TVT: Two-View Transformer Network for Video Captioning

Ming Chen FUNKYBLACK@ZJU.EDU.CN
Yingming Li * YINGMING@ZJU.EDU.CN
Zhongfei Zhang ZHONGFEI@ZJU.EDU.CN
Siyu Huang SIYUHUANG@ZJU.EDU.CN
College of Information Science and Electronic Engineering, Zhejiang University, Hangzhou, China

Editors: Jun Zhu and Ichiro Takeuchi

Abstract

Video captioning is a task of automatically generating the natural text description of a given video. There are two main challenges in video captioning under the context of an encoder-decoder framework: 1) How to model the sequential information; 2) How to combine the modalities including video and text. For challenge 1), the recurrent neural networks (RNNs) based methods are currently the most common approaches for learning temporal representations of videos, while they suffer from a high computational cost. For challenge 2), the features of different modalities are often roughly concatenated together without insightful discussion. In this paper, we introduce a novel video captioning framework, i.e., Two-View Transformer (TVT). TVT comprises of a backbone of Transformer network for sequential representation and two types of fusion blocks in decoder layers for combining different modalities effectively. Empirical study shows that our TVT model outperforms the state-of-the-art methods on the MSVD dataset and achieves a competitive performance on the MSR-VTT dataset under four common metrics.

Keywords: Video captioning; Sequence-to-sequence; Multi-modalities

1. Introduction

Describing videos with natural language, namely video captioning, has been a widely studied topic in computer vision and natural language processing communities. It can be applied in various applications such as video retrieval Song et al. (2018), visual question answering Antol et al. (2015), and helping the disabled with visual impairment. The difficulties of video captioning mainly lie in the modeling of temporal dynamics and the fusion of multiple modalities. In this paper, we novelly introduce the Two-View Transformer (TVT) model which is a variant of the Transformer network Vaswani et al. (2017) towards tackling the challenges.

The first challenge of video captioning is how to model the temporal dynamics. In the existing literature, the encoder-decoder architecture is commonly adopted, where most previous work utilizes convolutional neural networks (CNNs) for encoding visual contents in conjunction with the recurrent neural networks (RNNs), particularly, the long short-term memory unit (LSTM) Hochreiter and Schmidhuber (1997) and gated recurrent unit (GRU) Chung et al. (2014), for processing sequential data Venugopalan et al. (2015a); Yao et al. (2015); Yu et al. (2016); Pu et al. (2018). In RNNs, a hidden state is computed based on the previous hidden state such that it disallows the parallelization. The sequential computation introduces a high cost especially for long sequences.

* Corresponding author

Inspired by the promising results of the Transformer network Vaswani et al. (2017) in machine translation, we propose to use the Transformer network as our backbone network for video captioning. The Transformer network relies on the attention mechanism instead of RNNs to draw dependencies between sequential data. We discuss how to adapt the Transformer network to video captioning task in Section 3.2.

Another challenge of video captioning is how to fuse information of different modalities. Several previous methods roughly concatenate features of different modalities together Yao et al. (2015); Pan et al. (2016b). Recently, the attention mechanism has been used for better fusion Long et al. (2016); Hori et al. (2017). However, these approaches often fuse modalities from the video data only, such as image, motion, and audio representations. The fusion mechanism of video and text data has not been adequately studied. In this work, we attempt to build a more effective language decoder. We novelly propose the attentive fusion block to utilize the multi-head attention mechanism for the fusion of modalities from both video and text data period.

In summary, our proposed Two-View Transformer (TVT) model uses the Transformer network as the backbone network and fuses the modalities from video and text data with attention mechanism. TVT has the following advantages:

- Compared with RNNs, the attention layers of TVT allow for parallel computing, leading to a much more efficient training process without performance decrease.
- Two different types of fusion blocks are designed for a more effective decoding process than a simple concatenation. In particular, the attentive fusion block adjusts the flow of text information for generating more natural sentences.

We summarize the contributions of this paper as follows:

- We propose a novel TVT model for video captioning. TVT learns the long-term dependencies of sequential data based on the multi-head attention mechanism instead of the widely used RNN units.
- We propose two types of late fusion blocks to provide a novel way to exploit information from three different modalities containing features of frames, motions, and previous generated words.
- Empirical study shows that TVT achieves the state-of-the-art performance on the MSVD dataset and competitive performance on the MSR-VTT dataset. Ablation study further reveals the effectiveness of our proposed fusion blocks.

2. Related Work

Image Captioning. The goal of image captioning is to present a caption for describing a given image. Many existing image captioning approaches have adopted the encoder-decoder framework for generating image captions. Vinyals et al. (2015) proposed a model consisting of a deep CNN encoder for encoding the visual information and a following RNN decoder for generating descriptions. Xu et al. (2015) further utilized the spatial attention mechanism to improve the performance of RNN decoder. You et al. (2016) and Gan et al. (2017) incorporated the semantic features of images into a language model. Lu et al. (2017) proposed an adaptive attention model that can automatically decide whether to rely on the image when generating the next word. Our attentive fusion

block is inspired by the adaptive attention mechanism. Different from their approaches, in this work we consider more perspectives of visual contents, that better help the learning of video captioning model.

Video Captioning. Video captioning is a more challenging task than image captioning, mainly due to the temporal dynamics underlying in videos. Venugopalan et al. (2015b) proposed a model that averages all the frames' feature extracted by a 2-D CNN, and then feeds the averaged visual feature to an LSTM decoder to generate video descriptions. However, their model lacks the ability to address temporal dynamics of videos. Venugopalan et al. (2015a) proposed the S2VT model which employs LSTM as an additional encoder to model the long-term dependency of frames and feeds the output of last step in this LSTM layer to another LSTM decoder. Similarly to the sequence-to-sequence models used in other applications, Yao et al. (2015) proposed an attention mechanism to assign every frame a weight when generating words. As videos have different modalities, Jin et al. (2016) proposed a multi-modal fusion encoder to combine all the available modalities with one fully connected layer. Long et al. (2016) and Hori et al. (2017) both employed the attention mechanism in the multi-modal fusion models. In our approach, the attentive fusion block combines two modalities including the frame and motion features.

Machine Translation. In recent years, sequence-to-sequence models Bahdanau et al. (2014); Sutskever et al. (2014); Cho et al. (2014b,a); Vaswani et al. (2017) have been widely used in machine translation task. Specifically, an RNN encoder maps the source sentence into a context vector and then an RNN decoder generates the target sentence conditioned on the context vector. Since the context vector is fixed when generating each word for target sentence, Bahdanau et al. (2014) proposed an encoder-decoder network equipped with soft attention mechanism to adaptively learn the context vector according to individual words of the source sentence. Although the RNNs based encoder/decoder learns the long-range dependence of sequences, it has a high computational cost; further, the context vector in each position is not well balanced due to the chained architecture of RNNs. Instead of the use of recurrent layers, Vaswani et al. (2017) proposed the Transformer network to accelerate the training process based on dot-product attention mechanism, showing a promising performance in machine translation. Similar to machine translation, video captioning is also the sequence-to-sequence learning task as its goal is encoding a video into a sequence of vectors. Zhou et al. (2018) proposed an end-to-end transformer model for both detecting and describing events of dense video captioning. Thus, inspired by Transformer network, we adopt the dot-product attention mechanism Vaswani et al. (2017) as the main module of our video captioning approach.

3. Methodology

Our video captioning framework includes two basic modules, including 1) a 2-D CNN serving as the visual feature extractor, and 2) a Transformer network Vaswani et al. (2017) serving as the caption generator. Fig. 1 shows a general architecture of our framework. In this section, we first give a brief introduction to the original Transformer network before we describe our model.

3.1. Transformer Network

3.1.1. Dot-Product Attention

The *scaled dot-product attention* is different from the conventional attention mechanisms as its attention weights are computed by dot-product operation. Given queries $Q \in \mathbb{R}^{T_q \times d_k}$, keys $K \in$

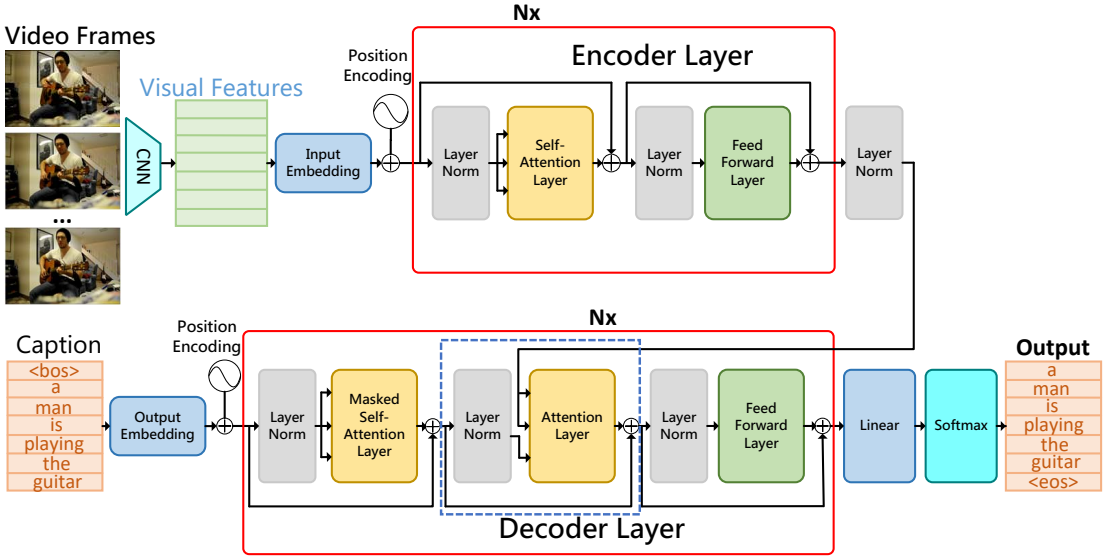

Figure 1: Base Model architecture

$\mathbb{R}^{T_v \times d_k}$ and values $V \in \mathbb{R}^{T_v \times d_v}$, the attention output is

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V, \tag{1}$$

where T_q is the sequence length of queries and T_v is the sequence length of keys and values. d_k is the vector dimension of queries and keys, and d_v is the vector dimension of values. $\sqrt{d_k}$ is used for scaling here which guarantees the numerical stability.

The *multi-head attention* is built upon the scaled dot-product attention. It consists of h different "heads" of (*query, key, value*), where each head is independently and computed in parallel. For the i-th head, the attention output is

$$\text{head}_i = \text{Attention}(QW_i^Q, KW_i^K, VW_i^V). \tag{2}$$

The multi-head attention concatenates the heads as

$$\text{MultiHead}(Q, K, V) = \text{Concat}_{i=1\ldots h}(\text{head}_i)W^O. \tag{3}$$

We employ the multi-head attention as the attention layers in our framework.

3.1.2. FEED-FORWARD NETWORK

Each block in our encoder and decoder contains a two-layered fully connected feed-forward network with an ReLU activation. This module is defined as:

$$\text{FFN}(\boldsymbol{x}) = \text{ReLU}(\boldsymbol{x}W_1 + \boldsymbol{b}_1)W_2 + \boldsymbol{b}_2, \tag{4}$$

where $W_1 \in \mathbb{R}^{d_{\text{model}} \times d_h}$, $\boldsymbol{b}_1 \in \mathbb{R}^{d_h}$, $W_2 \in \mathbb{R}^{d_h \times d_{\text{model}}}$, $\boldsymbol{b}_2 \in \mathbb{R}^{d_{\text{model}}}$ are trainable weights. d_h is the hidden state size.

3.1.3. POSITIONAL ENCODING

Since the multi-head attention and feed-forward network contain no convolutional layers or recurrent cells, positional encoding is essential for leveraging the relative position information in sequence. The encoding method is defined as:

$$\begin{aligned} \mathrm{PE}(pos, 2i) &= \sin(pos/10000^{2i/d_{\text{model}}}) \\ \mathrm{PE}(pos, 2i+1) &= \cos(pos/10000^{2i/d_{\text{model}}}), \end{aligned} \tag{5}$$

where pos is the position of a frame in the video on the encoding side or a word in the sentence on the decoding side. i denotes the corresponding dimension of the embeddings. Here different dimensions of the embedded positional vector represent different frequencies when using sinusoidal or cosinusoidal function. Different positions represent different phases in these periodic functions.

3.2. Two-View Transformer Network

3.2.1. TRANSFORMER ENCODER

Our proposed Two-View Transformer Network includes two views of visual representations extracted by the encoders, i.e., the frame representation E_f and the motion representation E_m, respectively. The frame representation $E_f \in \mathbb{R}^{T\times d_{\text{model}}}$ is obtained by a 2-D CNN on individual frames and subsequent self-attention layers. The motion representation $E_m \in \mathbb{R}^{T\times d_{\text{model}}}$ is obtained by a 3-D CNN on consecutive frames and independent self-attention layers.

3.2.2. TWO-VIEW TRANSFORMER DECODER

Two types of fusion blocks of our Two-View Transformer Decoder are shown in Fig. 2, where the fusion attention layer provides a special way for computing attention weights. Given the sentence representation D_s from previous masked self-attention layer and the outputs of two visual encoders E_f and E_m, the tuple of (*query, key, value*) is computed as

$$\begin{aligned} Q_f &= \mathrm{LayerNorm}(D_s)W_f^Q \\ K_f &= E_f W_f^K \\ V_f &= E_f W_f^V \end{aligned} \tag{6}$$

where W_f^Q, W_f^K, $W_f^V \in \mathbb{R}^{d_{\text{model}}\times d_{\text{model}}}$ are parametric matrices. Another tuple for motion presentation is computed in the same way with different parameters. Then the frame context C_f and the motion context C_m are computed by two independent multi-head attention layers as

$$\begin{aligned} C_f &= \mathrm{MultiHead}(Q_f, K_f, V_f) \\ C_m &= \mathrm{MultiHead}(Q_m, K_m, V_m). \end{aligned} \tag{7}$$

3.3. Fusion Block

3.3.1. ADD-FUSION BLOCK

To fuse the frame context C_f and the motion context C_m, we propose to use two types of fusion blocks. The first type of fusion blocks is the add-fusion block. It uses a simple add operation which

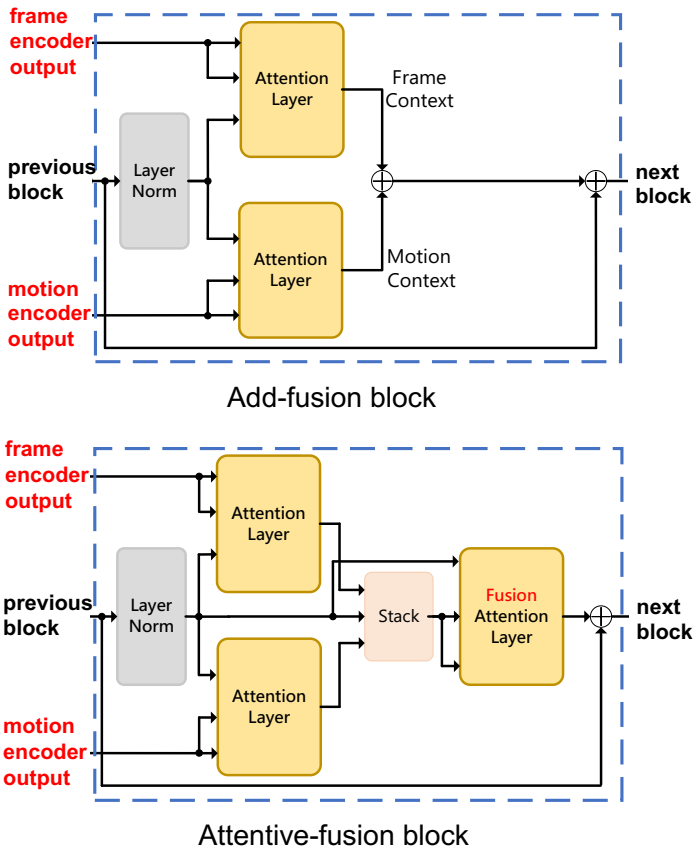

Figure 2: Two types of fusion blocks in Two-View Transformer decoder.

combines C_f and C_m with fixed weight $\alpha \in [0, 1]$ to balance the contributions from the two view visual representations

$$C = \alpha C_f + (1 - \alpha) C_m. \tag{8}$$

The add-fusion block is a non-parametric fusion method such that it does not change the way of fusion according to the context of words. It lacks robustness for integrating the complex and diverse two-view representations.

3.3.2. ATTENTIVE-FUSION BLOCK

We propose another type of fusion blocks, named attentive-fusion block, to fuse the two-view representations in a learnable way. For each position in a sentence, we generate a new query vector from the representation of current position and stack context vectors from different modalities to compute new keys and values matrices.

$$\begin{aligned} S_t &= \text{Stack}(\boldsymbol{c}_t^f, \boldsymbol{c}_t^m, \boldsymbol{d}_t) \\ \boldsymbol{q}_t &= W^q \boldsymbol{d}_t \\ K_t &= S_t W^K \\ V_t &= S_t W^V, \end{aligned} \tag{9}$$

where the weight matrices W^q, W^K and W^V are shared across different positions. $\boldsymbol{c}_t^f, \boldsymbol{c}_t^m$ and $\boldsymbol{d}_t$ are vectors of the t-th column of frame context C_f, motion context C_m, and sentence context D_s, respectively. Intuitively, the attentive-fusion block computes the attention weights depending on $\boldsymbol{q}_t$ and K_t, and performs a weighted average operation on V_t to obtain the context vector $\boldsymbol{c}_t$ in current position. Then, it adopts the scaled dot-product attention to give the output as

$$\boldsymbol{c}_t = \text{Attention}(\boldsymbol{q}_t, K_t, V_t). \tag{10}$$

3.3.3. Discussion on Fusion Methods

- **Early fusion.** Given a sequence of frame features $X = (\boldsymbol{x}_1, \boldsymbol{x}_2, ..., \boldsymbol{x}_T)$, and a sequence of motion features $Z = (\boldsymbol{z}_1, \boldsymbol{z}_2, ..., \boldsymbol{z}_T)$, an early fusion method fuses $\boldsymbol{x}_t$ and $\boldsymbol{z}_t$ as

$$\boldsymbol{x}'_t = \begin{bmatrix} \boldsymbol{x}_t \\ \boldsymbol{z}_t \end{bmatrix} \tag{11}$$

  where $\boldsymbol{x}'_t$ is then fed into a Transformer decoder. Early fusion lacks flexibility as it only works when the lengths of different sequences are the same. In this work, we use two independent Transformer encoders to encode two different features, respectively, enabling a much more flexible approach for fusion in the decoder.

- **Add-fusion and attentive-fusion.** Theoretically, an attentive-fusion block has two advantages over an add-fusion block. First, the attention weight varies depending on the context of current position. Second, the decoder can adopt an appropriate sentence context by selecting $\boldsymbol{d}_t$ in D_s, such that the frame representation, the motion representation, and the previously generated words are able to jointly guide the description generation process.

4. Experimental Setup

4.1. Datasets

We evaluate our model on two video captioning benchmark datasets: Microsoft Research Video Description (MSVD) Guadarrama et al. (2013) and Microsoft Research Video to Text (MSR-VTT) Xu et al. (2016).

- The **MSVD** dataset consists of 1,970 short Youtube video clips with an average length of 9s around. Each clip is labeled with about 40 English sentences provided by Amazon Mechanical Turk workers. Following the existing literature Venugopalan et al. (2015a); Dong et al. (2017), we split the datasets into three parts: 1200 videos for training, 100 videos for validation, and 670 videos for testing.

- The **MSR-VTT** dataset is a large-scale video benchmark dataset that contains 10,000 video clips, covering a wide variety of video categories. Each clip is annotated with about 20 natural sentences. This dataset is divided into three parts: 6513 videos for training, 497 videos for validation, and 2990 videos for testing.

4.2. Evaluation Metrics

We evaluate the performances of generating descriptions with four metrics: BLEU@4 Papineni et al. (2002), METEOR Denkowski and Lavie (2014), ROUGE-L Lin (2004) and CIDEr Vedantam et al. (2015). We use the standard evaluation protocol from Microsoft COCO evaluation server Chen et al. (2015).

4.3. Implementation Details

Preprocessing. For the MSVD dataset, we sample the videos at 5 *fps* and set the maximum number of frames as 50 to extract image features. For motion features, we sample the videos at 25 *fps* and extract features for every 64 consecutive frames with overlap, setting the interval as 5 frames.

For the MSR-VTT dataset, we sample the videos at 3 *fps* and set the maximum number of frames as 60 to extract image features. For motion features, we sample the videos at 15 *fps* and extract features for every 64 consecutive frames with overlap, setting the interval as 5 frames.

Model Details. On encoder side, we compare two image feature extractors, ResNet-152 and NasNet Zoph et al. (2017), which are both pretrained on the ImageNet dataset Krizhevsky et al. (2012). The extracted image features are 2048-dimension and 4032-dimension, respectively. For motion features, we use I3D Carreira and Zisserman (2017) network pre-trained on the Kinetics dataset Kay et al. (2017) to obtain 1024-dimension features. Since the MSR-VTT dataset contains audio tracks for most videos, pre-trained Vggish Hershey et al. (2017) network is utilized to extract deep audio features with 128-dimension. Our fusion models are simply extended in the same ways in two different fusion blocks for incorporating audio features to secure improvements.

On decoder side, for text descriptions, we remove punctuations in every sentence and build vocabulary containing 9861 and 10551 words by filtering words whose count is less than 1 and 3, respectively in two datasets. Maximum sentence length is set as 20 for two datasets.

For the Transformer network, model dimension d_{model} is set as 512 and hidden state size of the feed-forward layer is set as 2048. We use 8 heads in the multi-head attention layer where dimension $d_k = d_v = 64$. We set 4 encoder layers and 4 decoder layers to build the whole Transformer network. α in the add-fusion block is set as 0.4.

Learning settings. For the training process, dropout Srivastava et al. (2014) with a drop rate of 0.3 is adopted for regularization. We stop training after reaching 20 epochs or the METEOR score is not increased on the validation set in the last 10 checkpoints. We use Adam optimizer Kingma and Ba (2014) with a learning rate of 0.0001 to train neural networks. Beam search with a beam size of 5 is adopted for testing.

5. Results and Analysis

5.1. Comparison with State-of-the-Art Methods

Table. 1 shows performances of our proposed model and several state-of-the-art methods on the MSVD dataset. Our method performs significantly better than the previous methods on this dataset. There is a 7.84% relative improvement over the previous best CIDEr score. The performance of our model under BLEU@4 metric is also higher than all the previous methods.

Table. 2 shows performances against the top-3 teams from the MSR-VTT Challenge 2017, v2t_navigator, Aalto and VideoLAB, and results of three recent methods containing CIDEnt-RL,

Table 1: **Results on the MSVD dataset**. Att-TVT is our proposed model using Two-View Transformer with attentive fusion block. Here our ROUGE score is omitted since previous work did not report it.

Models	BLEU@4	METEOR	CIDEr
LSTM-YT Venugopalan et al. (2015b)	33.29	29.07	-
S2VT Venugopalan et al. (2015a)	-	29.80	-
LSTM-I Dong et al. (2017)	44.60	29.70	-
SA Yao et al. (2015)	41.92	29.60	51.67
LSTM-E Pan et al. (2016b)	45.30	31.00	-
GRU-RCN Ballas et al. (2015)	43.26	31.60	68.01
h-RNN decoder Yu et al. (2016)	49.90	32.60	65.80
h-RNN encoder Pan et al. (2016a)	46.70	33.90	-
SCN-LSTM Gan et al. (2017)	51.10	33.50	77.70
TSA Pan et al. (2017)	52.80	33.50	74.00
M&M TGM Chen et al. (2017)	48.76	34.36	80.45
dualAFR Pu et al. (2018)	51.77	**36.41**	72.21
RecNet Wang et al. (2018a)	52.30	34.10	80.30
Att-TVT	**53.21**	35.23	**86.76**

Table 2: **Results on the MSR-VTT dataset.**

Models	BLEU	METEOR	ROUGE	CIDEr
VideoLAB Ramanishka et al. (2016)	39.10	27.70	60.60	44.10
Aalto Shetty and Laaksonen (2016)	39.80	26.90	59.80	45.70
v2t_navigator Jin et al. (2016)	40.80	28.20	60.90	44.80
CIDEnt-RL Pasunuru and Bansal (2017)	40.50	28.40	61.40	**51.70**
Dense-Cap Shen et al. (2017)	41.40	28.30	61.10	48.90
HRL Wang et al. (2018b)	41.30	**28.70**	**61.70**	48.00
Att-TVT	40.12	27.86	59.63	47.72
Att-TVT(+audio)	**42.46**	28.24	61.07	48.53

Dense-Cap, and HRL on MSR-VTT dataset. For a fair comparison, we simply extend Att-TVT to three views by adding audio features. The results of our method are competitive on this dataset.

5.2. Ablation Study

We report ablation studies to empirically demonstrate the effectiveness of our proposed fusion methods. We compare the performances of our base models and two-view models with different types of fusion. Table. 3 and Table. 4 shows the results of our ablation study. Our base model achieves a

Table 3: **Ablation study on the MSVD dataset.** Here, R, N, I are short for ResNet-152, NasNet and I3D visual features. TVT is the Two-View Transformer with early fusion and Add-TVT is the Two-View Transformer with add-attention fusion decoder.

Models	BLEU	METEOR	ROUGE	CIDEr
Base model(R)	50.25	33.41	70.16	72.11
Base model(N)	52.55	34.36	70.12	75.94
TVT(R+I)	52.07	33.18	69.71	77.02
Add-TVT(R+I)	52.17	34.40	71.11	77.98
Att-TVT(R+I)	52.96	34.73	71.71	80.84
TVT(N+I)	53.04	34.52	70.79	77.69
Add-TVT(N+I)	**53.94**	34.77	71.88	78.95
Att-TVT(N+I)	53.21	**35.23**	**72.01**	**86.76**

Table 4: **Ablation study on the MSR-VTT dataset.** Here V is short for Vggish audio feature.

Models	BLEU	METEOR	ROUGE	CIDEr
Base model(R)	38.27	27.23	58.72	44.99
Base model(N)	37.96	27.05	58.79	45.60
TVT(N+I)	38.96	27.52	59.33	45.67
Add-TVT(N+I)	40.16	27.53	59.64	46.87
Att-TVT(N+I)	40.12	27.86	59.63	47.72
Add-TVT(N+I+V)	41.61	**28.29**	60.72	47.89
Att-TVT(N+I+V)	**42.46**	28.24	**61.07**	**48.53**

strong baseline with only frame features extracted by ResNet-152. To utilize better frame features for this task, we select the NasNet, which achieves a higher accuracy on the image classification problem, as another feature extractor for comparison. Results on both datasets show that NasNet performs slightly better for generating video descriptions.

Comparing different fusion methods, the early fusion with a simple concatenation provides a little performance boost than without motion features. The late fusion in the decoding stage is a better way to integrate motion features into frame features. The attention mechanism applied in the fusion block makes two-view Transformer decoder attend to appropriate context adaptively. For example, Att-TVT(R+I) has performed better than Add-TVT(R+I) and TVT(R+I) with 2.86% and 3.82% margin on the MSVD dataset in the metric of CIDEr, respectively.

The results in Fig. 3 show that a fixed fusion weight should be replaced by adaptive attention weights, which introduces only a little computational cost. Attentive fusion method actually brings in a significant performance boost especially on the CIDEr score.

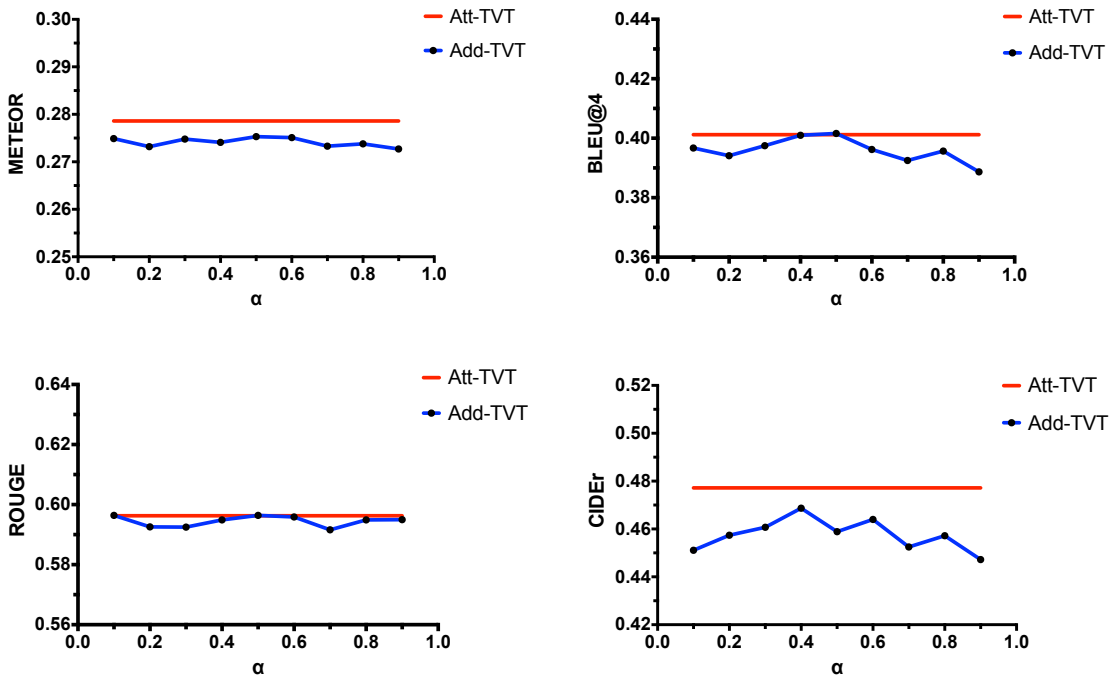

Figure 3: Results of Add-TVT with different fusion weights α and Att-TVT on MSR-VTT dataset.

5.3. Comparison with RNN Based Model

Table 5: Performance of transformer and RNN based models on the MSR-VTT dataset. All the experiments were performed using a GTX 1080Ti GPU.

Models	# Params	Training Time(sec)	BLEU	METEOR	ROUGE	CIDEr
BiLSTM(R)	26M	2717	38.38	26.72	59.43	43.46
Transformer(R)	24M	975	38.55	27.07	58.90	44.86
BiLSTM(N)	27M	3055	36.96	26.34	58.71	43.09
Transformer(N)	25M	1047	37.26	26.88	58.31	44.57

Table. 5 lists the training cost and performance of transformer and RNN based models under the same training strategy. The BiLSTM model uses a bidirectional lstm layer as its encoder and an unidirectional lstm layer as its decoder, equipped with attention mechanism proposed by Bahdanau et al. (2014). The Transformer model is the base model with 2 encoder and decoder layers for almost the same number of parameters with the BiLSTM model. The Transformer model achieves about 2.8$\times$ training speed compared with the BiLSTM model and better performance on three evaluation metrics.

5.4. Qualitative Results

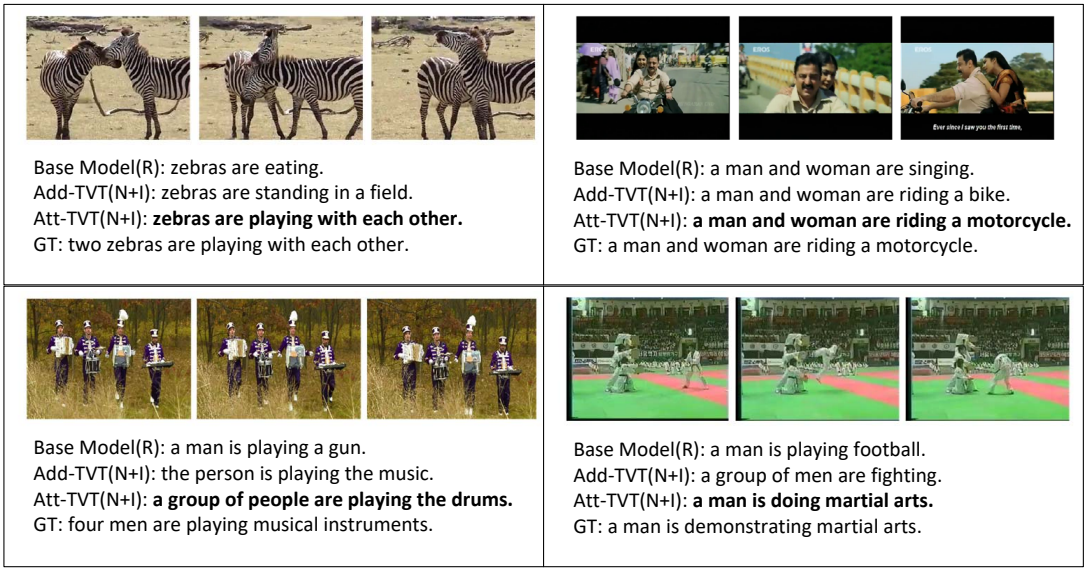

Figure 4: **Examples on MSVD testing set**. Here GT is short for ground truth, where a sample from candidates is shown.

Several examples on MSVD testing set generated by our proposed models are shown in Fig. 4. We see that without motion features our base model lacks the ability to capture the accurate action in videos. Our fusion methods, Add-TVT and Att-TVT, both generate correct descriptions of these samples, while Att-TVT performs slightly better than Add-TVT due to more detailed context of sentences caught by the attentive-fusion module.

To show the effectiveness of attentive-fusion module, Fig. 5 shows an example in the MSVD dataset and visualize the attention weights of the fusion block in the first layer of Att-TVT. It is clear that the nouns in the sentence, including words "woman", "liquid", and "glass", have a reasonable strong relationship with visual features extracted from every frame. The verb "mixing" is assigned with a higher weight of motion features. Note that the words "is", latter "a", and "⟨eos⟩" have a few cues from visual contents, while they secure more information from the context of the previously generated words.

6. Discussions

This paper presents a novel video captioning framework, i.e., Two-View Transformer (TVT) model. In the framework, TVT learns the long-term dependencies of sequential data based on the multi-head attention mechanism. The fusion blocks including Add-TVT and Att-TVT provide a novel way to exploit information from three different modalities containing features of frames, motions, and the previously generated words. Empirical results show that our framework achieves the state-of-the-art performance on the MSVD dataset and competitive results on the MSR-VTT dataset using visual and audio features. In the ablation study, we comprehensively demonstrate the effectiveness of our proposed fusion modules. An additional enlightenment of this work is that the Transformer

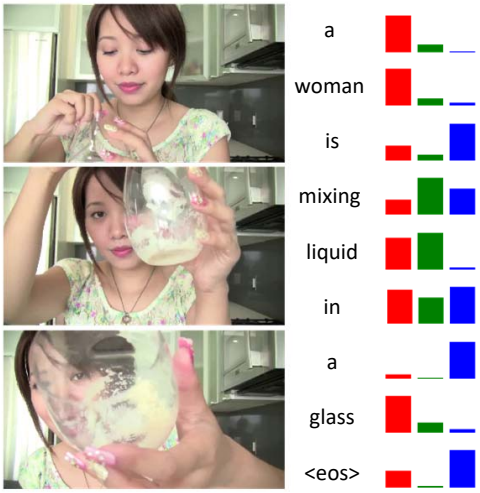

Figure 5: **Visualization of attention weights of the first layer of Att-TVT decoder.** The three bars show the attention weights of frame representation, motion representation, and the previously generated words, respectively.

network is able to well address the video captioning problem without the help of RNNs. In the future, more other modalities can be incorporated into the TVT framework. We also expect that this work may further inspire more future studies on the fusion approaches for video captioning.

Acknowledgments

This work was supported by NSFC (No. 61702448, 61672456) and the Fundamental Research Funds for the Central Universities (No. 2017QNA5008, 2017FZA5007). We thank all reviewers for their valuable comments.

References

Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C Lawrence Zitnick, and Devi Parikh. Vqa: Visual question answering. In *ICCV*, pages 2425–2433, 2015.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*, 2014.

Nicolas Ballas, Li Yao, Chris Pal, and Aaron Courville. Delving deeper into convolutional networks for learning video representations. *arXiv preprint arXiv:1511.06432*, 2015.

Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *CVPR*, pages 4724–4733, 2017.

Shizhe Chen, Jia Chen, Qin Jin, and Alexander Hauptmann. Video captioning with guidance of multimodal latent topics. In *ACM MM*, pages 1838–1846, 2017.

Xinlei Chen, Hao Fang, Tsung-Yi Lin, Ramakrishna Vedantam, Saurabh Gupta, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco captions: Data collection and evaluation server. *arXiv preprint arXiv:1504.00325*, 2015.

Kyunghyun Cho, Bart Van Merriënboer, Dzmitry Bahdanau, and Yoshua Bengio. On the properties of neural machine translation: Encoder-decoder approaches. *arXiv preprint arXiv:1409.1259*, 2014a.

Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*, 2014b.

Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:1412.3555*, 2014.

Michael Denkowski and Alon Lavie. Meteor universal: Language specific translation evaluation for any target language. In *Proceedings of the ninth workshop on statistical machine translation*, pages 376–380, 2014.

Yinpeng Dong, Hang Su, Jun Zhu, and Bo Zhang. Improving interpretability of deep neural networks with semantic information. In *CVPR*, pages 4306–4314, 2017.

Zhe Gan, Chuang Gan, Xiaodong He, Yunchen Pu, Kenneth Tran, Jianfeng Gao, Lawrence Carin, and Li Deng. Semantic compositional networks for visual captioning. In *CVPR*, 2017.

Sergio Guadarrama, Niveda Krishnamoorthy, Girish Malkarnenkar, Subhashini Venugopalan, Raymond Mooney, Trevor Darrell, and Kate Saenko. Youtube2text: Recognizing and describing arbitrary activities using semantic hierarchies and zero-shot recognition. In *ICCV*, pages 2712–2719, 2013.

Shawn Hershey, Sourish Chaudhuri, Daniel PW Ellis, Jort F Gemmeke, Aren Jansen, R Channing Moore, Manoj Plakal, Devin Platt, Rif A Saurous, Bryan Seybold, et al. Cnn architectures for large-scale audio classification. In *ICASSP*, pages 131–135, 2017.

Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8): 1735–1780, 1997.

Chiori Hori, Takaaki Hori, Teng-Yok Lee, Ziming Zhang, Bret Harsham, John R Hershey, Tim K Marks, and Kazuhiko Sumi. Attention-based multimodal fusion for video description. In *ICCV*, pages 4203–4212, 2017.

Qin Jin, Jia Chen, Shizhe Chen, Yifan Xiong, and Alexander Hauptmann. Describing videos using multi-modal fusion. In *ACM MM*, pages 1087–1091. ACM, 2016.

Will Kay, Joao Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, et al. The kinetics human action video dataset. *arXiv preprint arXiv:1705.06950*, 2017.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, pages 1097–1105, 2012.

Chin-Yew Lin. Rouge: A package for automatic evaluation of summaries. *Text Summarization Branches Out*, 2004.

Xiang Long, Chuang Gan, and Gerard de Melo. Video captioning with multi-faceted attention. *arXiv preprint arXiv:1612.00234*, 2016.

Jiasen Lu, Caiming Xiong, Devi Parikh, and Richard Socher. Knowing when to look: Adaptive attention via a visual sentinel for image captioning. In *CVPR*, 2017.

Pingbo Pan, Zhongwen Xu, Yi Yang, Fei Wu, and Yueting Zhuang. Hierarchical recurrent neural encoder for video representation with application to captioning. In *CVPR*, pages 1029–1038, 2016a.

Yingwei Pan, Tao Mei, Ting Yao, Houqiang Li, and Yong Rui. Jointly modeling embedding and translation to bridge video and language. In *CVPR*, pages 4594–4602, 2016b.

Yingwei Pan, Ting Yao, Houqiang Li, and Tao Mei. Video captioning with transferred semantic attributes. In *CVPR*, 2017.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting on association for computational linguistics*, pages 311–318. Association for Computational Linguistics, 2002.

Ramakanth Pasunuru and Mohit Bansal. Reinforced video captioning with entailment rewards. *arXiv preprint arXiv:1708.02300*, 2017.

Yunchen Pu, Martin Renqiang Min, Zhe Gan, and Lawrence Carin. Adaptive feature abstraction for translating video to text. 2018.

Vasili Ramanishka, Abir Das, Dong Huk Park, Subhashini Venugopalan, Lisa Anne Hendricks, Marcus Rohrbach, and Kate Saenko. Multimodal video description. In *ACM MM*, pages 1092–1096, 2016.

Zhiqiang Shen, Jianguo Li, Zhou Su, Minjun Li, Yurong Chen, Yu-Gang Jiang, and Xiangyang Xue. Weakly supervised dense video captioning. In *CVPR*, 2017.

Rakshith Shetty and Jorma Laaksonen. Frame-and segment-level features and candidate pool evaluation for video caption generation. In *ACM MM*, pages 1073–1076, 2016.

Jingkuan Song, Lianli Gao, Li Liu, Xiaofeng Zhu, and Nicu Sebe. Quantization-based hashing: a general framework for scalable image and video retrieval. *Pattern Recognition*, 75:175–187, 2018.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *JMLR*, 15(1):1929–1958, 2014.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. Sequence to sequence learning with neural networks. In *NIPS*, pages 3104–3112, 2014.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NIPS*, pages 6000–6010, 2017.

Ramakrishna Vedantam, C Lawrence Zitnick, and Devi Parikh. Cider: Consensus-based image description evaluation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4566–4575, 2015.

Subhashini Venugopalan, Marcus Rohrbach, Jeffrey Donahue, Raymond Mooney, Trevor Darrell, and Kate Saenko. Sequence to sequence-video to text. In *ICCV*, pages 4534–4542, 2015a.

Subhashini Venugopalan, Huijuan Xu, Jeff Donahue, Marcus Rohrbach, Raymond Mooney, and Kate Saenko. Translating videos to natural language using deep recurrent neural networks. In *NAACL*, pages 1494–1504, 2015b.

Oriol Vinyals, Alexander Toshev, Samy Bengio, and Dumitru Erhan. Show and tell: A neural image caption generator. In *CVPR*, pages 3156–3164, 2015.

Bairui Wang, Lin Ma, Wei Zhang, and Wei Liu. Reconstruction network for video captioning. *arXiv preprint arXiv:1803.11438*, 2018a.

Xin Wang, Wenhu Chen, Jiawei Wu, Yuan-Fang Wang, and William Yang Wang. Video captioning via hierarchical reinforcement learning. In *CVPR*, pages 4213–4222, 2018b.

Jun Xu, Tao Mei, Ting Yao, and Yong Rui. Msr-vtt: A large video description dataset for bridging video and language. In *CVPR*, 2016.

Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhudinov, Rich Zemel, and Yoshua Bengio. Show, attend and tell: Neural image caption generation with visual attention. In *ICML*, pages 2048–2057, 2015.

Li Yao, Atousa Torabi, Kyunghyun Cho, Nicolas Ballas, Christopher Pal, Hugo Larochelle, and Aaron Courville. Describing videos by exploiting temporal structure. In *ICCV*, pages 4507–4515, 2015.

Quanzeng You, Hailin Jin, Zhaowen Wang, Chen Fang, and Jiebo Luo. Image captioning with semantic attention. In *CVPR*, pages 4651–4659, 2016.

Haonan Yu, Jiang Wang, Zhiheng Huang, Yi Yang, and Wei Xu. Video paragraph captioning using hierarchical recurrent neural networks. In *CVPR*, pages 4584–4593, 2016.

Luowei Zhou, Yingbo Zhou, Jason J Corso, Richard Socher, and Caiming Xiong. End-to-end dense video captioning with masked transformer. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8739–8748, 2018.

Barret Zoph, Vijay Vasudevan, Jonathon Shlens, and Quoc V Le. Learning transferable architectures for scalable image recognition. *arXiv preprint arXiv:1707.07012*, 2017.