

End-to-End Video Captioning with Multitask Reinforcement Learning

Lijun Li
Beihang University
lilijun1990@buaa.edu.cn

Boqing Gong
Tencent AI Lab
Bellevue, WA 98004
boqinggo@outlook.com

Abstract

Although end-to-end (E2E) learning has led to promising performance on a variety of tasks, it is often impeded by hardware constraints (e.g., GPU memories) and is prone to overfitting. When it comes to video captioning, one of the most challenging benchmark tasks in computer vision and machine learning, those limitations of E2E learning are especially amplified by the fact that *both the input videos and output captions are lengthy sequences*. Indeed, state-of-the-art methods of video captioning process video frames by convolutional neural networks and generate captions by unrolling recurrent neural networks. If we connect them in an E2E manner, the resulting model is both memory-consuming and data-hungry, making it extremely hard to train. In this paper, we propose a *multitask reinforcement learning* approach to training an E2E video captioning model. The main idea is to *mine and construct as many effective tasks* (e.g., attributes, rewards, and the captions) as possible from the human captioned videos such that they can jointly regulate the search space of the E2E neural network, from which an E2E video captioning model can be found and generalized to the testing phase. To the best of our knowledge, this is the first video captioning model that is trained end-to-end from the raw video input to the caption output. Experimental results show that such a model outperforms existing ones to a large margin on two benchmark video captioning datasets.

1. Introduction

Video captioning, i.e., to automatically describe videos by full sentences or phrases, not only serves as a challenging testbed in computer vision and machine learning but also benefits many real-world applications. The generated captions may enable fast video retrieval, assist the visually impaired, and engage users in a versatile chatbot, to name a few.

Most recent works [1, 2, 3, 4] that tackle this problem fall under an *encoder-decoder framework* which has been

shown effective in *speech recognition* [5, 6], *natural language translation* [7, 8], and *image captioning* [9, 10]. The encoder extracts compact representations of the visual content. In the context of video captioning, the convolutional neural networks (CNNs) are usually used to encode the video frames followed by a temporal model [3, 11, 12, 13] or simply temporal pooling [14] and the decoder maps the codes to a sequence of words often by the recurrent neural networks (RNNs) [15, 16] (e.g., the long short-term memory (LSTM) [17] units are a popular choice). In order to train such networks, *most existing works employ a cross-entropy loss at each decoding step*. We refer the readers to the seminal work that spurs the resurging interests in video captioning, sequence to sequence - video to text (S2VT) [1], for a quick understanding about the backbone techniques.

Despite the impact of the encoder-decoder framework on video captioning, it inherently impedes the use of end-to-end (E2E) training which has shown very promising results across a large variety of tasks. Indeed, both CNNs and RNNs are memory consuming, leaving little GPU space to the training data which are yet key to the training procedure. Besides, the input videos and output sentences are both sequences, making the encoder-decoder framework very lengthy and data-hungry. On the one hand, it is tempting to test the broadly effective E2E training strategy on the video captioning task. On the other hand, this seemingly straightforward idea is confined by the hardware and the relatively small size of existing video captioning datasets. Our experiments show that the conventional cross-entropy loss coupled with stochastic gradient descent cannot extract the effectiveness of the E2E training.

In this paper, we propose a multitask reinforcement learning approach to training an E2E video captioning model. Our main idea is to mine and construct as many effective tasks as possible from the human captioned videos such that they can jointly regulate the search space of the encoder-decoder network, from which an E2E video captioning model can be found and generalized to the testing phase. When the training set is relatively small for the big encoder-decoder network, it is important to mine as much

information as possible from the limited data.

Furthermore, we propose an effective multitask end-to-end framework with the reinforcement learning and it can **improve the performance of video captioning greatly**. Although many existing models [1, 18, 19] can literally be executed at the end-to-end fashion at the test stage, ours is the first to end-to-end train the model in the training stage, to the best of our knowledge. This is nontrivial because the model becomes very large in order to take as input a raw video sequence and output a sequence of words, causing challenges to the computational resources and raising the need of large-scale well-labeled data. We have tested a number of techniques to train such a big model and report the ones with success in this paper. With the help of multitask end-to-end method, our network can learn much specific video representations. Instead of training with cross entropy, our work can avoid the exposure bias, objective discrepancy of training and testing. As a result, we achieve state-of-the-art results on MSVD dataset [20] and MSR-VTT dataset [21] for the video captioning task.

Our contributions are as following: (1) Unlike most works, our framework can gain performance not only from decoding part, but also in the visual encoding part. (2) We utilize the attributes in sentences to train multitask framework without using external data. (3) Our end-to-end pipeline makes the video representation more effective. (4) We modify the reinforcement learning to multisample the reward. (5) The results show that our approaches can lead to the state-of-the-art results both on MSVD dataset and MSR-VTT dataset.

2. Related works

Large amount of progress has been made in image and video captioning. A large part of it is due to the advances in machine translation. For example, the encoder-decoder framework and the attention mechanism were first introduced in machine translation [22, 23, 24] and then extended to captioning. Both image captioning approaches [9, 25, 26] and video captioning methods [27, 3, 28, 29] follow their pipeline and also apply attention mechanism in caption generation. Comparing with image captioning, video captioning describes dynamic scenes instead of static scenes. From Figure 1, we can clearly see that the video captioning is much difficult with large variance in appearance. Baraldi et al. [4] propose boundary-aware LSTM cell to automatically detect the temporal video segments. Venugopalan et al. [11] integrate natural language knowledge to their network by training language LSTM model on a large external text corpora. Zhu et al. [30] extend Gated Recurrent Unit (GRU) to multirate GRU to handle different video frame rates. Hendricks et al. [2] propose a deep compositional captioner to describe novel object with the help of lexical classifier training on external image description dataset.

In the recent years, maximum likelihood estimation algorithm has been widely used in video captioning which maximizes the probability of current words based on the previous ground truth words [31, 32, 33, 18, 14]. But they all have two major problems.

The first one is exposure bias which is the input mismatch in training and inference. In training, the output of decoder depends on ground truth words instead of model predictions. While in inference, the decoder only has access to the predictions. Bengio et al. [34] proposed scheduled sampling to mitigate the gap between the training and inference by selecting more often from the ground truth in the beginning but sampling more often from the model predictions in the end. However, it still optimizes at the word level.

The other problem is the objective mismatch between training and inference. In training, it optimizes the loss at the word level. While in inference, discrete metrics such as BLEU4 [35], METEOR [36], CIDEr [37], and ROUGE-L [38] are used for evaluation. A few image captioning works have been proposed to solve the problems and shown superior performance **with the help of reinforcement learning**. Ren et al. [39] introduce actor-critic method to image captioning and also propose a new lookahead inference algorithm which has better performance than beam search. Liu et al. [10] employ policy gradient method to optimize the SPIDER score. Dai et al. [40] combine a conditional generative adversarial network with policy gradient which can produce natural and diverse sentences. However, there are much less works using reinforcement learning in video captioning.

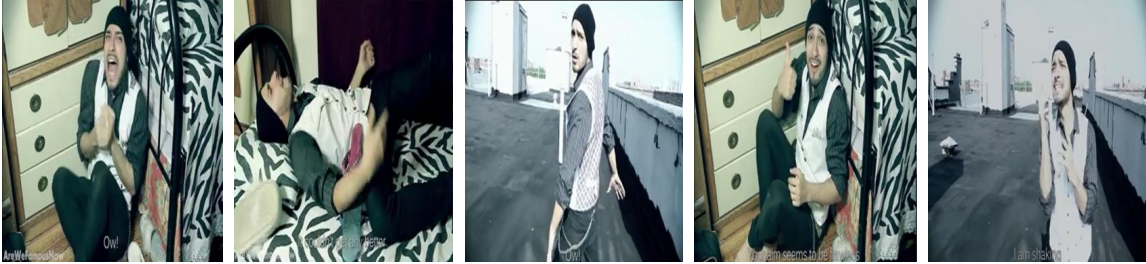
In this paper, we exploit the reinforcement learning in video captioning, especially for the **jointly training of CNNs and LSTMs**. Note that many video captioning models can actually be deployed in an end-to-end manner, such as [1, 18, 19], etc. Venugopalan et al. propose a stack of two LSTM networks [1]. Pan et al. propose a novel transfer unit to feed the semantic concept to LSTM [18]. Yu et al. develop a high-level word detector and semantic attention mechanism which combines the concept with caption decoder [19]. However, they actually treat CNN as feature extractor and do not train the CNN part of their framework. On the contrary, our method trains the CNN and the other part together.

Multitask learning is a kind of machine learning technique. During multitask learning, multiple tasks are solved at the same time with a shared representation and is especially useful with limited number of original data. It has been widely utilized not only in computer vision [41, 42, 43, 44], but also in natural language processing [45]. It becomes a natural choice for us since the model capacity likely outweigh the existing datasets when we aim to update all its weights from the raw video input to the caption



Caption: a man is playing a guitar and singing

Caption: a man is singing and playing guitar in an airport



Caption: a man is singing and doing funny act

Caption: a man making a music video and having slippers thrown at him

Figure 1. Examples in MSVD and MSR-VTT datasets

output. However, few works use multitask learning in video captioning. We explore the effectiveness and find the multitask learning can also be useful in video captioning.

3. An E2E trained captioning model

We describe the end-to-end (E2E) trained video captioning model in this section. It is essentially a deepened version of the S2VT model [1]. Despite its simplicity in concept, it is very challenging to train the whole big model to reach a good generalization capability over the test sets. Both our experiments and an earlier attempt by Yu et al. [19] indicate that **the gain of jointly training the CNNs and LSTMs is only marginal over fixing the CNNs as feature extractors**, if we do not have an effective training approach. To this end, one important contribution of this paper is the batch of techniques presented below which we find useful when they are combined for training the E2E video captioning model.

3.1. Model architecture

Figure 2 sketches the model architecture which consists of three main components. On the top right, **five copies of the same Inception-Resnet-v2 [46] CNN are used to transform the raw video frames to high-level feature representations**. Note that the last classification layer of the Inception-Resnet-v2 is replaced by a fully connected layer whose out-

put dimension is 500. The LSTMs **on the bottom right encode the video frames' feature representations and then decode a sentence** to describe the content in the video. **On the bottom left, there is a branch consisting of a temporal average pooling, and an attribute prediction layer** whose output dimension is 400 and the activation functions are sigmoid. **This branch is introduced to assign relevant attributes to the input video**. It is **not used in the testing phase** of the video captioning, but **it generates informative gradients in the training phase for updating the weights of the CNNs** in addition to those from the LSTMs. The design of the LSTMs (e.g., the number of hidden units, how to compute the input gates, etc.) is borrowed from S2VT [1].

3.2. The E2E training of the model

We train the model progressively in three steps. **The first two steps aim to find a good initialization to the LSTMs** (and the fully connected layer connecting the CNNs and the LSTMs) such that the last step, the E2E training of the whole model, can have a warm start. **The weights of the CNNs are frozen until the third step**.

Step 1 is the standard training approach to S2VT using the cross entropy loss. For an input frame I_t at time step t , we encode it with deep CNN and **embed it with projection matrix W_I** . Then for the projected feature representation x_t , the LSTM computes the hidden state h_t and cell state

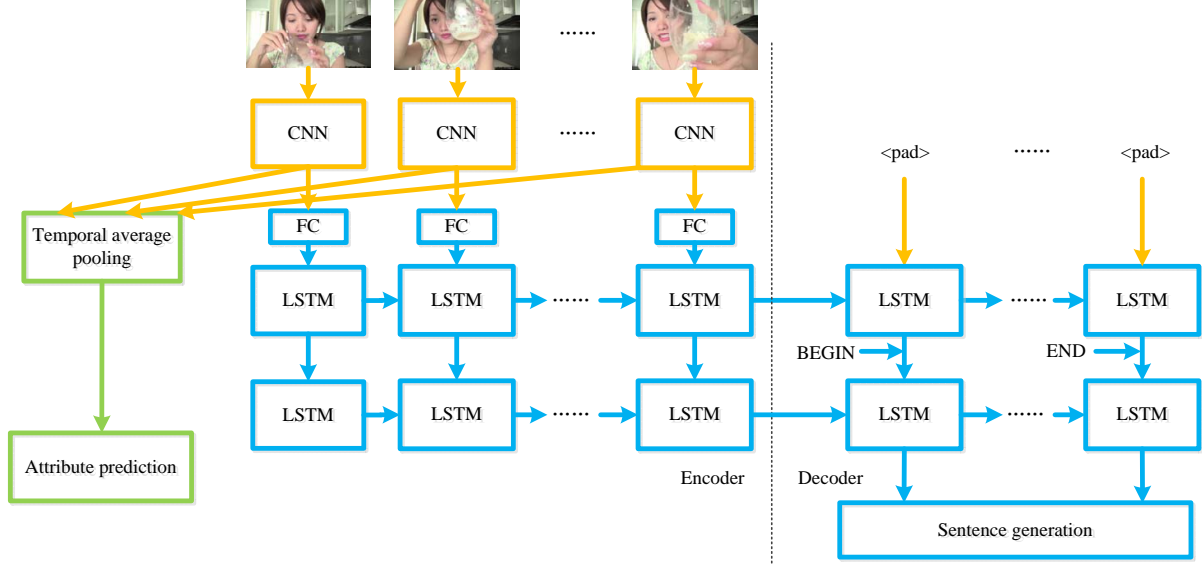


Figure 2. Multitask framework

c_t . The details about the computation of hidden state and cell state are in the following:

$$\begin{aligned}
 i_t &= \sigma(W_{ix}x_t + W_{ih}h_{t-1} + b_i) \\
 f_t &= \sigma(W_{fx}x_t + W_{fh}h_{t-1} + b_f) \\
 o_t &= \sigma(W_{ox}x_t + W_{oh}h_{t-1} + b_o) \\
 g_t &= \phi(W_{gx}x_t + W_{gh}h_{t-1} + b_g) \\
 c_t &= i_t \odot g_t + f_t \odot c_{t-1} \\
 h_t &= o_t \odot \phi(c_t)
 \end{aligned} \tag{1}$$

where σ is sigmoid function, ϕ is the hyperbolic tangent function, \odot is the element-wise multiplication. Second LSTM layer is similar to the first one, except that the input is the combination of first LSTM output and the ground truth word.

Given a “ground truth” sentence $s^* = \{w_1^*, w_2^*, \dots, w_T^*\}$ describing an input video, we minimize the cross entropy loss of the network,

$$L_x(\theta) := -\log p_\theta(s^*) = -\frac{1}{T} \sum_{t=1}^T \log p_\theta(w_t^* | w_1^*, \dots, w_{t-1}^*) \tag{2}$$

where θ denotes the model parameters.

Step 2: REINFORCE+ training of the LSTMs. After Step 1, we introduce the self-critical REINFORCE algorithm [47, 48] to the video captioning to seek better weights for the LSTMs in terms of their generalization performance on the validation and test sets.

Since there are exposure bias and objective mismatch between training and testing in video captioning training with

cross entropy loss, we propose to directly optimize the captioning system by REINFORCE learning as in [48]. In reinforcement learning, the goal is to train an agent to complete tasks by executing a series actions in some environment. While in the context of video captioning, the goal is to generate sentence with the help of video inputs and captioning model. The captioning model is the agent and the action is to predict the next word. The visual inputs and the predicted words is the environment. The reward is the evaluation metrics. For example, we use CIDEr score of sentences as reward. This is the pipeline of reinforcement learning: an agent receives a state which contains the visual features and the word predicted so far, and a reward from environment, then takes an action. With the new action and state, the environment provides another state and reward to the agent.

The objective function of reinforcement learning is:

$$L_r(\theta) = -\mathbb{E}(r(w^s)) \tag{3}$$

where w^s is the sentence consists of (w_1, w_2, \dots, w) sampled from the network, r is the reward of the sentence.

As in [48], we also use the REINFORCE algorithm [47]. The general updates of parameter θ can be written as:

$$\nabla_\theta L_r(\theta) = -\mathbb{E}[r(w) \nabla \log p(w)] \tag{4}$$

In order to reduce the variance, we need to use reward r to minus baseline b :

$$\nabla_\theta L_r(\theta) = -\mathbb{E}[(r(w) - b) \nabla_\theta \log p_\theta(w^s)] \tag{5}$$

It is obvious that the gradient is unchanged since baseline b does not depend on the sampled word w^s . In practice, Eq.

(5) is often approximated the expected gradient with one sampling:

$$\nabla_{\theta} L_r(\theta) \approx -(r(w^s) - b) \nabla_{\theta} \log p_{\theta}(w^s) \quad (6)$$

How to choose the baseline b can effect the performance of REINFORCE algorithm. We choose the greedy search of words as our b .

$$\hat{w}_t = \arg \max p(w_t | h_t) \quad (7)$$

At the beginning of each iteration, we sample up to M trajectories (i.e., sentences) from the current model. Denoting them by s_1, \dots, s_M , we can then write down the cost function for generating the gradients of this iteration,

$$L_r(\theta) \approx -\frac{1}{M} \sum_{m=1}^M (r(s_m) - b) \log p_{\theta}(s_m) \quad (8)$$

where $r(s_m)$ is the reward assigned to the trajectory s_m and b is a baseline to reduce the variance of the gradients — note that the gradients computed from Eq. (6) are only an estimate of the real gradients of the expected reward. Readers are referred to [47] for a better understanding. In this paper, we reward the sampled sentences by the CIDEr scores. For the baseline b , we calculate it as the CIDEr score of the sentence generated by the greedy search using the model at the beginning of the iteration. By subtracting the baseline, this loss can increase the probability of sampled captions that have higher rewards than the greedy search words and reduce the probability of sampled captions that have lower rewards. We denote this algorithm as REINFORCE+ or RFC+ in the following.

It is interesting to note that Eq. (8) acts as a running loss over the training course. It changes at different iterations, being realized by the sampled trajectories as opposed to the constant ground truth captions in the cross entropy loss L_x across different iterations. Moreover, the rewards offset by the baseline weigh the contributions of the trajectories to the gradients. Jointly, they push the model trained in Step 1 further to the point that generalizes better to the unseen data.

Step 3: Multitask training of the full model. We jointly tune the full model in this step, freeing the weights of the CNNs. As the starting point, it might seem natural to repeat Step 1 and/or Step 2 for the E2E optimization. However, this only gives rise to marginal gain over freezing the CNNs weights in our experiments. Such quick saturation of accuracy is actually common for very deep neural networks and may be alleviated by the skip connections between different layers of feedforward networks [49, 50]. Our model, however, heterogeneously mixes LSTMs and CNNs, leaving it unclear how to apply the skip connections.

Instead, we propose to supply extra and informative gradients directly to the CNNs, so as to complement those reached to the CNNs indirectly through the LSTMs. Such direct gradients are provided by the attribute prediction branch (cf. Figure 2).

We mine the attributes in the video captions following the previous practice on image captioning [51]. Among the words in the sentences of the training set, we extract the most frequent words including nouns, verbs and adjectives as the attributes. Accordingly, the attribute prediction branch is equipped by sigmoid functions in order to each predict the existence or not (y_i) of an attribute in the input video. We define a binary cross entropy loss for this network branch, denoted by $L_a(\theta) = -\frac{1}{N} \sum_i [y_i \log p_{\theta}(i) + (1 - y_i) \log(1 - p_{\theta}(i))]$, where N is the number of attributes in total and $p_{\theta}(i)$ is the network output for the i -th attribute.

The overall cost function we use in Step 3 is a convex combination of the attribute loss and the REINFORCE loss:

$$L(\theta) = \alpha L_r(\theta) + (1 - \alpha) L_a(\theta) \quad (9)$$

where $\alpha = 0.95$ is selected by the validation set.

4. Comparison experiments

4.1. Datasets and experiment details

In this section, we report the results of our E2E trained model and compare with other state-of-the-art methods on two popular video captioning datasets. One is the MSVD dataset [20]. MSVD consists of 1,970 video clips and 70,028 captions collected via Amazon Mechanical Turk (www.mturk.com) which covers a lots of topics. On average, the video duration is about 10 seconds and each sentence contains about 8 words. A somehow official split of the videos is provided by [1] and maintained by the existing works as well as this paper: 1,200 videos for training, 100 for validation, and 670 for testing. The other is the MSR-VTT dataset which contains 10,000 video clips and 200,000 captions. We use the data split defined in [21] in our experiments: 6,513 videos for training, 497 for validation, and 2,990 for testing. It is the largest publicly available video captioning dataset in terms of the number of sentences. The average duration of the videos is 20 seconds.

We implement our algorithm with Tensorflow [55] which is one of the most widely used deep learning frameworks. In our end-to-end trained model, we keep the layers of Inception-Resnet-v2 [46] until the last pooling layer whose dimension is 1,536. After that, we add a fully connected layer whose output dimension is 500. The dimension of the LSTM hidden layers is 1000. A dropout layer is attached to each LSTM unit during training with dropout rate of 0.2. Each word is represented as one-hot vector. The image embedding dimension and word embedding dimension are both 500. We fix the encoder step size to 5 and decoder

Table 1. Comparison with state-of-the-art methods on the MSVD dataset.

Models/Metrics	BLEU4	ROUGE-L	METEOR	CIDEr
h-RNN [28]	0.499	–	0.326	0.658
Attention fusion [3]	0.524	–	0.320	0.688
BA encoder [4]	0.425	–	0.324	0.635
SCN [31]	0.502	–	<u>0.334</u>	0.770
TDDF [52]	0.458	<u>0.697</u>	<u>0.333</u>	0.730
LSTM-TSA [18]	<u>0.528</u>	–	<u>0.335</u>	0.740
MVRM [30]	0.538	–	0.344	0.812
S2VT (our Step 1) [1]	0.428	0.687	0.325	0.750
REINFORCE (our Step 2) [48]	0.456	0.690	0.329	0.806
REINFORCE+ (our Step 2) [48]	0.466	<u>0.694</u>	0.330	0.816
E2E (ours, greedy search)	0.480	0.705	<u>0.336</u>	0.865
E2E (ours, beam search)	0.503	0.708	0.341	0.875

Table 2. Comparison with state-of-the-art methods on the MSR-VTT dataset.

Models	BLEU4	ROUGE-L	METEOR	CIDEr
TDDF [52]	0.372	0.586	<u>0.277</u>	0.441
v2t_navigator[53]	0.408	<u>0.609</u>	0.282	0.448
Aalto [54]	<u>0.398</u>	0.598	0.269	0.457
Attention fusion [3]	<u>0.394</u>	–	0.257	0.404
S2VT (our Step 1) [1]	0.353	0.578	0.266	0.407
REINFORCE (our Step 2) [48]	0.392	0.603	0.267	0.448
REINFORCE+ (our Step 2) [48]	<u>0.398</u>	<u>0.609</u>	0.271	0.468
E2E (ours, greedy search)	0.404	0.610	0.270	0.483
E2E (ours, beam search)	0.404	0.610	0.270	0.483

Table 3. Ablation experiment: video captioning results on MSVD with greedy decoding.

Models	BLEU4	ROUGE-L	METEOR	CIDEr
S2VT (Step 1) [1]	0.428	0.687	0.325	0.750
RFC (Step 2) [48]	0.456	<u>0.690</u>	<u>0.329</u>	0.806
RFC+ (Step 2) [48]	<u>0.466</u>	<u>0.694</u>	<u>0.330</u>	0.816
E2E (xentropy)	0.439	<u>0.690</u>	<u>0.328</u>	0.767
E2E (att+xentropy)	0.453	<u>0.694</u>	<u>0.331</u>	0.790
E2E (ours)	0.480	0.705	0.336	0.865

Table 4. Ablation experiment: video captioning results on MSR-VTT dataset with greedy decoding.

Models	BLEU4	ROUGE-L	METEOR	CIDEr
S2VT (Step 1) [1]	0.353	0.578	<u>0.266</u>	0.407
RFC (Step 2) [48]	0.392	<u>0.603</u>	0.267	0.448
RFC+ (Step 2) [48]	<u>0.398</u>	0.609	0.271	0.468
E2E (ours)	0.404	0.610	0.270	0.483

step size to 35. All the trainable parameters are initialized with the uniform distribution $[-0.1, 0.1]$. The ADAM optimizer is used in our experiments. The learning rate is $1e-4$ to train S2VT. While for other methods, it is $1e-6$. The hyperparameter α is 0.95 in Eq. (9). For both datasets, we resize the video frames to 224x224. For inference, we use beam search to keep multiple generated words and select the best sentence with beam size 3 in the end. All the free

parameters are chosen by the validation sets. For the evaluation metrics, we choose four types of widely used caption metrics: BLEU4 [35], METEOR [36], CIDEr [37], and ROUGE-L [38]. The scores are calculated using the MS COCO evaluation code [56].

4.2. Baseline methods

Table 1 and 2 show the comparison results with several recently proposed methods on the two datasets, respectively. On the MSVD dataset, we compare our approach with following seven recent methods.

h-RNN [28] proposes a hierarchical-RNN framework and designs an attention scheme over both temporal and spatial dimensions to focus on the visual elements.

Attention fusion [3] develops a modality-dependent attention mechanism together with temporal attention to combines the cues of multiple modalities, which can attend not only specific time, but also the specific modalities.

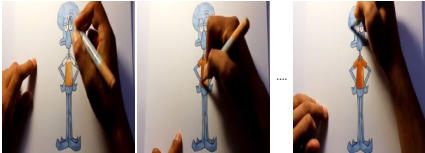


BA encoder [4] presents a new boundary-aware LSTM cell to detect the discontinuity of consecutive frames. Then the cell is used to build a hierarchical encoder and makes its structure adopt to the inputs.

SCN [31] detects semantic concepts from videos, and proposes a tag-dependent LSTM whose weights matrix de-

Table 5. Qualitative results of video captioning on MSVD dataset. Baseline is the sentence generated by our baseline model, MR stands for sentence generated by our multitask reinforce model and GT represents Ground Truth captions

		
Captions:	Captions:	Captions:
S2VT: a man is giving a woman	S2VT: a woman is putting some meat in a pan	S2VT: a soccer player is kicking a soccer ball
E2E: a man is talk	E2E: a woman is frying meat	E2E: men are playing soccer
GT: a man is talking	GT: a woman is frying meat	GT: the men are playing soccer

Table 6. Qualitative results of video captioning on MSR-VTT dataset. Baseline is the sentence generated by our baseline model, MR stands for sentence generated by our multitask reinforce model and GT represents Ground Truth captions

		
Captions:	Captions:	Captions:
S2VT: a person is talking about a computer	S2VT: a man is singing	S2VT: a man is talking about a guitar
E2E: a person is drawing a cartoon	E2E: a man is jumping on a trampoline	E2E: a man is playing a guitar
GT: a person is drawing a cartoon	GT: a man is jumping on a trampoline	GT: a man is playing a guitar

depends on the semantic concepts.

TDDE [52] combines motion feature and appearance feature, and automatically determines which feature should be focused according to the word.

LSTM-TSA [18] presents a transfer unit to control and fuse the attribute, motion, and visual features for the video representations.

MVRM [30] learns a multirate video representation which can adaptively fit the different motion speed in videos.

On the MSR-VTT dataset, we include four methods: TDDE [52], v2t_navigator [53], Aalto [54], and Attention fusion [3].

V2t_navigator [53] ranks top one on the leaderboard of MSR-VTT. It represents the videos by their visual, aural, speech, and category cues, while we only employ the raw video frames in our approach.

Aalto [54] is the second best method on the leaderboard of MSR-VTT. It trains an evaluator network to drive the captioning model towards semantically interesting sentences.

During the experiments, REINFORCE (RFC) denotes the self-critical REINFORCE algorithm extended to video captioning, and REINFORCE+ (RFC+) represents our multi-sampling trajectories REINFORCE algorithm. The E2E is our final multitask reinforcement learning approach to train end-to-end which optimizes the Eq. (9).

4.3. Comparison results

Table 1 and 2 present the results evaluated by BLEU4 [35], METEOR [36], CIDEr [37], and ROUGE-L [38] on MSVD and MSR-VTT, respectively. The CIDEr scores on MSVD dataset are much higher than those on MSR-VTT dataset. The reason may due to the much complex scenes, actions and large variance in MSR-VTT dataset. Our approach is denoted by E2E and two decoding results are reported, one by the greedy search and the other by the beam search of a window size of three. We can see that our approach outperforms the existing ones to large margins under the CIDEr metric, which is taken as the reward function in our training procedure. Under the other metrics, ours is also among the top performing methods while we notice that one can conveniently replace the CIDEr reward by the other metrics as the reward functions. On MSVD dataset, our E2E method can achieve 0.865 in CIDEr. Comparing with the baseline method, S2VT, our E2E model can make relative improvement by 15.3% in CIDEr with greedy decoding. On MSR-VTT dataset, our E2E method can reach 0.483 in CIDEr. It can make relative improvement by 18.6%.

Several factors may have contributed to the superior performance of our model. First, we fine-tune the CNNs such that the extract features of the video frames are purposely tailored for the video captioning task, as opposed to the generic features pre-trained from the ImageNet. Besides,

the LSTM architecture inherently exploits the temporal nature of the videos. At last but not the least, the performance attributes to the progressive and multitask techniques of training the model. Next, we provide in-depth analyses about the last point by some ablated studies.

4.4. Ablation experiments

Due to the time and computation resource constraint, we mainly run the ablation experiments on the MSVD dataset. See Table 3 for the results. Nonetheless, we also report some key results on MSR-VTT in Table 4.

First of all, we note that Step 2 is able to significantly improve the results of Step 1, reinforcing the effectiveness of the REINFORCE algorithm [47]. Besides, by sampling multiple trajectories (cf. row RFC+) we can boost the original REINFORCE by 1% to 2%.

If we skip Step 2 and directly fine-tune the CNNs using the cross entropy loss L_x in Step 3, the results (cf. row E2E (xentropy)) are only marginal better than those of freezing the CNNs. This observation is not surprising, given that the full model is actually both deep in CNNs and long in terms of the unrolled LSTMs, making it very hard to train.

If we skip Step 2 and instead minimize the convex combination of the attribute prediction loss L_a and the cross entropy loss L_x of the video captions, the results are much better than those of Step 1 and yet still worse than Step 2's. Hence, we conclude that 1) the attribute prediction branch helps the video captioning task and 2) the REINFORCE training is inevitable for eliminating the exposure mismatch [34] of the LSTMs between the training and testing stages.

If we remove the attribute prediction branch from our model and only use the REINFORCE+ to fine-tune the CNNs in Step 3, the results cannot be improved at all and even decrease if we use a larger learning rate. This verifies the necessity of the attribute prediction branch. Indeed, this branch back-propagate the gradients from an albeit different attribute prediction task directly to the CNNs, being able to complement the indirect gradients from the video captioning task and yet through the LSTMs.

5. Qualitative analysis of generated captions

In Table 5 and Table 6, a few video caption instances are generated on MSVD dataset and MSR-VTT dataset. The captions are generated by the S2VT model and our E2E model. We compare the sentences with the ground truth sentences in the Tables. Generally, our E2E model can generate relevant sentences. The sentences generated by our E2E model can reflect the visual content more faithfully with less grammar errors. For instance, our multitask model generates “a woman is frying meat” and it shows exactly what the woman is doing in the middle image of Table 5. It is more reasonable and relevant to the video content

than “a woman is putting some meat in a pan” generated by baseline model. Our E2E model also describe the event correctly, it recognizes there are a group of players playing soccer instead of one player in the right image of Table 5. On the other dataset, the examples also illustrate the correctness and faithfulness of our method. It can correctly detect drawing a cartoon compared to talking about a computer, the action of jumping on a trampoline and playing instead of talking about guitar. Obviously, our model can be more descriptive and more accurate.

6. Conclusion

We propose a novel method which combines the reinforcement learning with attribute prediction to train the whole framework end-to-end for video captioning. For our E2E model, it is a multitask end-to-end network and combines multisampling reinforce algorithm to generate captions. It is the first time that the CNNs are learned together with RNNs in video captioning and show much improvement, to best of our knowledge. The experiments on two standard video captioning datasets show our model can outperform the current methods. It also shows that the domain adopted video representation is more powerful than the generic image features. In the future, we will explore more representative video representations. As the 3D convolution methods are effective in the video classification, e.g I3D [57], we believe our model can also benefit from employing the effective video representation in video classification field. We may also explore other multitasks to better fine-tune the video representation.

References

- [1] Venugopalan, S., Rohrbach, M., Donahue, J., Mooney, R., Darrell, T., Saenko, K.: Sequence to sequence-video to text. In: Proceedings of the IEEE international conference on computer vision. (2015) 4534–4542
- [2] Anne Hendricks, L., Venugopalan, S., Rohrbach, M., Mooney, R., Saenko, K., Darrell, T.: Deep compositional captioning: Describing novel object categories without paired training data. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2016) 1–10
- [3] Hori, C., Hori, T., Lee, T.Y., Zhang, Z., Harsham, B., Hershey, J.R., Marks, T.K., Sumi, K.: Attention-based multimodal fusion for video description. In: Proceedings of the IEEE international conference on computer vision. (2017) 4193–4202
- [4] Baraldi, L., Grana, C., Cucchiara, R.: Hierarchical boundary-aware neural encoder for video captioning. arXiv preprint arXiv:1611.09312 (2016)

- [5] Chan, W., Jaitly, N., Le, Q., Vinyals, O.: Listen, attend and spell: A neural network for large vocabulary conversational speech recognition. In: Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on, IEEE (2016) 4960–4964
- [6] Toshniwal, S., Tang, H., Lu, L., Livescu, K.: Multitask learning with low-level auxiliary tasks for encoder-decoder based speech recognition. arXiv preprint arXiv:1704.01631 (2017)
- [7] Johnson, M., Schuster, M., Le, Q.V., Krikun, M., Wu, Y., Chen, Z., Thorat, N., Viégas, F., Wattenberg, M., Corrado, G., et al.: Google’s multilingual neural machine translation system: enabling zero-shot translation. arXiv preprint arXiv:1611.04558 (2016)
- [8] Chung, J., Cho, K., Bengio, Y.: A character-level decoder without explicit segmentation for neural machine translation. arXiv preprint arXiv:1603.06147 (2016)
- [9] Xu, K., Ba, J., Kiros, R., Cho, K., Courville, A., Salakhudinov, R., Zemel, R., Bengio, Y.: Show, attend and tell: Neural image caption generation with visual attention. In: International Conference on Machine Learning. (2015) 2048–2057
- [10] Liu, S., Zhu, Z., Ye, N., Guadarrama, S., Murphy, K.: Improved image captioning via policy gradient optimization of spider. In: Proceedings of the IEEE international conference on computer vision. (2017) 873–881
- [11] Venugopalan, S., Hendricks, L.A., Mooney, R., Saenko, K.: Improving lstm-based video description with linguistic knowledge mined from text. arXiv preprint arXiv:1604.01729 (2016)
- [12] Pan, P., Xu, Z., Yang, Y., Wu, F., Zhuang, Y.: Hierarchical recurrent neural encoder for video representation with application to captioning. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2016) 1029–1038
- [13] Yao, T., Pan, Y., Li, Y., Qiu, Z., Mei, T.: Boosting image captioning with attributes. OpenReview **2**(5) (2016) 8
- [14] Venugopalan, S., Xu, H., Donahue, J., Rohrbach, M., Mooney, R., Saenko, K.: Translating videos to natural language using deep recurrent neural networks. arXiv preprint arXiv:1412.4729 (2014)
- [15] Rumelhart, D.E., Hinton, G.E., Williams, R.J.: Learning internal representations by error propagation. Technical report, California Univ San Diego La Jolla Inst for Cognitive Science (1985)
- [16] Werbos, P.J.: Generalization of backpropagation with application to a recurrent gas market model. Neural networks **1**(4) (1988) 339–356
- [17] Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural computation **9**(8) (1997) 1735–1780
- [18] Pan, Y., Yao, T., Li, H., Mei, T.: Video captioning with transferred semantic attributes. In: Proceedings of the IEEE conference on computer vision and pattern recognition. (2017) 6504–6512
- [19] Yu, Y., Ko, H., Choi, J., Kim, G.: End-to-end concept word detection for video captioning, retrieval, and question answering. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2017) 3165–3173
- [20] Chen, D.L., Dolan, W.B.: Collecting highly parallel data for paraphrase evaluation. In: Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1, Association for Computational Linguistics (2011) 190–200
- [21] Xu, J., Mei, T., Yao, T., Rui, Y.: Msr-vtt: A large video description dataset for bridging video and language. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2016) 5288–5296
- [22] Bahdanau, D., Cho, K., Bengio, Y.: Neural machine translation by jointly learning to align and translate. arXiv preprint arXiv:1409.0473 (2014)
- [23] Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., Bengio, Y.: Learning phrase representations using rnn encoder-decoder for statistical machine translation. arXiv preprint arXiv:1406.1078 (2014)
- [24] Sutskever, I., Vinyals, O., Le, Q.V.: Sequence to sequence learning with neural networks. In: Advances in neural information processing systems. (2014) 3104–3112
- [25] You, Q., Jin, H., Wang, Z., Fang, C., Luo, J.: Image captioning with semantic attention. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2016) 4651–4659
- [26] Chen, L., Zhang, H., Xiao, J., Nie, L., Shao, J., Chua, T.S.: Sca-cnn: Spatial and channel-wise attention in convolutional networks for image captioning. arXiv preprint arXiv:1611.05594 (2016)
- [27] Yao, L., Torabi, A., Cho, K., Ballas, N., Pal, C., Larochelle, H., Courville, A.: Describing videos

- by exploiting temporal structure. In: Proceedings of the IEEE international conference on computer vision. (2015) 4507–4515
- [28] Yu, H., Wang, J., Huang, Z., Yang, Y., Xu, W.: Video paragraph captioning using hierarchical recurrent neural networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition. (2016) 4584–4593
- [29] Pu, Y., Min, M.R., Gan, Z., Carin, L.: Adaptive feature abstraction for translating video to language. arXiv preprint arXiv:1611.07837 (2016)
- [30] Zhu, L., Xu, Z., Yang, Y.: Bidirectional multirate reconstruction for temporal modeling in videos. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2017) 2653–2662
- [31] Gan, Z., Gan, C., He, X., Pu, Y., Tran, K., Gao, J., Carin, L., Deng, L.: Semantic compositional networks for visual captioning. In: Proceedings of the IEEE conference on computer vision and pattern recognition. (2017) 5630–5639
- [32] Donahue, J., Anne Hendricks, L., Guadarrama, S., Rohrbach, M., Venugopalan, S., Saenko, K., Darrell, T.: Long-term recurrent convolutional networks for visual recognition and description. In: Proceedings of the IEEE conference on computer vision and pattern recognition. (2015) 2625–2634
- [33] Shen, Z., Li, J., Su, Z., Li, M., Chen, Y., Jiang, Y.G., Xue, X.: Weakly supervised dense video captioning. arXiv preprint arXiv:1704.01502 (2017)
- [34] Bengio, S., Vinyals, O., Jaitly, N., Shazeer, N.: Scheduled sampling for sequence prediction with recurrent neural networks. In: Advances in Neural Information Processing Systems. (2015) 1171–1179
- [35] Papineni, K., Roukos, S., Ward, T., Zhu, W.J.: Bleu: a method for automatic evaluation of machine translation. In: Proceedings of the 40th annual meeting on association for computational linguistics, Association for Computational Linguistics (2002) 311–318
- [36] Banerjee, S., Lavie, A.: Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In: Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization. Volume 29. (2005) 65–72
- [37] Vedantam, R., Lawrence Zitnick, C., Parikh, D.: Cider: Consensus-based image description evaluation. In: Proceedings of the IEEE conference on computer vision and pattern recognition. (2015) 4566–4575
- [38] Lin, C.Y.: Rouge: A package for automatic evaluation of summaries. In: Text summarization branches out: Proceedings of the ACL-04 workshop. Volume 8., Barcelona, Spain (2004)
- [39] Ren, Z., Wang, X., Zhang, N., Lv, X., Li, L.J.: Deep reinforcement learning-based image captioning with embedding reward. arXiv preprint arXiv:1704.03899 (2017)
- [40] Dai, B., Lin, D., Urtasun, R., Fidler, S.: Towards diverse and natural image descriptions via a conditional gan. arXiv preprint arXiv:1703.06029 (2017)
- [41] Wang, X., Zhang, C., Zhang, Z.: Boosted multi-task learning for face verification with applications to web image and video search. In: computer vision and pattern recognition, 2009. CVPR 2009. IEEE conference on, IEEE (2009) 142–149
- [42] Yuan, X.T., Liu, X., Yan, S.: Visual classification with multitask joint sparse representation. IEEE Transactions on Image Processing **21**(10) (2012) 4349–4360
- [43] Yan, Y., Ricci, E., Subramanian, R., Liu, G., Lanz, O., Sebe, N.: A multi-task learning framework for head pose estimation under target motion. IEEE transactions on pattern analysis and machine intelligence **38**(6) (2016) 1070–1083
- [44] Gebru, T., Hoffman, J., Fei-Fei, L.: Fine-grained recognition in the wild: A multi-task domain adaptation approach. arXiv preprint arXiv:1709.02476 (2017)
- [45] Collobert, R., Weston, J.: A unified architecture for natural language processing: Deep neural networks with multitask learning. In: Proceedings of the 25th international conference on Machine learning, ACM (2008) 160–167
- [46] Szegedy, C., Ioffe, S., Vanhoucke, V., Alemi, A.A.: Inception-v4, inception-resnet and the impact of residual connections on learning. In: AAAI. (2017) 4278–4284
- [47] Williams, R.J.: Simple statistical gradient-following algorithms for connectionist reinforcement learning. Machine learning **8**(3-4) (1992) 229–256
- [48] Rennie, S.J., Marcheret, E., Mroueh, Y., Ross, J., Goel, V.: Self-critical sequence training for image captioning. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR). (July 2017)
- [49] He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of

the IEEE conference on computer vision and pattern recognition. (2016) 770–778

- [50] Srivastava, R.K., Greff, K., Schmidhuber, J.: Highway networks. arXiv preprint arXiv:1505.00387 (2015)
- [51] Wu, Q., Shen, C., Liu, L., Dick, A., van den Hengel, A.: What value do explicit high level concepts have in vision to language problems? In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2016) 203–212
- [52] Zhang, X., Gao, K., Zhang, Y., Zhang, D., Li, J., Tian, Q.: Task-driven dynamic fusion: Reducing ambiguity in video description. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2017) 3713–3721
- [53] Jin, Q., Chen, J., Chen, S., Xiong, Y., Hauptmann, A.: Describing videos using multi-modal fusion. In: Proceedings of the 2016 ACM on Multimedia Conference, ACM (2016) 1087–1091
- [54] Shetty, R., Laaksonen, J.: Frame-and segment-level features and candidate pool evaluation for video caption generation. In: Proceedings of the 2016 ACM on Multimedia Conference, ACM (2016) 1073–1076
- [55] Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., Devin, M., Ghemawat, S., Irving, G., Isard, M., et al.: Tensorflow: A system for large-scale machine learning. In: OSDI. Volume 16. (2016) 265–283
- [56] Chen, X., Fang, H., Lin, T.Y., Vedantam, R., Gupta, S., Dollár, P., Zitnick, C.L.: Microsoft coco captions: Data collection and evaluation server. arXiv preprint arXiv:1504.00325 (2015)
- [57] Carreira, J., Zisserman, A.: Quo vadis, action recognition? a new model and the kinetics dataset. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE (2017) 4724–4733