Move Forward and Tell: A Progressive Generator of Video Descriptions

Yilei Xiong, Bo Dai, and Dahua Lin

CUHK-SenseTime Joint Lab, The Chinese University of Hong Kong
{xy014,db014,dhlin}@ie.cuhk.edu.hk

Abstract. We present an efficient framework that can generate a coherent paragraph to describe a given video. Previous works on video captioning usually focus on video clips. They typically treat an entire video as a whole and generate the caption conditioned on a single embedding. On the contrary, we consider videos with rich temporal structures and aim to generate paragraph descriptions that can preserve the story flow while being coherent and concise. Towards this goal, we propose a new approach, which produces a descriptive paragraph by assembling temporally localized descriptions. Given a video, it selects a sequence of distinctive clips and generates sentences thereon in a coherent manner. Particularly, the selection of clips and the production of sentences are done jointly and progressively driven by a recurrent network – what to describe next depends on what have been said before. Here, the recurrent network is learned via self-critical sequence training with both sentence-level and paragraph-level rewards. On the ActivityNet Captions dataset, our method demonstrated the capability of generating high-quality paragraph descriptions for videos. Compared to those by other methods, the descriptions produced by our method are often more relevant, more coherent, and more concise.

Keywords: Video Captioning · Move Forward and Tell · Recurrent Network · Reinforcement Learning · Repetition Evaluation

1 Introduction

Textual descriptions are an important way to characterize images and videos. Compared to class labels or semantic tags, descriptions are usually more informative and distinctive. In recent years, image captioning, a task to generate short descriptions for given images, becomes an active research topic [1, 8, 25, 30] and has seen remarkable progress thanks to the wide adoption of recurrent neural networks. However, how to extend the captioning techniques to describe videos remains an open question.

Over the past several years, various methods have been proposed for generating video descriptions. Early efforts [28] simply extend the encoder-decoder paradigm in image captioning to videos. Such methods follow a similar pipeline, namely embedding an entire video into a feature vector, feeding it to a decoding

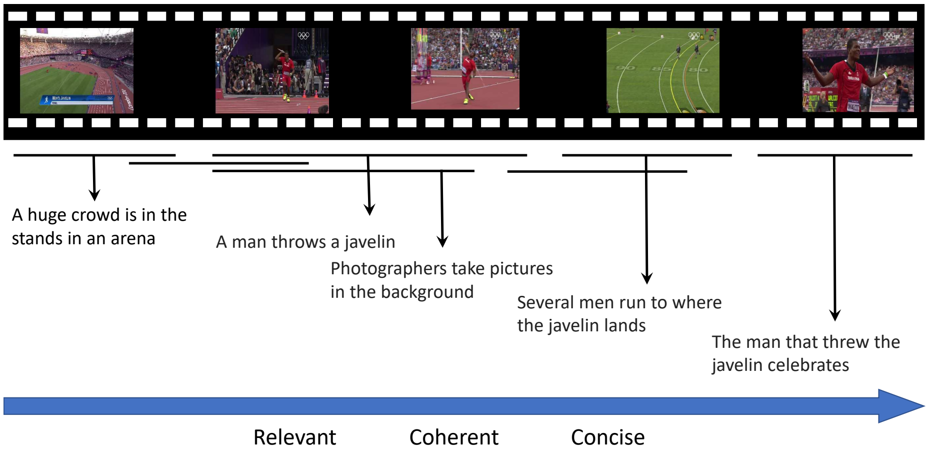
A huge crowd is in the stands in an arena
A man throws a javelin
Photographers take pictures in the background
Several men run to where the javelin lands
The man that threw the javelin celebrates
Relevant
Coherent
Concise

Fig. 1. As shown in this figure, our framework localizes important events in a video, and picks a sequence of coherent and independent events, on which generates a coherent and concise descriptive paragraph for the video.

network to obtain a descriptive sentence. However, for a video with rich temporal structures, a single sentence is often not enough to capture all important aspects of the underlying events. Recently, a new stream of efforts emerge [10,20], which attempt to use multiple sentences to cover a video. Whereas such methods can provide more complete characterization of a video, they are still subject to various issues, *e.g.* lack of coherence among sentences and high redundancy. These issues, to a large extent, are ascribed to two reasons: (1) failing to align the temporal structure of the given video with the narrative structure of the generated description; and (2) neglecting the dependencies among sentences.

In this work, we aim to develop a new framework for generating *paragraph* descriptions for videos with rich temporal structures. The goal is to generate descriptions that are *relevant*, *coherent*, and *concise*. According to the discussion above, the key to achieving this goal lies in two aspects: (1) associate the temporal structures in the given video with the linguistic generation process; and (2) encourage coherence among sentences in an effective way.

Specifically, our approach is based on two key observations. First, a natural video is usually composed of multiple short and meaningful segments that reflect a certain step in a procedure or an episode of a story. We refer to such video segments as *events*. While a single sentence may not be enough to describe a long video, it often suffices to characterize an individual event. Second, when people describe a video with a paragraph, there exist strong logical and linguistic relations among consecutive sentences. What to describe in a sentence depends strongly on what have been said.

Inspired by these observations, we devise a two-stage framework. This framework first localizes candidate events from the video through video action detection. On top of these candidates, the framework then generate a coherent paragraph in a *progressive* manner. At each step, it selects the next event to describe and produces a sentence therefor, both conditioned on what have been said before. The progressive generation process is driven by a variant of LSTM network that takes into account both temporal and linguistic structures. To effectively learn this network, we adopt the *self-critical sequence training* method and introduce rewards in two different levels, namely the sentence level and the paragraph level. On ActivityNet Captions [10], the proposed framework outperforms previous ones under multiple metrics. Qualitatively, the descriptions produced by our method are generally more relevant, more coherent, and more concise.

The key contribution of this work lies in a new framework for generating descriptions for given videos. This framework is distinguished from previous ones in three key aspects: (1) It aligns the temporal structure of the given video and the narrative structure of the generated description via a recurrent network. (2) It maintains coherence among the sentences in a paragraph by explicitly conditioning *what to say next* on both the temporal structures and *what have been said*. (3) It is learned via reinforcement learning, guided by rewards in both the sentence level and the paragraph level.

2 Related Work

Image Captioning Early attempts of image captioning rely on visual concept detection, followed by templates filling [11] or nearest neighbour retrieving [6]. Recently, Vinyals *et al* [24] proposed the encoder-decoder paradigm, which extracts image features using a CNN, followed by an RNN as the decoder to generate captions based on the features. This model outperforms classical methods and becomes the backbone of state-of-the-art captioning models. Many variants [2,3,27] are proposed following the encoder-decoder paradigm, For example, Xu *et al* [27] improved it by introducing an attention mechanism to guide the decoding process.

While many methods for image captioning can be seamlessly converted into methods for video captioning, video contains richer semantic content that spreads on the temporal dimension, directly applying methods for image captioning often lead to the loss of temporal information.

Video Dense Captioning Video dense captioning, is a topic that closely related to video captioning, where it densely generates multiple sentences, covering different time spans of the input video. Specifically, Krishna *et al* [10] proposed a method that obtains a series of proposals from the input video and uses a captioning model to generate a sentence for each, where the temporal relationships among the proposals are taken into account. On the other hand, Shen *et al* [20] proposed a weakly-supervised method, which uses multi-instance multi-label learning to detect words from the input video and then uses these words to select spatial regions to form region sequences. Finally, it employs a sequence-to-sequence submodule to transfer region sequences into captions.

Although closely related, video dense captioning is different from video captioning. Particularly, a model for video dense captioning could generate multiple captions, each covers a small period of the input video, where the periods can be overlapped with each other, leading to a lot of redundancy in the corresponding captions. On the contrary, a model for video captioning should generate a *single* description consisting of several *coherent* sentences for the *entire* input video.

Video Captioning Our method targets the topic of video captioning. Related works can be roughly divided into two categories based on whether a single sentence or a paragraph is generated for each input video. In the first category, a single sentence is generated. Among all the works in this category, Rohrbach *et al* [18] detected a set of visual concepts at first, including verbs, objects and places, and then applied an LSTM net to fuse these concepts into a caption. Yu *et al* [30] and Pan *et al* [13] followed a similar way, but respectively using a semantic attention model and a transfer unit to select detected concepts and generate a caption. Instead of relying on visual concepts, Hori *et al* [8] and Venugopalan *et al* [23] use features from multiple sources including appearance and motion to improve quality of the generated caption. There are also efforts devoted to improving the decoder side. Wang *et al* [25] added a memory network before the LSTM net during the decoding process to share features at different timestamps. Baraldi *et al* [1] applied a boundary detecting module to share features hierarchically. While they are able to produce great captions, a single sentence is difficult to capture all semantic information in a video, as one video usually contains several distintive events.

The second category is to generate a paragraph to describe a video. Our method belongs to this category. In this category, Yu *et al* [29] applied hierarchical recurrent neural networks, where a sentence generator is used to generate a single sentence according to a specific *topic*, and a paragraph generator is used to capture inter-sentence statistics and feed the sentence generator with a series of topics. The most similar work to our method is the one presented in [19]. This method first select a subset of clips from the input video, and then use a decoder to generate sentence from these clips to form a paragraph summary of the entire video. Our method is different from these existing works from two aspects. (1) When generating each sentence of the paragraph, the method in [29] requires the features from the entire video, which is expensive for very long videos, while our method only requires features in selected proposals. (2) In [19], the clips are selected according to frame quality *in advance* as a preprocessing step, without taking into account the coherence of narration. This way will lead to redundancy in the resulting paragraph. On the contrary, our method selects key events *along with* the generation of captions in a progressive way. The selection of the next key event depends on what has been said before in preceding captions. Also, this process takes into account temporal and semantic relationships among the selected events in order to ensure the coherence of the resultant paragraph.

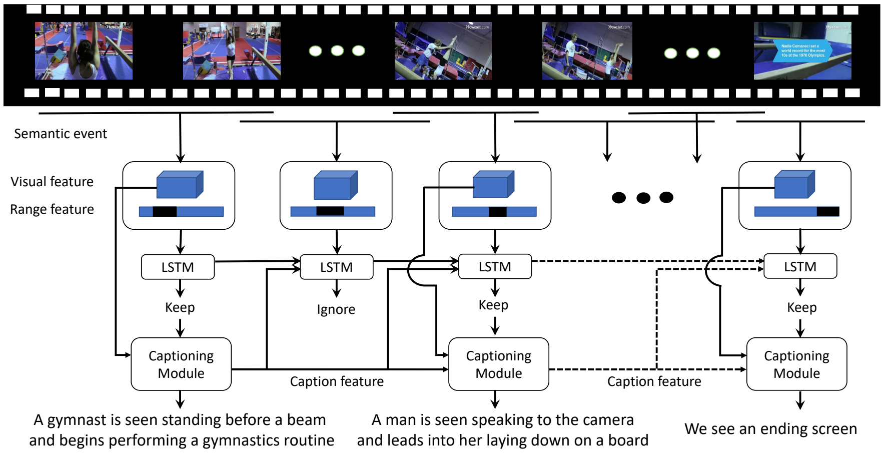

Fig. 2. An overview of our framework, which at first localize important events from the entire video. It then generates a coherent and concise descriptive paragraph upon these localized events. Specifically, an LSTM net, serves as a selection module, will pick out a sequence of coherent and semantically independent events, based on appearances, temporal locations of events, as well as their semantic relationships. And based on this selected sequence, another LSTM net, serves as a captioning module, will generate a single sentence for each event in the sequence conditioned on previous generated sentences, which are then concatenated sequentially as the output of our framework.

3 Generation Framework

Our task is to develop a framework that can generate coherent paragraphs to describe given videos. Specifically, a good description should possess three properties: (1) *Relevant*: the narration aligns well with the events in their temporal order. (2) *Coherent*: the sentences are organized into a logical and fluent narrative flow. (3) *Concise*: each sentence conveys a distinctive message, without repeating what has been said.

3.1 Overview

A natural video often comprises multiple events that are located sparsely along the temporal range. Here, *events* refer to those video segments that contain distinctive semantics that need to be conveyed. Taking the entire video as input for generating the description is inefficient, and is likely to obscure key messages when facing numerous noisy clips. Therefore, we propose a framework as shown in Figure 2, which generates a descriptive paragraph in two stages, namely *event localization* and *paragraph generation*. In event localization, we localize *candidate events* in the video with a high recall. In paragraph generation, we first filter out redundant or trivial candidates, so as to get a sequence of important and distinctive events. We then use this sequence to generate a single descriptive paragraph for the entire video in a progressive manner, taking into account the coherence among sentences.

3.2 Event Localization

To localize event candidates, we adopted the clip proposal generation scheme presented in [31], using the released codes. This scheme was shown to be effective in locating important clips from untrimmed videos with high accuracy. Specifically, following [31], we calculate frame-wise importance scores, and then group the frames into clips of via a watershed procedure. This method outputs a collection of clips as *event candidates*. These clips have varying durations and can overlap with each other. As our focus is on paragraph generation, we refer readers to [31] for more details of event localization.

Note that not all the candidates derived in this stage is worthy of description. The paragraph generation will select a subset of candidates that contain important and distinctive messages, *along with* the generation process.

3.3 Progressive Event Selection and Captioning

Given a sequence of events, there are various ways to generate a descriptive paragraph. While the most straightforward way is to generate a single sentence for each event in the sequence, it is very likely to introduce a large amount of redundancy. To generate coherent and concise description, we can select a subset of distinctive events and generate sentences thereon. The key challenge here is to stride a good balance between sufficient coverage and conciseness.

In this work, we develop a *progressive* generation framework that couples two recurrent networks, one for event selection and the other for caption generation.

Event Selection With all event candidates arranged in the chronological order, denoted as $(e_1, \dots, e_T)$, the event selection network begins with the first candidate in the sequence and moves forward gradually as follows:

$$\mathbf{h}_0 = \mathbf{0}, \quad \mathbf{h}_t = \text{LSTM}(\mathbf{h}_{t-1}, \mathbf{v}_t, \mathbf{r}_t, \mathbf{c}_{k_t}), \tag{1}$$

$$p_t = \text{sigmoid}(\mathbf{w}_p^T \mathbf{h}_t), \quad y_t = 1[p_t > \delta]. \tag{2}$$

Specifically, it initializes the latent state vector $\mathbf{h}_0$ to be zeros. At each step t, it updates the latent state $\mathbf{h}_t$ with an LSTM cell and computes p_t, the probability of e_t containing relevant and distinctive information, by applying a sigmoid function to $\mathbf{w}_p^T \mathbf{h}_t$. If p_t is above a threshold δ, y_t would be set to 1, indicating that the candidate e_t will be selected for sentence generation.

The updating of $\mathbf{h}_t$ depends on four different inputs: (1) $\mathbf{h}_{t-1}$: the latent state at the preceding step. (2) $\mathbf{v}_t$: the visual feature of e_t, extracted using a Temporal Segmental Network (TSN) [26]. (3) $\mathbf{r}_t$: the range feature, similar as the image mask in [4], represented by a binary mask that indicates the normalized time span of e_t relative to the entire duration. (4) $\mathbf{c}_{k_t}$: the caption feature of e_{k_t}, where k_t is the index of the *last* selected event candidate (before t). Here, the caption feature is from the *caption generation network*, which we will present next. Particularly, $\mathbf{c}_{k_t}$ is chosen to be the latent state of the caption generation network at the final decoding step when generating the description for e_{k_t}.

With the previous caption feature $\mathbf{c}_{k_t}$ incorporated, the event selection network is aware of what have been said in the past when making a selection. This allows it to avoid selecting the candidates that are semantically redundant.

Caption Generation On top of the selected events, the *caption generation network* will produce a sequence of sentences, one for each event, as follows:

$$\mathbf{g}_0^{(k)} = \begin{cases} \mathbf{0}, & k = 1 \\ \mathbf{g}_*^{(k-1)}, & k > 1 \end{cases}, \quad \mathbf{g}_l^{(k)} = \mathrm{LSTM}(\mathbf{g}_{l-1}^{(k)}, \mathbf{u}_l^{(k)}, w_{l-1}^{(k)}), \tag{3}$$

$$\mathbf{s}_l^{(k)} = \mathbf{W}_s \mathbf{g}_l^{(k)}, \quad w_l^{(k)} \sim \mathrm{softmax}(\mathbf{s}_l^{(k)}). \tag{4}$$

Here, $\mathbf{g}_l^{(k)}$ denotes the latent state at the l-th step of the caption generation network when describing the k-th selected event. $\mathbf{u}_l^{(k)}$ denotes a visual feature of a subregion of the event. Here, the computation of $\mathbf{u}_l^{(k)}$ follows the scheme presented in [16], which allows the network to dynamically attend to different subregions as it proceeds[1]. $w_l^{(k)}$ is the word produced at the l-th step, which is sampled from softmax($\mathbf{s}_l^{(k)}$).

This network is similar to a standard LSTM for image captioning except for an important difference: When $k > 1$, the latent state is initialized to $g_*^{(k-1)}$, the latent state at the last decoding step while generating the previous sentence. This means that the generation of each sentence (except the first one) is conditioned on the preceding one, which allows the generation to take into account the coherence among sentences.

Discussions The event selection network and the caption generation network work hand in hand with each other when generating a description for a given video. On one hand, the selection of next event candidate depends on what has been said. Particularly, one input to the event selection network is $\mathbf{c}_{k_t}$, which is set to the $\mathbf{g}_*^{(k_t)}$, the last latent state of the caption generation network in generating the previous sentence. On the other hand, the caption generation network is invoked only when the event selection network outputs $y_t = 1$, and the generation of the current sentence depends on those that come before. The collaboration between both networks allows the framework to produce paragraphs that can cover major messages, while being coherent and concise.

Note that one may also use Non-Maximum Suppression (NMS) to directly remove temporal overlapped events. This simple way is limited compared to ours, as it only considers temporal overlap while ignoring the semantic relevance.

[1] We provide more details of the computation of $\mathbf{u}_l^{(k)}$ in the supplemental materials

Another way is to first generate sentences for all events and then select a subset of important ones based on text summarization [19]. This approach, however, does not provide a mechanism to encourage linguistic coherence among sentences, which is crucial for generating high-quality descriptions.

4 Training

Three modules in our framework need to be trained, namely *event localization*, *caption generation*, and *event selection*. In particular, we train the event localization module simply following the procedure presented in [31]. The other two modules, the caption generation network and the event selection network, are trained separately. We first train the caption generation network using the ground-truth event captions. Thereon, we then train the event selection network, which requires the caption generation states as input.

4.1 Training Caption Generation Network

The caption generation network models the distribution of each word conditioned on previous ones and other inputs, including the visual features of the corresponding event $\mathbf{u}^{(k)}$ and the final latent state of for the preceding sentence $\mathbf{g}_*^{(k-1)}$. Hence, this distribution can be expressed as $p_{\boldsymbol{\theta}}(w_l | w_{1:l-1}; \mathbf{u}^{(k)}, \mathbf{g}_*^{(k-1)})$, where $\boldsymbol{\theta}$ denotes the network parameters. We train this network through two stages: (1) initial supervised training, and (2) reinforcement learning.

The initial supervised training is performed based on pairs of events and their corresponding ground-truth descriptions, with the standard cross entropy loss. Note that this network requires $\mathbf{g}_*^{(k-1)}$ as input, which is provided *on the fly* during training. In particular, we feed the ground-truth sentences for each video one by one. At each iteration, we cache the final latent state for the current sentence and use it as an input for the next one.

Supervised training encourages the caption generation network to emulate the training sentences word by word. To further improve the quality of the resultant sentences, we resort to reinforcement learning. In this work, we employ the *Self-Critical Sequence Training (SCST)* [16] technique. Particularly, we consider the caption generation network as an *"agent"*, and choosing a word an *"action"*. Following the practice in [16], we update the network parameters using approximated policy gradient in the reinforcement learning stage.

The key to reinforcement learning is the design of the rewards. In our design, we provide *rewards* at two levels, namely the *sentence-level* and the *paragraph-level*. As mentioned, the network takes in the ground-truth events of a video sequentially, producing one sentence for each event (conditioned on the previous state). These sentences together form a paragraph. When a sentence is generated, it receives a sentence-level reward. When the entire paragraph is completed, it receives a paragraph-level reward. The reward is defined to be the CIDEr [22] metric between the generated sentence/paragraph and the ground truth.

4.2 Training Event Selection Network

The event selection network is a recurrent network that takes a sequence of candidate events as input and produces a sequence of binary indicators (for selecting a subset of candidates to retain). We train this network in a supervised manner. Here, the key question is how to obtain the training samples. We accomplish this in two steps: (1) labeling and (2) generating training sequences.

First, for each video, we use the event localization module to produce a series of event candidates as $(e_1, \ldots, e_T)$. At the same time, we also have a set of ground-truth events provided by the training set, denoted as $(e^*_1, \ldots, e^*_{T^*})$. For each ground-truth event e^*_j, we find the candidate e_i that has the highest overlap with it, in terms of the temporal IoU, and label it as positive, *i.e.* setting $y_i = 1$. All the other event candidates are labeled as negative.

Second, to generate training sequences, we consider three different ways:

- *(S1) Complete sequences*, which simply uses the whole sequence of candidates for every video, *i.e.* $(e_1, e_2, \cdots, e_m)$.
- *(S2) Subsampling at intervals*, which samples event candidates at varying intervals, *e.g.* $(e_2, e_4, \cdots, e_m)$, to obtain a larger set of sequences.
- *(S3) Subsampling negatives*, which keeps all positive candidates, while randomly sampling the same number of negative candidates in between.

Note that the positive and negative candidates are highly imbalanced. For each video, positive candidates are sparsely located, while negative ones are abundant. The scheme (S3) explicitly rebalances their numbers. Our experiment shows that (S3) often yields the best performance.

The event selection network is trained with the help of the caption generation network. To be more specific, whenever the event selection network yields a positive prediction, the caption feature $\mathbf{c}_{k_t}$ will be updated based on the caption generation network, which will be fed as an input to the next recurrent step.

5 Experiment

We report our experiments on ActivityNet Captions [10], where we compared the proposed framework to various baselines, and conducted ablation studies to investigate its characteristics.

5.1 Experiment Settings

The ActivityNet Captions dataset [10] is the largest publicly available dataset for video captioning. This dataset contains $10,009$ videos for training, and $4,917$ for validation. Compared to previous datasets [7, 15, 17], it has two orders of magnitude more videos. The videos in this dataset are 3 minutes long on average. Each video in the training set has one set of human labeled annotations, and each video in the validation set has two such sets. Here, a set of annotations is a series of sentences, each aligned with a long or short segment in the video.

About 10% of all segments have overlaps with each other. On average, each set of annotations contains 3.65 sentences.

While ActivityNet Captions was originally designed for the task of video dense captioning, we adapt it to our task with the ground-truth paragraphs derived by sequentially concatenating the sentences within each set of segment-based annotations. As a result, for each video in the training set, there is one ground-truth paragraph, and for each video in the validation set, there are two ground-truth paragraphs. Since the annotations for testing videos are not publicly accessible, we randomly split the validation set in half, resulting in $2,458$ videos for tuning hyperparameters, and $2,459$ videos for performance evaluation.

We set $\delta = 0.3$, $L = 100$ in the event selection module, and $N = 10$ in the captioning module. We separately train the three modules in our framework. In particular, the event localization module is trained according to [31], where the localized events have a recall of 63.77% at 0.7 tIoU threshold. For captioning module, the LSTM hidden size is fixed to 512. As discussed in section 4, we first train the model under the cross-entropy objective using ADAM [9] with an initial learning rate of 4×10^{-4}. We choose the model with best CIDEr score on the validation set. We then run SCST training initialized with this model. The reward metric is CIDEr, for both sentence and paragraph reward. SCST training uses ADAM with a learning rate 5×10^{-5}. One batch contains at least 80 events, since events in the same video are fed into a batch simultaneously. For the event selection module, we train it on a collection of labeled training sequences prepared as described in Section 4, where each training sequence contains 64 candidate events. We use cross-entropy as the loss function and SGD with momentum as the optimizer. The learning rate is initialized to 0.1 and scaled down by a factor of 0.1 every $10,000$ iterations. We set the SGD momentum to 0.9, weight decay to 0.0005, and batch size to 80. [2]

5.2 Evaluation

We evaluate the performance using multiple metrics, including BLEU [14], METEOR [5], CIDEr [22] and Rouge-L [12]. Besides, we notice there exists a general problem for video captioning results, *i.e.*, repetition or redundancy. This may be due to that captioning module can't distinguish detailed events. For example, the captioner may take both ironing collar and ironing sleeve as ironing. Describing repeated things definitely hurts the coherence of the descriptions. However this can not reflect in the above metrics.

To measure this effect, a recent work proposes Self-BLEU [32] by evaluating how one sentence resembles the rest a generated paragraph. We also propose another metric, called Repetition Evaluation (RE). Given a generating description c_i, the number of time an n-gram w_k occurs in it is denoted as $h_k(c_i)$. The Repetition Evaluation computes a redundancy score for every description c_i:

$$RE(c_i) = \frac{\sum_k \max\left(h_k(c_i) - 1, 0\right)}{\sum_k h_k(c_i)}, \tag{5}$$

[2] Code will be made publicly available soon

where the gram length, n, takes a large number, like 4 in our experiments. The corpus-level score is the mean score across all descriptions. Ideally, a description with n repetitions would get a score around $(n-1)/n$.

5.3 Comparison with Other Methods

We compared our framework with various baselines, which we describe as below.

(1) *Sentence-Concat*: a simple baseline that equally splits a video into four disjoint parts, and describes each part with a single sentence using the captioning model as ours. The final paragraph is derived by concatenating these four sentences. Using this baseline, we are able to study the effect of localizing events in the input video.

(2) *Hierarchical-RNN* [29]: a more sophisticated way to generate paragraphs from a video, where a topic RNN generates a sequence of topics to control the generation of each individual sentence. With the topic embeddings as input, a sentence RNN generates a sentence for each topic in the sequence.

(3) *Dense-Caption* [10]: one of the state-of-the-art methods for video dense captioning, which generates a single sentence for each candidate event and then concatenate them all into a paragraph, regardless of their similarities. This baseline is used to demonstrate the differences between *video dense captioning* and *video captioning*.

(4) *Dense-Caption-NMS*: a method based on the above Dense-Caption. It selects events from candidate events of Dense-Caption using Non-Maximum Suppression (NMS), removing those that are highly overlapped with others on temporal range.

(5) *Semantic-Sum* [19]: a recent method that also identifies the video segments as ours. We find that this method gets best performance when setting sentence length as 3 and using Latent Semantic Analysis [21] in summarization module.

(6) *Move Forward and Tell (MFT)*: our proposed framework which progressively select events and produce sentences conditioned on what have been said before.

(7) *GT-Event*: this baseline directly applies our captioning module on *ground-truth events*. In principle, this should serve as a performance upper bound, as it has access to the ground-truth event locations.

Table 1 presents the results for different methods on ActivityNet Captions, from which we have the following observations: (1) *Dense-Caption* performs very poorly since there are many redundant proposals and therefore repeated sentences indicating by RE and Self-BLEU metrics. The RE score is terribly high, meaning two thirds of the descriptions are likely redundant. This clearly shows the differences between video dense captioning and video captioning. (2) *Sentence-Concat* and *Dense-Caption-NMS* are at a comparable level, much better than *Dense-Caption*. These two methods may benefit from a common aspect, *i.e.* their events are almost not overlapped. But some important events may not be localized here. (3) *Hierarchical-RNN* improves the result to the next level, achieving 25.53% Rouge-L. This suggests that it produces more coherent

Table 1. Comparison results of video captioning. The last method GT-Event uses ground truth information and can be seen as the upper bound of event selecting. The value is in %. For metrics RE and Self-BLEU, the lower is better. For the others, the higher is better

Model	CIDEr	BLEU@4	BLEU@3	BLEU@2	BLEU@1	Rouge-L	METEOR	RE	Self-BLEU
Sentence-Concat	4.51	4.18	6.45	10.41	17.52	22.59	8.79	18.41	49.40
Hierarchical-RNN [29]	6.99	7.32	12.13	21.23	39.02	25.53	10.79	18.79	54.38
Dense-Caption [10]	0.29	0.99	1.51	2.33	3.52	8.63	6.75	64.05	89.79
Dense-Caption-NMS	3.52	4.45	6.96	11.21	18.09	21.41	12.08	23.98	62.46
Semantic-Sum [19]	10.43	6.44	11.21	20.36	37.22	25.44	12.66	29.94	67.49
MFT (Ours)	**14.15**	**8.45**	**13.52**	**22.26**	**39.11**	**25.88**	**14.75**	**17.59**	**45.80**
GT-Event	19.56	10.33	16.44	27.24	46.77	29.70	15.09	15.88	42.95

Table 2. This table compares different training schemes for our model. See section 4 for the detail of schemes

Training Scheme	CIDEr	BLEU@4	BLEU@3	BLEU@2	BLEU@1	Rouge-L	METEOR	RE	Self-BLEU
(S1) Complete sequenses	9.25	7.12	11.56	19.18	33.80	23.84	11.06	26.11	51.41
(S2) Subsampling at intervals	8.04	6.49	10.52	17.58	31.16	21.27	10.16	24.09	46.19
(P1) Supervised training only	12.81	**8.53**	13.25	21.76	37.68	25.53	13.24	19.10	46.77
(P1 + P2 & S3) Ours	**14.15**	8.45	**13.52**	**22.26**	**39.11**	**25.88**	**14.75**	**17.59**	**45.80**

result by modelling the sentence relationships. (4) *Semantic-Sum* further improves the result, achieving 10.43% CIDEr. This shows the effect of localizing video events for captioning. (5) Our method *MFT* significantly outperforms all above methods, *e.g.* it attains 14.15% for CIDEr, compared to 10.43% from *Semantic-Sum*. It also performs comparably well with *GT-Event*, a method utilizing ground-truth events.

5.4 Ablation Studies

Training Scheme In section 4, we introduce our training strategies. For caption generation network, the training includes two phases – (P1) *initial supervised training* and (P2) *reinforcement learning*. For event selection network, we propose three different ways to generate training sequences. (S1) *complete sequences*, (S2) *subsampling at intervals* and (S3) *subsampling negatives* where our model adopts scheme (S3).

Table 2 shows the performance under different training schemes. First of all, among all three schemes of generating training sequences for event selection network, (S3) *subsampling negatives* performs best. (S1) scheme only gets 9.25% CIDEr while (S2) scheme gets 8.04% CIDEr, a considerable drop from our model. This indicates the importance of balancing the positive candidates and negative candidates. Besides, by utilizing the *reinforcement learning* for caption generation network, our model gains performance improvement, especially for the target metric, CIDEr. Our model yields 14.15% CIDEr score while only using *supervised training* yields 12.81%. Other metrics get consistent improvement with reinforcement learning, including RE and Self-BLEU metrics.

Table 3. This table lists the results of using different feature combinations for event selection, where the full combination is shown to be the best configuration

Feature	CIDEr	BLEU@4	BLEU@3	BLEU@2	BLEU@1	Rouge-L	METEOR	RE	Self-BLEU
Visual	10.19	6.41	10.41	17.50	31.47	25.74	12.05	29.00	56.57
Visual + range	10.54	7.53	12.16	20.32	36.58	25.42	11.86	27.62	54.11
Visual + caption	11.29	7.90	12.61	20.83	36.55	23.82	13.24	18.92	49.14
Visual + range + caption	**14.15**	**8.45**	**13.52**	**22.26**	**39.11**	**25.88**	**14.75**	**17.59**	**45.80**

Feature for Event Selection In this part, we study the effect of using different features for event selection. Specifically, the following combinations are tested. (1) *visual*: using visual features $\mathbf{v}$ in isolation, which serves as a baseline for other combinations. (2) *visual + range*: combining visual features $\mathbf{v}$ with range features $\mathbf{t}$, which provides the temporal ranges in the original videos in addition to visual features. (3) *visual + caption*: combining visual features $\mathbf{v}$ with caption feature $\mathbf{c}$, which provides event selection module the ability to know *what have been said*. (4) *visual + range + caption*: using all features.

As shown in Table 3, compared to *visual*, both *visual + range* and *visual + caption* outperforms it by a significant margin, indicating both the temporal range features and the caption features are complementary to the visual information. It is reasonable as the visual features contain no temporal information, which is explicitly captured in temporal range features, and semantically captured in caption features. Moreover, caption features and temporal range features are also complementary to each other, as the full combination *visual + range + caption* outperforms both *visual + range* and *visual + caption*.

5.5 Human Study

To have a more pertinent assessment, we also conduct a human study, where 20 users are asked to pick the best descriptive paragraph among those generated by the proposed MFT, Hierarchical-RNN, and Semantic-Sum, in terms of relevance, coherence and conciseness respectively. Figure 3 shows that MFT generates the best ones across all aspects.

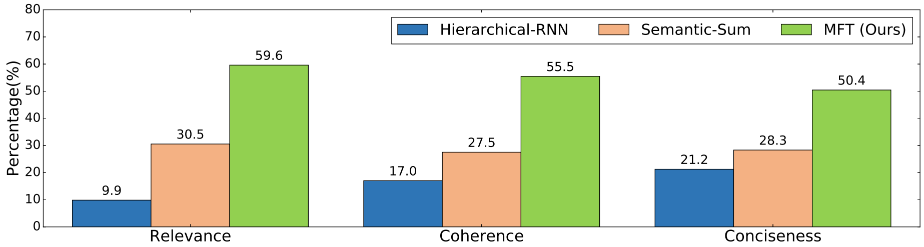

Fig. 3. The figure shows the results of human evaluation, which compares paragraphs generated by different methods, with respect to relevance, coherence and conciseness.

5.6 Qualitative Example

In Figure 4, we also include one qualitative example, where a video is shown with paragraphs generated by *Dense-Caption-NMS*, *Hierarchical-RNN*, *Semantic-Sum* and our method *MFT*. As shown in Figure 4, compared to the baselines, our method produces more concise and more coherent paragraphs. We will provide more qualitative results in the supplementary materials.

Dense-Caption-NMS	A gymnast is seen walking ready to a beam and leads into her performing a routine. The girl then begins to do a routine and flips flips and flips and flips off the beam. He mounts his arms up and down on the mat. A woman is seen standing on a horse and begins walking around. A girl is seen standing on a beam beam with her arms around and performing flips… (remaining skipped)
Hierarchical-RNN	A gymnast is seen performing a routine and leads into several people performing a gymnastics routine. The gymnast performs a routine on a beam. The gymnast dismounts and lands on the pommel horse.
Semantic-Sum	A gymnast is seen standing ready with her arms up in the air. A man is seen standing before a beam and begins performing a gymnastics routine. A man is seen standing ready to perform a routine on the bars.
MFT (ours)	A gymnast is seen standing ready to a beam and performing a gymnastics routine on a beam. The girl performs several flips and flips around the beam and ends by jumping off the beam.
Ground-Truth	A girl jumps onto a balance beam. She does a gymnastics routine on the balance beam. She does a flip off the balance beam and lands on a mat.

Fig. 4. This figure lists a qualitative example, where paragraphs generated by *Dense-Caption-NMS*, *Hierarchical-RNN*, *Semantic-Sum* and our method *MFT* are shown.

6 Conclusion

We presented a new framework for generating coherent descriptions for given videos. The generated paragraphs well align with the temporal structure of the given video, covering major semantics without redundancy. Specifically, it sequentially locates important events via an LSTM net, which selects events from a candidate pool, based on their appearances, temporal locations and mutual semantic relationships. When respectively describing events in the acquired sequence, it explicitly decides *what to say next* based on temporal structure and *what have been said*. On ActivityNet Captions, our method significantly outperformed others on a wide range of metrics, while producing more coherent and more concise paragraphs.

Acknowledgement This work is partially supported by the Big Data Collaboration Research grant from SenseTime Group (CUHK Agreement No. TS1610626), the Early Career Scheme (ECS) of Hong Kong (No. 24204215).

References

1. Baraldi, L., Grana, C., Cucchiara, R.: Hierarchical boundary-aware neural encoder for video captioning. arXiv preprint arXiv:1611.09312 (2016)
2. Dai, B., Fidler, S., Urtasun, R., Lin, D.: Towards diverse and natural image descriptions via a conditional gan. In: International Conference on Computer Vision (ICCV). pp. 2989–2998. IEEE (2017)
3. Dai, B., Lin, D.: Contrastive learning for image captioning. In: Advances in Neural Information Processing Systemsr(NIPS). pp. 898–907 (2017)
4. Dai, B., Zhang, Y., Lin, D.: Detecting visual relationships with deep relational networks. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 3298–3308. IEEE (2017)
5. Denkowski, M., Lavie, A.: Meteor universal: Language specific translation evaluation for any target language. In: Proceedings of the ninth workshop on statistical machine translation. pp. 376–380 (2014)
6. Farhadi, A., Hejrati, M., Sadeghi, M.A., Young, P., Rashtchian, C., Hockenmaier, J., Forsyth, D.: Every picture tells a story: Generating sentences from images. In: European conference on computer vision. pp. 15–29. Springer (2010)
7. Geiger, A., Lenz, P., Stiller, C., Urtasun, R.: Vision meets robotics: The kitti dataset. The International Journal of Robotics Research **32**(11), 1231–1237 (2013)
8. Hori, C., Hori, T., Lee, T.Y., Sumi, K., Hershey, J.R., Marks, T.K.: Attention-based multimodal fusion for video description. arXiv preprint arXiv:1701.03126 (2017)
9. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)
10. Krishna, R., Hata, K., Ren, F., Fei-Fei, L., Niebles, J.C.: Dense-captioning events in videos. In: International Conference on Computer Vision (ICCV) (2017)
11. Kulkarni, G., Premraj, V., Ordonez, V., Dhar, S., Li, S., Choi, Y., Berg, A.C., Berg, T.L.: Babytalk: Understanding and generating simple image descriptions. IEEE Transactions on Pattern Analysis and Machine Intelligence **35**(12), 2891–2903 (2013)
12. Lin, C.Y.: Rouge: A package for automatic evaluation of summaries. In: Text summarization branches out: Proceedings of the ACL-04 workshop. vol. 8. Barcelona, Spain (2004)
13. Pan, Y., Yao, T., Li, H., Mei, T.: Video captioning with transferred semantic attributes. arXiv preprint arXiv:1611.07675 (2016)
14. 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. pp. 311–318. Association for Computational Linguistics (2002)
15. Regneri, M., Rohrbach, M., Wetzel, D., Thater, S., Schiele, B., Pinkal, M.: Grounding action descriptions in videos. Transactions of the Association for Computational Linguistics **1**, 25–36 (2013)
16. Rennie, S.J., Marcheret, E., Mroueh, Y., Ross, J., Goel, V.: Self-critical sequence training for image captioning. arXiv preprint arXiv:1612.00563 (2016)
17. Rohrbach, A., Rohrbach, M., Qiu, W., Friedrich, A., Pinkal, M., Schiele, B.: Coherent multi-sentence video description with variable level of detail. In: German conference on pattern recognition. pp. 184–195. Springer (2014)
18. Rohrbach, A., Rohrbach, M., Schiele, B.: The long-short story of movie description. In: German Conference on Pattern Recognition. pp. 209–221. Springer (2015)

19. Sah, S., Kulhare, S., Gray, A., Venugopalan, S., Prud'Hommeaux, E., Ptucha, R.: Semantic text summarization of long videos. In: Applications of Computer Vision (WACV), 2017 IEEE Winter Conference on. pp. 989–997. IEEE (2017)
20. 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)
21. Steinberger, J., Jezek, K.: Using latent semantic analysis in text summarization and summary evaluation. Proc. ISIM **4**, 93–100 (2004)
22. 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. pp. 4566–4575 (2015)
23. 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. pp. 4534–4542 (2015)
24. Vinyals, O., Toshev, A., Bengio, S., Erhan, D.: Show and tell: A neural image caption generator. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 3156–3164 (2015)
25. Wang, J., Wang, W., Huang, Y., Wang, L., Tan, T.: Multimodal memory modelling for video captioning. arXiv preprint arXiv:1611.05592 (2016)
26. Wang, L., Xiong, Y., Wang, Z., Qiao, Y., Lin, D., Tang, X., Van Gool, L.: Temporal segment networks: Towards good practices for deep action recognition. In: European Conference on Computer Vision. pp. 20–36. Springer (2016)
27. 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. pp. 2048–2057 (2015)
28. 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. pp. 4507–4515 (2015)
29. 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. pp. 4584–4593 (2016)
30. 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. pp. 3165–3173 (2017)
31. Zhao, Y., Xiong, Y., Wang, L., Wu, Z., Tang, X., Lin, D.: Temporal action detection with structured segment networks. In: ICCV (2017)
32. Zhu, Y., Lu, S., Zheng, L., Guo, J., Zhang, W., Wang, J., Yu, Y.: Texygen: A benchmarking platform for text generation models. arXiv preprint arXiv:1802.01886 (2018)