.4
CaMEL w/ mesh	82.6	96.8	67.5	91.9	52.8	83.9	40.5	73.8	29.9	39.4	59.8	74.9	135.1	137.7
CaMEL w/ mesh†	**83.2**	**97.3**	**68.3**	**92.7**	**53.6**	**84.8**	**41.2**	**74.9**	**30.2**	**39.7**	**60.2**	**75.6**	**137.5**	**140.0**

metrics the best one is $\text{CaMEL}_{\text{all}}$ always without the use of additional algorithms to pair captions.

C. Comparison with the state of the art

We compare the results of CaMEL with those of several recent image captioning models trained without large-scale vision-and-language pre-training. In our analysis, we include methods with LSTM-based language models and attention over image regions such as Up-Down [6], either enhanced with graph-based encoding (*i.e.* GCN-LSTM [34], SGAE [35], and MT [36]) or self-attention (*i.e.* AoANet [7], X-LAN [12], DPA [8], and TCTS [52]), and captioning architectures entirely based on the Transformer network such as ORT [11], $\mathcal{M}^2$ Transformer [13], X-Transformer [12], CPTR [37], DLCT [14], and RSTNet [45].

Performance on COCO. As it can be observed from Table III, our proposal reaches 138.9 CIDEr points, beating all the compared approaches. Adding the mesh-like connectivity to the decoder further improves the results to 140.6 CIDEr points. This represents an increase of 5.0 CIDEr points with respect to the current state-of-the-art when training on the COCO dataset only [45]. Further, in Fig. 1 we compare the aforementioned approaches in terms of both CIDEr and number of parameters. As it can be noticed, not only CaMEL reports state-of-the-art results in terms of caption quality, but it also features a significant reduction in terms of number of trainable weights. As most of previous literature has increased caption quality by increasing the model capacity, our approach represents an outlier in this trend, and demonstrates that state-of-the-art CIDEr levels can be obtained even with a lightweight model.

Finally, in Table III and Fig. 1 we also report the performance obtained by a recent approach which employs large-scale pre-training on external data, *i.e.* VinVL [20]. While this approach is not directly comparable with CaMEL considering that it employs more data, we notice that our proposal is on pair with the Base version of VinVL, and only 0.3 CIDEr points below its Large version. In terms of model size and number of parameters, VinVL is also extremely more demanding than our proposal (cfr. Fig. 1). This further confirms that the usage of proper visual features and of mean teacher learning strategy can achieve a good caption quality with a reduced model size.

Online evaluation. Finally, we also report the performance of our method on the online COCO test server[1]. In this case, we also employ an ensemble of four models trained with the mesh-like connectivity. The evaluation is done on the COCO test split, for which ground-truth annotations are not publicly available. Results are reported in Table IV, in comparison with the top-performing approaches of the leaderboard. As it can be seen, our method surpasses the current state of the art on all metrics, achieving an advancement of 4.2 CIDEr points with respect to the best performer.

V. CONCLUSION

In this work, we investigated the use of the mean teacher learning paradigm applied to the image captioning task. We presented CaMEL, Captioner with Mean tEacher Learning, a novel Transformer-based network that is trained with the interaction of two different language models that learn from each other through knowledge distillation and model averaging. Experimentally, we validated our method with different knowledge distillation strategies and visual feature extractors, surpassing the current state of the art on the COCO dataset without using external data and pre-training strategies. Furthermore, CaMEL achieves similar performance to other recent proposals that make use of large-scale pre-training, while being much smaller in terms of number of parameters.

ACKNOWLEDGMENT

This work has been supported by "Fondazione di Modena", by the "Artificial Intelligence for Cultural Heritage (AI4CH)" project, co-funded by the Italian Ministry of Foreign Affairs and International Cooperation, and by the PRIN project "CREATIVE: CRoss-modal understanding and gEnerATIon of Visual and tExtual content" (CUP B87G22000460001), co-funded by the Italian Ministry of University and Research.

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

References

[1] M. Stefanini, M. Cornia, L. Baraldi, S. Cascianelli, G. Fiameni, and R. Cucchiara, "From Show to Tell: A Survey on Deep Learning-based Image Captioning," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022.

[2] A. Karpathy and L. Fei-Fei, "Deep visual-semantic alignments for generating image descriptions," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2015.

[3] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan, "Show and tell: A neural image caption generator," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2015.

[4] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhutdinov, R. S. Zemel, and Y. Bengio, "Show, attend and tell: Neural image caption generation with visual attention," in *Proceedings of the International Conference on Machine Learning*, 2015.

[5] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan, "Show and Tell: Lessons Learned from the 2015 MSCOCO Image Captioning Challenge," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 4, pp. 652–663, 2016.

[6] P. Anderson, X. He, C. Buehler, D. Teney, M. Johnson, S. Gould, and L. Zhang, "Bottom-up and top-down attention for image captioning and visual question answering," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018.

[7] L. Huang, W. Wang, J. Chen, and X.-Y. Wei, "Attention on Attention for Image Captioning," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019.

[8] F. Liu, X. Ren, X. Wu, S. Ge, W. Fan, Y. Zou, and X. Sun, "Prophet Attention: Predicting Attention with Future Attention," in *Advances in Neural Information Processing Systems*, 2020.

[9] S. J. Rennie, E. Marcheret, Y. Mroueh, J. Ross, and V. Goel, "Self-critical sequence training for image captioning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2017.

[10] F. Landi, L. Baraldi, M. Cornia, and R. Cucchiara, "Working Memory Connections for LSTM," *Neural Networks*, vol. 144, pp. 334–341, 2021.

[11] S. Herdade, A. Kappeler, K. Boakye, and J. Soares, "Image Captioning: Transforming Objects into Words," in *Advances in Neural Information Processing Systems*, 2019.

[12] Y. Pan, T. Yao, Y. Li, and T. Mei, "X-Linear Attention Networks for Image Captioning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020.

[13] M. Cornia, M. Stefanini, L. Baraldi, and R. Cucchiara, "Meshed-Memory Transformer for Image Captioning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020.

[14] Y. Luo, J. Ji, X. Sun, L. Cao, Y. Wu, F. Huang, C.-W. Lin, and R. Ji, "Dual-Level Collaborative Transformer for Image Captioning," in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2021.

[15] M. Cornia, L. Baraldi, and R. Cucchiara, "Explaining transformer-based image captioning models: An empirical analysis," *AI Communications*, pp. 1–19, 2021.

[16] S. Shen, L. H. Li, H. Tan, M. Bansal, A. Rohrbach, K.-W. Chang, Z. Yao, and K. Keutzer, "How Much Can CLIP Benefit Vision-and-Language Tasks?" *arXiv preprint arXiv:2107.06383*, 2021.

[17] M. Cornia, L. Baraldi, G. Fiameni, and R. Cucchiara, "Universal Captioner: Inducing Content-Style Separation in Vision-and-Language Model Training," *arXiv preprint arXiv:2111.12727*, 2022.

[18] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, and I. Sutskever, "Learning Transferable Visual Models From Natural Language Supervision," *arXiv preprint arXiv:2103.00020*, 2021.

[19] X. Li, X. Yin, C. Li, P. Zhang, X. Hu, L. Zhang, L. Wang, H. Hu, L. Dong, F. Wei *et al.*, "Oscar: Object-semantics aligned pre-training for vision-language tasks," in *Proceedings of the European Conference on Computer Vision*, 2020.

[20] P. Zhang, X. Li, X. Hu, J. Yang, L. Zhang, L. Wang, Y. Choi, and J. Gao, "VinVL: Revisiting visual representations in vision-language models," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021.

[21] Z. Wang, J. Yu, A. W. Yu, Z. Dai, Y. Tsvetkov, and Y. Cao, "SimVLM: Simple visual language model pretraining with weak supervision," in *Proceedings of the International Conference on Learning Representations*, 2022.

[22] X. Hu, Z. Gan, J. Wang, Z. Yang, Z. Liu, Y. Lu, and L. Wang, "Scaling Up Vision-Language Pre-training for Image Captioning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022.

[23] S. Liu, Z. Zhu, N. Ye, S. Guadarrama, and K. Murphy, "Improved Image Captioning via Policy Gradient Optimization of SPIDEr," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2017.

[24] A. Tarvainen and H. Valpola, "Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results," in *Advances in Neural Information Processing Systems*, 2017.

[25] M. Caron, H. Touvron, I. Misra, H. Jégou, J. Mairal, P. Bojanowski, and A. Joulin, "Emerging Properties in Self-Supervised Vision Transformers," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021.

[26] G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," in *Advances in Neural Information Processing Systems Workshops*, 2015.

[27] R. Anil, G. Pereyra, A. Passos, R. Ormandi, G. E. Dahl, and G. E. Hinton, "Large scale distributed neural network training through online distillation," in *Proceedings of the International Conference on Learning Representations*, 2018.

[28] Q. Xie, M.-T. Luong, E. Hovy, and Q. V. Le, "Self-Training With Noisy Student Improves ImageNet Classification," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020.

[29] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft COCO: Common Objects in Context," in *Proceedings of the European Conference on Computer Vision*, 2014.

[30] J. Donahue, L. Anne Hendricks, S. Guadarrama, M. Rohrbach, S. Venugopalan, K. Saenko, and T. Darrell, "Long-term recurrent convolutional networks for visual recognition and description," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2015.

[31] J. Lu, C. Xiong, D. Parikh, and R. Socher, "Knowing when to look: Adaptive attention via a visual sentinel for image captioning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2017.

[32] Q. You, H. Jin, Z. Wang, C. Fang, and J. Luo, "Image captioning with semantic attention," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2016.

[33] J. Lu, J. Yang, D. Batra, and D. Parikh, "Neural Baby Talk," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018.

[34] T. Yao, Y. Pan, Y. Li, and T. Mei, "Exploring Visual Relationship for Image Captioning," in *Proceedings of the European Conference on Computer Vision*, 2018.

[35] X. Yang, K. Tang, H. Zhang, and J. Cai, "Auto-Encoding Scene Graphs for Image Captioning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019.

[36] Z. Shi, X. Zhou, X. Qiu, and X. Zhu, "Improving Image Captioning with Better Use of Captions," in *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, 2020.

[37] W. Liu, S. Chen, L. Guo, X. Zhu, and J. Liu, "CPTR: Full Transformer Network for Image Captioning," *arXiv preprint arXiv:2101.10804*, 2021.

[38] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly *et al.*, "An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale," in *Proceedings of the International Conference on Learning Representations*, 2021.

[39] H. Touvron, M. Cord, M. Douze, F. Massa, A. Sablayrolles, and H. Jégou, "Training data-efficient image transformers & distillation through attention," in *Proceedings of the International Conference on Machine Learning*, 2021.

[40] P. Bruno, R. Amoroso, M. Cornia, S. Cascianelli, L. Baraldi, and R. Cucchiara, "Investigating Bidimensional Downsampling in Vision Transformer Models," in *Proceedings of the International Conference on Image Analysis and Processing*, 2022.

[41] J. Aneja, A. Deshpande, and A. G. Schwing, "Convolutional image captioning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018.

[42] X. Yang, H. Zhang, and J. Cai, "Learning to Collocate Neural Modules for Image Captioning," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019.

[43] G. Li, L. Zhu, P. Liu, and Y. Yang, "Entangled Transformer for Image Captioning," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019.

[44] M. Cagrandi, M. Cornia, M. Stefanini, L. Baraldi, and R. Cucchiara, "Learning to Select: A Fully Attentive Approach for Novel Object Captioning," in *Proceedings of the ACM International Conference on Multimedia Retrieval*, 2021.

[45] X. Zhang, X. Sun, Y. Luo, J. Ji, Y. Zhou, Y. Wu, F. Huang, and R. Ji, "RSTNet: Captioning with Adaptive Attention on Visual and Non-Visual Words," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021.

[46] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems*, 2017.

[47] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proceedings of the Annual Conference of the North American Chapter of the Association for Computational Linguistics*, 2018.

[48] S. Sukhbaatar, E. Grave, G. Lample, H. Jegou, and A. Joulin, "Augmenting Self-attention with Persistent Memory," *arXiv preprint arXiv:1907.01470*, 2019.

[49] L. Guo, J. Liu, X. Zhu, P. Yao, S. Lu, and H. Lu, "Normalized and Geometry-Aware Self-Attention Network for Image Captioning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020.

[50] M. Cornia, L. Baraldi, and R. Cucchiara, "SMArT: Training Shallow Memory-aware Transformers for Robotic Explainability," in *Proceedings of the IEEE International Conference on Robotics and Automation*, 2020.

[51] L. Zhou, H. Palangi, L. Zhang, H. Hu, J. J. Corso, and J. Gao, "Unified Vision-Language Pre-Training for Image Captioning and VQA," in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2020.

[52] Y. Huang and J. Chen, "Teacher-Critical Training Strategies for Image Captioning," *arXiv preprint arXiv:2009.14405*, 2020.

[53] R. Vedantam, C. Lawrence Zitnick, and D. Parikh, "CIDEr: Consensus-based Image Description Evaluation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2015.

[54] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: towards real-time object detection with region proposal networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 6, pp. 1137–1149, 2017.

[55] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "BLEU: a method for automatic evaluation of machine translation," in *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, 2002.

[56] S. Banerjee and A. Lavie, "METEOR: An automatic metric for MT evaluation with improved correlation with human judgments," in *Proceedings of the Annual Meeting of the Association for Computational Linguistics Workshops*, 2005.

[57] C.-Y. Lin, "Rouge: A package for automatic evaluation of summaries," in *Proceedings of the Annual Meeting of the Association for Computational Linguistics Workshops*, 2004.

[58] P. Anderson, B. Fernando, M. Johnson, and S. Gould, "SPICE: Semantic Propositional Image Caption Evaluation," in *Proceedings of the European Conference on Computer Vision*, 2016.

[59] R. Sennrich, B. Haddow, and A. Birch, "Neural Machine Translation of Rare Words with Subword Units," in *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, 2016.

[60] D. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," in *Proceedings of the International Conference on Learning Representations*, 2015.

[61] R. Krishna, Y. Zhu, O. Groth, J. Johnson, K. Hata, J. Kravitz, S. Chen, Y. Kalantidis, L.-J. Li, D. A. Shamma, M. Bernstein, and L. Fei-Fei, "Visual Genome: Connecting Language and Vision Using Crowdsourced Dense Image Annotations," *International Journal of Computer Vision*, vol. 123, no. 1, pp. 32–73, 2017.

[62] H. W. Kuhn, "The Hungarian method for the assignment problem," *Naval Research Logistics Quarterly*, vol. 2, no. 1-2, pp. 83–97, 1955.