

Learning Multimodal Attention LSTM Networks for Video Captioning

Jun Xu[†], Ting Yao[‡], Yongdong Zhang[†], Tao Mei[‡]

[†]University of Science and Technology of China, Hefei, China

[‡]Microsoft Research, Beijing, China

junx1992@gmail.com; {tiyao,tmei}@microsoft.com; zhyd73@ustc.edu.cn

ABSTRACT

Automatic generation of video caption is a challenging task as video is an information-intensive media with complex variations. Most existing methods, either based on language templates or sequence learning, have treated video as a flat data sequence while ignoring intrinsic multimodality nature. Observing that different modalities (e.g., frame, motion, and audio streams), as well as the elements within each modality, contribute differently to the sentence generation, we present a novel deep framework to boost video captioning by learning Multimodal Attention Long-Short Term Memory networks (MA-LSTM). Our proposed MA-LSTM fully exploits both multimodal streams and temporal attention to selectively focus on specific elements during the sentence generation. Moreover, we design a novel child-sum fusion unit in the MA-LSTM to effectively combine different encoded modalities to the initial decoding states. Different from existing approaches that employ the same LSTM structure for different modalities, we train modality-specific LSTM to capture the intrinsic representations of individual modalities. The experiments on two benchmark datasets (MSVD and MSR-VTT) show that our MA-LSTM significantly outperforms the state-of-the-art methods with **52.3 BLEU@4** and **70.4 CIDEr-D** metrics on MSVD dataset, respectively.

KEYWORDS

Multimodal Fusion; Video Captioning; CNN; LSTM; Deep Learning

1 INTRODUCTION

Automatic generation of natural language description for video, a.k.a. video captioning, has played a fundamental challenge and received extensive research interest in both

This work was performed when the first author Jun Xu was visiting Microsoft Research as a research intern.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

MM '17, October 23–27, 2017, Mountain View, CA, USA

© 2017 Association for Computing Machinery.

ACM ISBN 978-1-4503-4906-2/17/10...\$15.00

<https://doi.org/10.1145/3123266.3123448>



A white car is driving fast through a sharp curve with soft music on.

Figure 1: An Example video with human annotated sentence. Words in red color, purple color, green color can be referred to visual frame, motion and audio stream respectively.

multimedia and vision communities. With the rapid development of deep learning techniques, impressive progress has been made in this emerging area.

Existing approaches to video captioning mainly proceed along two dimensions: template-based language model [3, 10, 17] and sequence learning model [13, 14, 24, 30, 33]. The former predefines a set of language templates, which follows specific grammar rules for sentence generation and aligns each part of sentence with video content. This dimension of approaches highly depend on the predefined templates and the recognizable words from videos, making the generated sentences limited to constant syntactical structure. Sequence learning-based approaches, in contrast, leverage sequence learning models, which are commonly used in the machine translation area [21], to directly translate the video content into a sentence. The network architecture of this kind, often follows an encoder by Convolutional Neural Networks (CNNs) with Recurrent Neural Networks (RNNs) reading the whole video sequence and producing the video representation, and in turn a decoder RNNs generating a natural sentence based on the syntactical patterns learned from training data.

However, video is an information-intensive media with large variations and complexities. For example, video contains multiple modalities (i.e., frame, motion, audio, and so on). Different modalities depict unique nature and at the same time, complement each other. This has made video captioning a very challenging task. Figure 1 shows an example of this nature: the object words “a white car” and “a sharp curve” can be recognized from individual frames, “driving fast” may have high correlations with motion information, while “with soft music” can only be recognized from the audio stream. Previous research [13, 14, 24, 30, 33] simply concatenates different features into one single video representation, while neglecting the intrinsic modality nature. This has resulted two critical issues. First, different streams in one video (e.g., frame, motion , and audio streams) demonstrate different temporal dynamics and thus should be modeled individually

(rather than by one single network). For example, the objects in a video could be the same throughout the lifetime of the video, while the motion and audio may change time to time. **Second, different modalities, as well as the visual or aural elements within each modality, contribute differently to the sentence generation.** Therefore, the learning model should not only investigate the different contributions from those modalities, but also exploit temporally attended part within each modality.

Observed by the above challenges, we propose a novel deep framework to boost video captioning by learning Multimodal Attention Long-Short Term Memory networks (MA-LSTM). Our proposed MA-LSTM fully exploits both multimodal streams and temporal attention to selectively focus on specific elements during the sentence generation. Different with previous dimension of sequence learning-based models, which treat video as a flat data sequence, we fully utilize the multimodal sequences (frame, motion, and audio) as the inputs. In our work, we select the Long-Short Term Memory (LSTM) to capture the sequential information for both video and sentence sequences as it is designed to avoid the long-term dependency problem. In the encoding stage, multiple encoding LSTMs are used to model the temporal sequence for different modalities. Furthermore, we devise two kinds of fusion units to merge the input states of multiple streams to get the whole video representations and the initial states for decoding LSTM. Besides, we adopt the soft attention mechanism, which has been approved the effectiveness by previous work [30]. In particular, we extend the soft attention mechanism by not only localizing the attended parts within each modality but also exploring the contributions across different modalities.

The main contribution of this work is the proposal of MA-LSTM for exploiting the video representations using different modalities in video captioning and generating the sentence with attention from different modalities and their related elements. Besides, we conduct extensive experiments and show that our framework outperforms several state-of-the-art methods for objective evaluation metrics and human judgments in a user study with 30 subjects.

The remaining sections are organized as follows. Section 2 describes related work on video representations and video captioning. Section 3 presents our MA-LSTM model. Section 4 provides the implementation details and results for both objective metrics and human judgement on two popular video captioning datasets, followed by the conclusions in Section 5.

2 RELATED WORK

In this section, we categorize the related work into two subgroups: video representations and video captioning.

Video Representations. In the recent decades, deep learning approaches have been used to learn video representations and have produced state-of-art results [1, 9, 19, 22]. Karpathy *et al.* [9] propose to use a CNNs feature to represent the video. In [22], Tranet *al.* propose to use a 3D CNN for video representations, which leveraging large training datasets

such as Sport-1M. Different with image classification [20], CNNs do not yield large improvement over traditional methods highlighting the difficulty of learning video representations even with large training dataset. Simonyan and Zisserman [19] bring in a two-stream framework where they train CNNs independently on RGB and optical flow inputs. The optical flow stream focuses only on motion information, while the RGB stream can leverage 2D CNN pre-trained on image datasets. Based on the two stream representations, Wang *et al.* [26] extract deep feature and conducted trajectory constrained pooling to aggregate convolutional feature as video representations. Furthermore, Ballas *et al.* leverage convolutional GRU-RNN to extract visual representations from all levels of a deep convolutional network in [1].

Video Captioning. Research in this direction mainly includes two different directions: template-based language methods [3, 7, 10, 17, 28] and sequence learning approaches [6, 13–15, 24, 30, 33]. Template based language methods firstly align each sentence fragments (e.g., subject, verb, object) with detected words from visual content and then generate the sentence with predefined language templates, which highly depend on the templates of sentence. [10] is one of the earlier works that builds a concept hierarchy of actions for natural language description of human activities. Rohrbach *et al.* learn a CRF to model the relationships between different components of the input video and generate description for video [17]. Recently, a deep joint video-language embedding model in [28] is designed for video sentence generation. Different from template-based language methods, sequence learning can be applied to video description as video is naturally a sequence of objects and actions. Donahue *et al.* leverage CNN to learn the single frame representation as the input to the long-term recurrent convolutional networks to output sentences [6]. In [24], Venugopalan *et al.* design an encoder-decoder neural network to generate descriptions. By mean pooling, the features over all frames can be represented by one single vector, which is the input of the RNN. Furthermore, Pan *et al.* additionally consider the relevance between sentence semantics and video content as a regularizer in LSTM based architecture [14]. Compared to mean-pooling, Li *et al.* propose to utilize the temporal attention mechanism to exploit temporal structure as well as a spatiotemporal convolutional neural network [30]. Besides, in [13, 33], both of the works move one step forward to exploit the long-term temporal structures in videos. Most recently, video commenting, which could be regarded as a variant or extension of video captioning, is proposed in [11].

Different from existing works, in this paper we present a novel MA-LSTM framework for video captioning, which exploits multiple video representations with different modalities. With a well-designed fusion unit, different modalities could be adaptively integrated to obtain a complementary video representation. By further incorporating a two-layer attention mechanism, attentions over temporal sequences and across multiple streams are both exploited to better generate video descriptions.

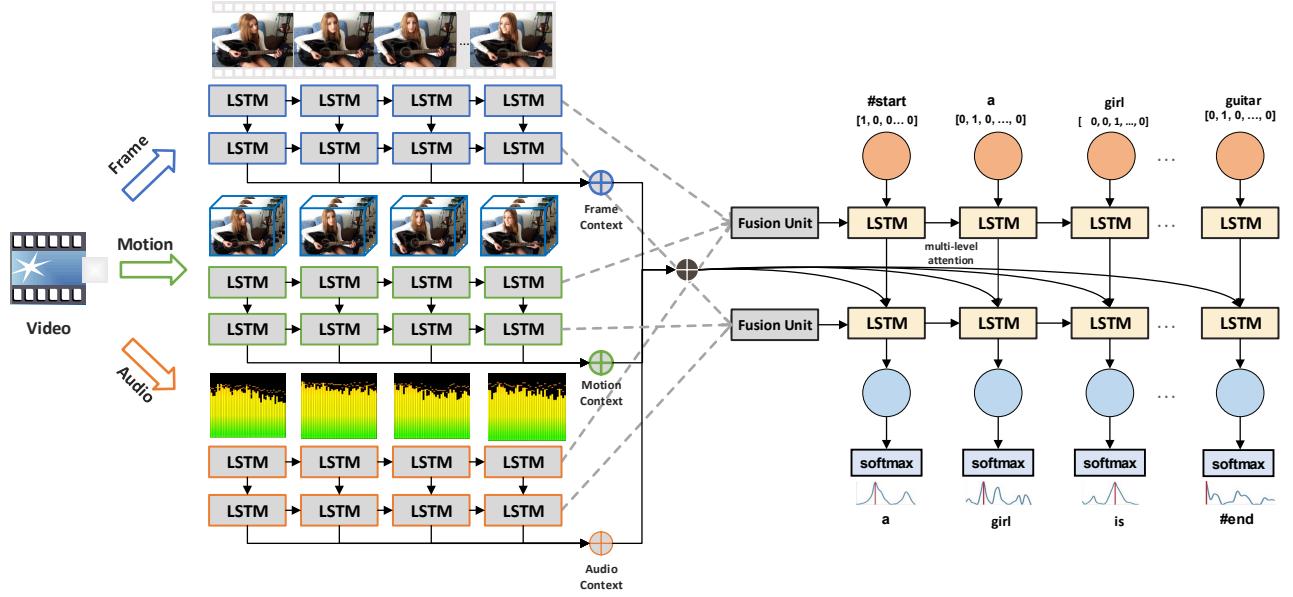


Figure 2: The overall architecture of MA-LSTM. The model is based on sequence to sequence framework, which contains an encoder and decoder. Both of them are established using LSTM. First, three LSTM models are used to encode features of different modalities (video frames, video motion and audio) separately. Then a fusion unit is used to elegantly combine different modal streams, and output the initialization of the decoder. A multi-level attention mechanism (the brown circle) is further used to help better capture the key clues in videos, which leverages attentions both from temporal sequences and across multimodal streams. After that, the decoder predicts words sequentially. Note that for different time step t in decoding stage, the attention weights are different.

3 APPROACH

Different from image captioning, which only cares about the visual information, video captioning requires much more ability of perceiving multimodal data, including static vision, motion and audio. We aim at fully exploiting the multimodality nature of video, as well as the intuitive attention mechanism, to carry out the challenging task. First, multiple LSTM models are trained to represent multimodal data separately. Then we use a fusion unit to elegantly combine different modal streams, from which complementary information is dynamically learned. Furthermore, we design a multi-level attention mechanism to pick key elements both from temporal sequences and across multimodal streams.

We begin this section with the overview of the proposed MA-LSTM in section 3.1. In section 3.2, we give our multi-stream fusion method in detail. The proposed attention mechanism is shown in section 3.3. Finally, we present the caption generation inference in section 3.4.

3.1 Overview of the Proposed Method

Given a video V , we observe k different data streams with the corresponding feature sequences $(x_1^1, x_2^1, \dots, x_{n_1}^1), (x_1^2, x_2^2, \dots, x_{n_2}^2), \dots, (x_1^l, x_2^l, \dots, x_{n_l}^l), \dots, (x_1^k, x_2^k, \dots, x_{n_k}^k)$, in which n_l denotes the sequence index for the l -th stream. The goal

of video captioning is to output a textual sentence $S = \{w_1, w_2, \dots, w_t, \dots, w_{N_s}\}$ with N_s words based on the above observed data. w_t represents the individual word in the output sentence at time stamp t . As mentioned in Section 2, recent deep based sequence learning methods have achieved significant successes in this task. Therefore we build our method on a typical sequence to sequence video captioning framework. Figure 2 shows the overall architecture of the proposed MA-LSTM.

Our approach contains two components, i.e. the encoder and decoder. Taking the inspiration from image/video captioning [24, 31, 32], both of the encoder and decoder are established using the popular Long Short-Term Memory (LSTM) model, which has been shown to be capable of capturing important sequence information by using a memory cell c [8]. In particular, we train multiple LSTM models to represent the video, where each model encodes only one single modal data stream. While most existing works [1, 25] only utilize single model to extract video features, our method can benefit from the multimodal nature of video. To better exploit the complementary information from multiple streams, we propose to use a fusion unit to combine different representations. The fusion unit outputs the initial states of the decoder. In addition, a two layer attention mechanism is applied in the

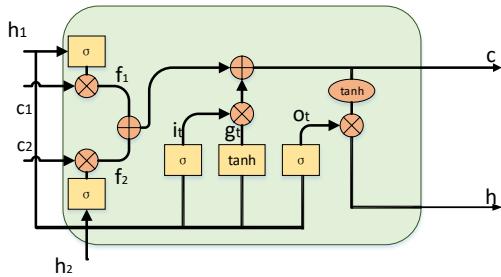


Figure 3: The structure of Child-Sum fusion unit (two streams example). The inputs are the parameters of multiple encode models. The output is used as the initialization of the decoder. h_i, c_i are the input hidden states, memory cells. i, f, o, c, h are input gate, forget gate, output gate, cell state and hidden state respectively.

decoding stage. By leveraging attentions from both temporal sequence level channel and stream level channel, the decoder can output more accurate description. After that, a series of words are obtained sequentially from the decoder.

3.2 Fusion Unit for Multiple Streams

How to fuse different modal features and utilize complementary clues from them is very important in multimodal analysis. Conventional fusion methods, including early fusion and late fusion, simply concatenate feature vectors or make the average pooling on the features/scores. Such fusions completely ignore the differences and relationships between those modalities. For example, audio stream plays a totally different role when the video content is about “music concert” and “making cake” separately. Therefore, some fusion methods are proposed in [34] to learn different weights of the feature streams in Neural Machine Translation (NMT) area. Nevertheless, only considering modality weights but treating them equally may still not yield satisfactory performance, given the large varieties of modal features. The model should have the ability to identify which part is useful in different modal streams. To this end, we propose two kinds of fusion unit here. One employs a simple linear transformation to learn stream weights, and the other uses multiple “gates” to determine the contribution of each modality adaptively. We further compare them in the experiment section.

Basic Combination Fusion Unit: The goal of fusion unit is to integrate k hidden states (h_1, h_2, \dots, h_k) and memory cells (c_1, c_2, \dots, c_k) into one single hidden state h and cell state c . Here we apply a \tanh non-linearity on the linear transformation of all encode models, obtaining a single decoder initialization. The detailed transformations are as follows,

$$h = \tanh(W_c[h_1; h_2; \dots; h_k]), \quad (1)$$

$$c = \sum_{i=1}^k (c_i), \quad (2)$$

where W_c is the transformation parameter, i.e. the weights of representation models, which are learned automatically in

training. This fusion unit learns weights for different feature modalities. However, it still treats each modal stream equally.

Child-Sum Fusion Unit: To better learn the important clues from different streams, we further use the child-sum fusion unit. Different from traditional LSTM, this unit takes multiple model parameters as inputs. One forget gate is used to learn the temporal saliency and pick useful information from each input stream. Moreover, it provides a seamless integration with the whole LSTM structure. Figure 3 shows the unit structure. The detailed calculations of unit input, output, cell value, hidden state and gate state are as follows:

$$i = \sigma(\sum_{l=1}^k W_l^i h_l + b_i), \quad f_l = \sigma(W_l^f h_l + b_l^f), \quad (3)$$

$$o = \sigma(\sum_{l=1}^k W_l^o h_l + b_o), \quad g = \tanh(\sum_{l=1}^k W_l^g h_l + b_g), \quad (4)$$

$$c = i \odot g + \sum_{l=1}^k f_l \odot c_l, \quad h = o \odot \tanh(c), \quad (5)$$

where i, f, o, c, h represents input gate, forget gate, output gate, cell state and hidden state respectively in our child-sum unit. σ and \odot represents a sigmoid activation function and an element wise multiplication, respectively. $W_l^i, W_l^f, W_l^o, W_l^g, b_i, b_l^f, b_o$ and b_g are parameters for each gate in l -th stream.

This unit provides a more accurate fusion strategy to select useful parts from each stream, which dynamically learns the effect of each modal stream.

3.3 Multi-level Attention Mechanism

Given the multimodal streams, MA-LSTM outputs words embedding sequentially via multi-level attention reasoning.

In many cases, the caption is only related to a small part of the whole video. For example, in Figure 1, the key clues of the caption are only “a white car” “driving fast” “a sharp curve” and “soft music” which do not always present along the whole video. Using the global information to describe the video could lead to suboptimal results due to the noises introduced from regions that are irrelevant to the potential caption. Instead, the following two select strategies should be considered. First, we need to find out salient part from each modal data. Second, we need to learn important clues across all the multimodal data. To this end, by utilizing a multi-level attention mechanism, where both temporal sequence level and modal stream level attentions are used, our MA-LSTM is able to gradually filter out noises and figure out the clues that are highly relevant to the caption.

Assuming that we have a sequence of feature vector in one stream with length of n , we first generate the caption word by word using only one stream. At each word step, we need to select relevant information from different temporal sequence of the feature vectors, which are referred as context vectors ($\varphi_1, \varphi_2, \dots, \varphi_{N_s}$). Instead of simply pooling the context vectors into a single vector, which neglecting the inherent structure and difference among the temporal information, the sequence level attention mechanism calculates weights α_t^i range from 0 to 1 for each φ_t , conditioning on the input

vector v_i at each time step t . In this case, the output context vector in time t can be represented as:

$$\varphi_t(V) = \sum_{i=1}^n \alpha_i^t v_i, \quad (6)$$

where $\sum_{i=1}^n \alpha_i^t = 1$. We refer to α_i^t as the attention weight at time t for the input stream.

The sequence level attention weight α_i^t reflects the relevance of the i -th temporal feature in the input feature vector sequences give all the previously generated words, i.e., w_1, \dots, w_{t-1} . Followed by [30], we use the function that takes as input the previous hidden state h_{t-1} of the LSTM decoder, which summarizes all the previously generated words and the feature vector of the i -th temporal feature and returns the unnormalized relevance score e_i^t :

$$e_i^t = w^\tau \tanh(W_a h_{t-1} + U_a v_i + b_a), \quad (7)$$

where w^τ , W_a , U_a and b_a represent the first level attention parameters that are estimated together with all other parameters of the whole networks.

After the calculation of relevance scores e_i^t , the attention weight α_i^t at time t can be normalized as:

$$\alpha_i^t = \exp\{e_i^t\} / \sum_{j=1}^n \exp\{e_j^t\}. \quad (8)$$

While for modality level attention, we consider another mechanism on different streams. The input context vectors from multiple modalities can be represented as $(\varphi_1^1, \varphi_2^1, \dots, \varphi_{N_s}^1)$, $(\varphi_1^2, \varphi_2^2, \dots, \varphi_{N_s}^2)$, \dots , $(\varphi_1^k, \varphi_2^k, \dots, \varphi_{N_s}^k)$. So we have the following formulations for each stream l at time t :

$$\varphi_l^l(V_l) = \sum_{i=1}^{n_l} \alpha_{li}^t v_{li}, \quad (9)$$

$$e_{li}^t = w_{la}^\tau \tanh(W_{la} h_{t-1} + U_{la} v_{li} + b_{la}), \quad (10)$$

$$\alpha_{li}^t = \exp\{e_{li}^t\} / \sum_{j=1}^{n_l} \exp\{e_{lj}^t\}, \quad (11)$$

where w_{la} , W_{la} , U_{la} and b_{la} are the second level attention parameters to be estimated with the whole networks.

Similarly, we define the modality level attention weight β_l^t range from 0 to 1 to reflect the relevance of the l -th stream context feature $\varphi_l^l(V_l)$ with the all previously generated words. The whole video context at time t as well as the relevance scores s_l to different streams can be represented as:

$$\varphi_l(V) = \sum_{l=1}^k \beta_l^t \varphi_l^l(V_l), \quad (12)$$

$$s_l^t = w_b^\tau \tanh(W_b h_{t-1} + U_b \varphi_l^l(V_l) + b_b), \quad (13)$$

$$\beta_l^t = \exp\{s_l^t\} / \sum_{j=1}^k \exp\{s_j^t\}. \quad (14)$$

3.4 Caption Generation Inference

Given multiple input sequence $X = (x_1^1, \dots, x_{n_1}^1), (x_1^2, \dots, x_{n_2}^2), \dots, (x_1^k, \dots, x_{n_k}^k)$, the proposed MA-LSTM obtains a set of hidden states (h_1, \dots, h_k) for k streams. After the fusion unit, only a single LSTM decoder is obtained with a hidden state h_d . Given the input sequence X , the distribution $p(S|X)$ over the output sequence $S = (w_1, \dots, w_{N_s})$ in the decoding stage is calculated as

$$p(w_1, \dots, w_{N_s}|X) = \prod_{t=1}^{N_s} p(w_t|h_d^{n+t-1}, w_{t-1}), \quad (15)$$

where the distribution of $p(w_t|h_d^{n+t})$ is given by a softmax over all the words in the vocabulary. Besides, parameters

among the attention mechanism are computed at the same time while decoding process.

While training in the decoding stage, the model maximizes the log-likelihood of the predicted output sentence given the hidden representation of the video, and the previous words it has seen. The model over the parameter θ and output sequence $S = (w_1, \dots, w_{N_s})$ is then optimized as:

$$\theta^* = \arg \max_{\theta} \sum_{t=1}^{N_s} \log p(w_t|h_d^{n+t-1}, w_{t-1}; \theta). \quad (16)$$

θ not only contains all the network parameters but also the parameters during the soft attention. Each time in the training, the parameters α_i^t and β_l^t in Section 3.3 are updated according the entire network loss. This log-likelihood is optimized over the entire training dataset by using stochastic gradient descent. While in testing stage, α_i^t and β_l^t are fixed based on the given video encoding results.

4 EXPERIMENTS

We evaluate and compare our proposed MA-LSTM model with several state-of-the arts approaches on two video captioning benchmarks, i.e., Microsoft Research Video Description Corpus (MSVD) [4] and MSR-Video to Text (MSR-VTT) [27]. The former is the most popular video captioning benchmark of YouTube videos and the latter is a recently released video captioning dataset.

4.1 Datasets

MSVD. The MSVD dataset contains 1,970 videos clips collected from YouTube. There are roughly 40 available English descriptions per video. MSVD includes about 80,000 video-description pairs with around 13,000 unique words in total. In our experiments, we follow the setting used in prior works [14, 25, 30, 33], taking 1,200 videos for training, 100 for validation and 670 for testing.

MSR-VTT. MSR-VTT is a recently released large-scale video benchmark for video captioning, consisting of 10K web video clips with 41.2 hours in 20 well-defined categories. In this dataset, the vocabulary size is about 30,000. Each video clip is equipped with around 20 natural sentences annotated by AMT workers. Following the original split in MSR-VTT, we utilize 6,513, 2,990, and 467 video clips for training, testing and validation, respectively.

4.2 Experimental Settings

We uniform sample 25 frames/clips for each video and each word in the sentence is represented as “one-hot” vector (binary index vector in a vocabulary). For frame representation, we take the output of 4096-way fc7 layer from VGG and 1024-dimensional pool5/7x7 layer of GoogleNet pre-trained on ImageNet dataset[18]. For motion representation, we take the output of 4096-way fc6 layer from C3D pre-trained on Sports-1M video dataset [9]. Moreover, Mel-Frequency Cepstral Coefficients (MFCC) [29] is leveraged to represent audio information. Note that videos in MSVD do not contain any

Model	BLEU@1	BLEU@2	BLEU@3	BLEU@4	METEOR	CIDEr-D
LSTM (G) [25]*	72.8	57.8	46.8	36.4	28.6	48.2
LSTM (V) [25]*	75.1	59.4	47.6	36.3	28.5	48.1
LSTM (C) [25]*	75.2	60.2	48.2	36.4	29.3	47.3
LSTM (G+C) [25]*	78.2	66.5	55.1	42.5	30.8	56.9
LSTM-E (V+C) [14]	78.8	66.0	55.4	45.3	31.0	-
S2VT (V+O) [24]	-	-	-	-	29.8	-
SA (G+M) [30]	80.0	64.7	52.6	41.9	29.6	51.7
GRU-RCN (G) [1]	-	-	-	43.3	31.6	68.0
p-RNN (V+C) [33]	81.5	70.4	60.4	49.9	32.6	65.8
HRNE (V+C) [13]	81.1	68.6	57.8	46.7	33.9	-
M-LSTM (G+C)(basic)	80.1	68.1	58.6	49.2	32.6	67.2
M-LSTM (G+C)(child-sum)	81.2	69.5	59.1	50.3	32.9	68.4
MA-LSTM (G+C)(basic)	81.6	70.3	60.9	51.4	33.4	69.7
MA-LSTM (G+C)(child-sum)	82.3	71.1	61.8	52.3	33.6	70.4

Table 1: BLEU@N, METEOR and CIDEr-D scores of our proposed models and other state-of-the-art methods on MSVD dataset. All values are reported as percentage (%). The short name in the brackets indicates the frame/motion features, where G, V, C, O and M denotes GoogleNet, VGGNet, C3D, optical flow and motion feature learnt by 3D CNN on hand-crafted descriptors, respectively. “*” indicates that the reported performances are based on our implementations. “-” means that the authors did not report their performance on this dataset.

audio information and thus we include the audio stream only for the experiments on MSR-VTT.

In the training phase, we add a begin-of-sentence tag $\langle BOS \rangle$ to start each sentence and an end-of-sentence tag $\langle EOS \rangle$ to end each sentence. In the testing phase, we input $\langle BOS \rangle$ into video decoder and then generate the final output sentence by beam search strategy (the beam size is set as 5). For LSTM and child-sum fusion unit, the dimension of the input and hidden layers are both set to 1,024. We use different LSTM cells to model the sequence information for different streams. We apply the optimizer ADAM to minimize the negative log-likelihood loss for training process and set the learning rate $l = 10^{-4}$.

For quantitative evaluation of our proposed models, we adopt three common metrics in image/video captioning tasks: BLEU@N [16], METEOR [2], and CIDEr-D [23] in MSVD dataset. In MSR-VTT dataset, following the official evaluation metrics in Microsoft Multimedia Challenge¹, we adopt BLEU@4, METEOR, CIDEr-D, and ROUGE-L [12] for evaluation. All the metrics are computed by using the codes² released by Microsoft COCO Evaluation Server [5].

4.3 Compared Approaches

To empirically verify the merit of our MA-LSTM models, we compared the following state-of-the-art methods.

(1) LSTM [25]: LSTM attempts to directly translate from video pixels to natural language with a CNN plus RNN framework. The video representation is generated by performing mean pooling over all the frame/clip features.

(2) Sequence to Sequence–Video to Text (S2VT) [24]: S2VT leverages the stacked LSTM as an encoder-decoder model for

video sentence generation, which firstly encodes the whole video and then decodes the video description.

(3) Soft-Attention (SA) [30]: SA exploits a weighted attention mechanism dynamically attend to specific temporal regions of the video while generating sentence.

(4) Long Shot-Term Memory with visual-semantic Embedding (LSTM-E) [14]: LSTM-E simultaneously explores the learning of LSTM and visual-semantic embedding for video sentence generation.

(5) Convolutional Gated-Recurrent-Unit Recurrent Networks (GRU-RCN) [1]: GRU-RCN leverages convolutional GRU-RNN to extract visual representation and generate sentence based on the LSTM with attention mechanism [30].

(6) paragraph Recurrent Neural Networks (p-RNN) [33]: Proposed most recently, p-RNN exploits both spatial and temporal attention mechanisms for video captioning.

(7) Hierarchical Recurrent Neural Encoder (HRNE) [13]: HRNE encodes the frame sequence with hierarchical RNN and decodes the sentence with attention mechanism.

(8) Multimodal Attention Long-Short Term Memory networks (MA-LSTM) is our proposal in this work. A slightly different of this run is named as M-LSTM, which is trained without attention mechanism.

4.4 Experimental Results on MSVD

Table 1 shows the performances of different models on MSVD video captioning dataset. It is worth noting that the performances of different approaches here are based on different video representations. In view that GoogleNet and VGGNet are comparable, we compare directly with results. Overall, the results across five out of six evaluation metrics consistently indicate that our proposed MA-LSTM model exhibits better performance than all the state-of-the-art techniques

¹<http://ms-multimedia-challenge.com/>

²<https://github.com/tylin/coco-caption>

Model	B@4	M	C	R
LSTM (G) [25]*	33.5	24.2	34.1	54.1
LSTM (C) [25]*	33.7	24.4	34.6	54.7
LSTM (G+C) [25]*	34.1	24.8	35.5	55.8
LSTM (G+C+A) [25]*	35.7	25.6	38.1	58.2
LSTM-E (G+C+A) [14]*	36.1	25.8	38.5	58.6
S2VT (G+C+A) [24]*	36.0	26.0	39.1	58.4
SA (G+C+A) [30]*	34.8	25.1	36.7	57.1
M-LSTM (G+C)(basic)	34.9	25.2	36.5	57.0
M-LSTM (G+C)(child-sum)	35.2	25.5	37.4	58.2
MA-LSTM (G+C)(basic)	35.4	25.8	38.1	58.2
MA-LSTM (G+C)(child-sum)	35.8	26.0	39.6	58.6
MA-LSTM (G+C+A)(basic)	36.3	26.3	40.1	59.1
MA-LSTM (G+C+A)(child-sum)	36.5	26.5	41.0	59.8

Table 2: Performances of our proposed models and other state-of-the-art methods on MSR-VTT dataset, where B@4, M, C and R are short for BLEU@4, METEOR, CIDEr-D and ROUGE-L scores. All values are reported as percentage (%). The short name in the brackets indicates the frame/motion/audio features, where G, C and A denotes GoogleNet, C3D and audio feature, respectively. “*” indicates that the reported performances are based on our implementations.

including non-attention models (LSTM, LSTM-E, S2VT) and attention-based approaches (SA, GRU-RCN, p-RNN and HRNE). In particular, the CIDEr-D of MA-LSTM can achieve 70.4%, making the relative improvement over p-RNN and GRU-RCN by 7.0% and 3.5%, respectively. By additionally incorporating attention to control the impacts of different temporal parts in a video represented by different modalities, MA-LSTM improves M-LSTM. Furthermore, MA-LSTM modeling not only temporal attention but also modality attention leads to a performance boost against SA, GRU-RCN, p-RNN and HRNE which only capitalizes on attention mechanism on temporal dimension. The results basically indicate the advantage of exploring attention to fuse all modalities for enhancing video understanding. There is a performance gap between S2VT and MA-LSTM. Though both runs involve utilization of multiple modalities in a sequence to sequence architecture for video captioning, they are fundamentally different in the way that the performance of S2VT is as a result of directly late fusing the results of different modalities, and MA-LSTM is by integrating the fusion of different modalities into the encoder-encoder process. This somewhat reveals the weakness of late fusion, where the influences of different modalities are not fully explored. Compared to basic fusion which treats each modality equally, child-sum fusion dynamically determining the contribution of each modality leads to better performance.

4.5 Experimental Results on MSR-VTT

The performance comparisons on MSR-VTT are summarized in Table 2. Our MA-LSTM performs constantly better than other baselines. Specifically, the BLEU@4, METEOR, CIDEr-D and ROUGE-L score of MA-LSTM can reach 36.5%, 26.5%,

Model	C1	C2	C3	C4
LSTM (G+C)	3.11	2.85	3.22	0.42
LSTM-E (G+C)	3.16	2.86	3.23	0.45
S2VT (G+C)	3.22	2.86	3.23	0.45
SA (G+C)	3.17	2.91	3.24	0.44
MA-LSTM (G+C)(child-sum)	3.31	2.95	3.24	0.51
HUMAN	4.02	4.31	4.11	0.81

Table 3: The user study on MSVD dataset.

Model	C1	C2	C3	C4
LSTM (G+C+A)	3.14	2.75	2.96	0.38
LSTM-E (G+C+A)	3.16	2.81	3.05	0.40
S2VT (G+C+A)	3.15	2.80	3.13	0.40
SA (G+C+A)	3.16	2.82	3.11	0.41
MA-LSTM (G+C+A)(child-sum)	3.18	2.86	3.24	0.44
HUMAN	4.13	4.24	4.14	0.89

Table 4: The user study on on MSR-VTT dataset.

41.0% and 59.8%, making the relative improvement over the best competitor S2VT by 1.4%, 1.9%, 4.9% and 2.4%, respectively. As expected, MA-LSTM utilizing all three modalities exhibits better performance than that only using two modalities in the context of both basic and child-sum fusion. Similar to the observations on MSVD dataset, MA-LSTM outperforms M-LSTM by further taking attention mechanism into account to fuse representations of different parts in temporal video sequence. Compared to SA which also exploits attention, MA-LSTM is benefited from the utilization of attention additionally on modality level and leads to improvement. Furthermore, the child-sum fusion consistently shows better performance than basic fusion, demonstrating the advantage of dynamically fusing the influences of different modalities in understanding videos.

4.6 Qualitative Analysis

Figure 4 showcases a few sentence examples generated by different methods and human-annotated ground truth sentences. From these exemplar results, it is easy to see that all of these automatic methods can generate somewhat relevant sentences, while our proposed MA-LSTM can predict more relevant keywords by elegantly incorporating the fusion of multiple modalities into video captioning with attention mechanism. For example, compared to verb term “playing” in the sentence generated by LSTM, “eating” in our MA-LSTM is more precise to describe the video content in the first video. Similarly, the term “explaining cooking” presents the fourth video more exactly. Moreover, the generated sentences are further enriched by involving audio information on MSR-VTT dataset. For instance, the output sentence “Men and women are dancing with music” of the fifth video depicts the video content very comprehensive.

		...			LSTM(G+C): a boy is playing tools MA-LSTM(G+C)(child-sum): a cat is eating	Ground Truth: 1. kitten is eating food 2. a cat is eating from a bowl 3. The animals are eating
		...			LSTM(G+C): a man is riding a horse MA-LSTM(G+C)(child-sum): a boy is running and playing basketball	Ground Truth: 1. a man runs while dribbling a basketball 2. a man slowly dribbles towards basket 3. a man is running and dribbling a basketball
		...			LSTM(G+C): a man is playing guitar MA-LSTM(G+C)(child-sum): a man is sitting and playing guitar	Ground Truth: 1. a man is playing the guitar on a park bench 2. a man is playing the guitar seated on a bench in an outdoor location 3. a man is sitting on a bench playing a guitar
		...			LSTM(G+C+A): a person is cooking MA-LSTM(G+C+A)(child-sum): a man is explaining cooking	Ground Truth: 1. a man is explaining about the preparation of naan 2. a man demonstrates how to make a good fish 3. chef explains how to make a meal
		...			LSTM(G+C+A): a group of people are dancing MA-LSTM(G+C+A)(child-sum): Men and women are dancing with music	Ground Truth: 1. a bunch of people dancing 2. a group of people are all dancing in a room 3. dancers dance to the beat of a love song

Figure 4: Sentence generation results on MSVD (the first three examples) and MSR-VTT (the last two examples). The videos are represented by sampled frames, the output sentences generated by 1) LSTM, 2) MA-LSTM with child-sum fusion and 3) Ground Truth: Randomly selected three ground truth sentences.

4.7 Human Evaluation

To better understand how satisfactory are the sentences generated from different methods, we conduct a user study to compare our MA-LSTM against four approaches, i.e., LSTM, LSTM-E, S2VT and SA. A total number of 30 evaluators (15 females and 15 males) from different education backgrounds, including computer science (10), business (6), linguistics (6) and engineering (8), are invited and a subset of 500 videos is randomly selected from testing set of MSVD and MSR-VTT, respectively, for the subjective evaluation.

The evaluation process is as follows. All the evaluators are organized into two groups. We show the first group all the five sentences generated by each approach plus one human-annotated sentences and ask them to rank all the sentences from 1 to 5 (lower to better) with respect to the three criteria: 1) Coherence: judge the logic and readability of the sentence; 2) Relevance: whether the sentence contains the more relevant and important objects/actions/events in the video clip? 3) Helpful for blind: how helpful would the sentence be for a blind person to understand what is happening in this video clip? We average the scores on each criterion of all the generated sentences by each method and obtain three metrics, i.e., C1, C2 and C3. In contrast, we show the second group once only one sentence generated by different approach or human annotation and they are asked: Can you determine whether the given sentence has been generated by a system or by a human being? From evaluators' responses, we calculate another metric of C4, which is the percentage of captions that pass the Turing Test. Table 3 and 4 lists the results of the user study on MSVD and MSR-VTT dataset, respectively.

Overall, our MA-LSTM is clearly the winner across all the four criteria on two datasets. In particular, C4 performance achieves 51% and 44%, making the absolute improvement over SA by 7% and 3% on MSVD and MSR-VTT, respectively.

5 CONCLUSIONS

In this paper we propose a novel deep framework to boost video captioning by learning Multimodal Attention LSTM model. The proposed MA-LSTM fully exploits both multimodal streams and two layer attention to selectively focus on specific elements during the sentence generation. In particular, a child-sum fusion unit is proposed to elegantly combine different modal streams, which dynamically integrates complementary video representations. A multi-level attention mechanism is designed to obtain key clues both over temporal sequences and across multimodal streams. Extensive experiments have shown the superiority of the proposed method compared with the state-of-the-arts.

Our future works are as follows. First, a more complete video structure explore can be conducted to get a better understanding for video. Besides, how to localize the captioning sentence and generate multiple sentences for long video are also expected.

ACKNOWLEDGEMENTS

This work was supported by National Natural Science Foundation of China (Grant No.61525206).

REFERENCES

- [1] Nicolas Ballas, Li Yao, Chris Pal, and Aaron Courville. 2016. Delving Deeper into Convolutional Networks for Learning Video Representations. In *ICLR*.
- [2] Satanjeev Banerjee and Alon Lavie. 2005. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *ACL*.
- [3] Andrei Barbu, Alexander Bridge, Zachary Burchill, Dan Coroian, Sven Dickinson, Sanja Fidler, Aaron Michaux, Sam Mussman, Siddharth Narayanaswamy, Dhaval Salvi, and others. 2012. Video In Sentences Out. *UAI* (2012).
- [4] David L Chen and William B Dolan. 2011. Collecting highly parallel data for paraphrase evaluation. In *ACL*.
- [5] Xinlei Chen, Hao Fang, Tsung-Yi Lin, Ramakrishna Vedantam, Saurabh Gupta, Piotr Dollár, and C Lawrence Zitnick. 2015. Microsoft COCO captions: Data collection and evaluation server. *arXiv preprint arXiv:1504.00325* (2015).
- [6] Jeff Donahue, Lisa Anne Hendricks, Sergio Guadarrama, Marcus Rohrbach, Subhashini Venugopalan, Kate Saenko, and Trevor Darrell. 2015. Long-term recurrent convolutional networks for visual recognition and description. In *CVPR*.
- [7] Sergio Guadarrama, Niveda Krishnamoorthy, Girish Malkarnenkar, Subhashini Venugopalan, Raymond Mooney, Trevor Darrell, and Kate Saenko. 2013. Youtube2text: Recognizing and describing arbitrary activities using semantic hierarchies and zero-shot recognition. In *CVPR*.
- [8] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation* (1997).
- [9] Andrej Karpathy, George Toderici, Sanketh Shetty, Thomas Leung, Rahul Sukthankar, and Li Fei-Fei. 2014. Large-scale video classification with convolutional neural networks. In *CVPR*.
- [10] Atsuhiro Kojima, Takeshi Tamura, and Kunio Fukunaga. 2002. Natural language description of human activities from video images based on concept hierarchy of actions. *International Journal of Computer Vision* (2002).
- [11] Yehao Li, Ting Yao, Tao Mei, Hongyang Chao, and Yong Rui. 2016. Share-and-Chat: Achieving Human-Level Video Commenting by Search and Multi-View Embedding. In *ACM MM*.
- [12] Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *ACL*.
- [13] Pingbo Pan, Zhongwen Xu, Yi Yang, Fei Wu, and Yueling Zhuang. 2016. Hierarchical recurrent neural encoder for video representation with application to captioning. In *CVPR*.
- [14] Yingwei Pan, Tao Mei, Ting Yao, Houqiang Li, and Yong Rui. 2016. Jointly modeling embedding and translation to bridge video and language. In *CVPR*.
- [15] Yingwei Pan, Ting Yao, Houqiang Li, and Tao Mei. 2017. Video Captioning with Transferred Semantic Attributes. In *CVPR*.
- [16] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: a method for automatic evaluation of machine translation. In *ACL*.
- [17] Marcus Rohrbach, Wei Qiu, Igor Titov, Stefan Thater, Manfred Pinkal, and Bernd Schiele. 2013. Translating video content to natural language descriptions. In *ICCV*.
- [18] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, and others. 2015. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision* (2015).
- [19] Karen Simonyan and Andrew Zisserman. 2014. Two-stream convolutional networks for action recognition in videos. In *NIPS*.
- [20] Karen Simonyan and Andrew Zisserman. 2014. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556* (2014).
- [21] Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. In *NIPS*.
- [22] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. 2015. Learning spatiotemporal features with 3d convolutional networks. In *CVPR*.
- [23] Ramakrishna Vedantam, C Lawrence Zitnick, and Devi Parikh. 2015. Cider: Consensus-based image description evaluation. In *CVPR*.
- [24] Subhashini Venugopalan, Marcus Rohrbach, Jeff Donahue, Raymond Mooney, Trevor Darrell, and Kate Saenko. 2015. Sequence to Sequence—Video to Text. In *ICCV*.
- [25] Subhashini Venugopalan, Huijuan Xu, Jeff Donahue, Marcus Rohrbach, Raymond Mooney, and Kate Saenko. 2014. Translating videos to natural language using deep recurrent neural networks. *arXiv preprint arXiv:1412.4729* (2014).
- [26] Limin Wang, Yuanjun Xiong, Zhe Wang, and Yu Qiao. 2015. Towards good practices for very deep two-stream convnets. *arXiv preprint arXiv:1507.02159* (2015).
- [27] Jun Xu, Tao Mei, Ting Yao, and Yong Rui. 2016. MSR-VTT: A large video description dataset for bridging video and language. In *CVPR*.
- [28] Ran Xu, Caiming Xiong, Wei Chen, and Jason J Corso. 2015. Jointly Modeling Deep Video and Compositional Text to Bridge Vision and Language in a Unified Framework.. In *AAAI*.
- [29] Zhongwen Xu, Yi Yang, Ivor Tsang, Nicu Sebe, and Alexander G Hauptmann. 2013. Feature weighting via optimal thresholding for video analysis. In *CVPR*.
- [30] Li Yao, Atousa Torabi, Kyunghyun Cho, Nicolas Ballas, Christopher Pal, Hugo Larochelle, and Aaron Courville. 2015. Describing videos by exploiting temporal structure. In *ICCV*.
- [31] Ting Yao, Yingwei Pan, Yehao Li, and Tao Mei. 2017. Incorporating Copying Mechanism in Image Captioning for Learning Novel Objects. In *CVPR*.
- [32] Ting Yao, Yingwei Pan, Yehao Li, Zhaofan Qiu, and Tao Mei. 2017. Boosting Image Captioning with Attributes. In *ICCV*.
- [33] Haonan Yu, Jiang Wang, Zhiheng Huang, Yi Yang, and Wei Xu. 2016. Video paragraph captioning using hierarchical recurrent neural networks. In *CVPR*.
- [34] Barret Zoph and Kevin Knight. 2016. Multi-Source Neural Translation. In *NAACL-HLT*.