SBAT: Video Captioning with Sparse Boundary-Aware Transformer

Tao Jin[1] , Siyu Huang[2], Ming Chen[3], Yingming Li[1*], Zhongfei Zhang[4]
[1]College of Information Science & Electronic Engineering, Zhejiang University, China
[2]Baidu Research, China
[3]Alibaba Group, China
[4]Department of Computer Science, Binghamton University, USA
{jint_zju,yingming}@zju.edu.cn, huangsiyu@baidu.com, black.cm@alibaba-inc.com,
zzhang@binghamton.edu

Abstract

In this paper, we focus on the problem of applying the transformer structure to video captioning effectively. The vanilla transformer is proposed for uni-modal language generation task such as machine translation. However, video captioning is a multimodal learning problem, and the video features have much redundancy between different time steps. Based on these concerns, we propose a novel method called sparse boundary-aware transformer (SBAT) to reduce the redundancy in video representation. SBAT employs boundary-aware pooling operation for scores from multihead attention and selects diverse features from different scenarios. Also, SBAT includes a local correlation scheme to compensate for the local information loss brought by sparse operation. Based on SBAT, we further propose an aligned cross-modal encoding scheme to boost the multimodal interaction. Experimental results on two benchmark datasets show that SBAT outperforms the state-of-the-art methods under most of the metrics.

1 Introduction

Recently, the combination of vision and language attracts more and more attention [You *et al.*, 2016; Pan *et al.*, 2017; Antol *et al.*, 2015; Li *et al.*, 2019]. Video captioning is a valuable but challenging task in this topic, where the goal is to generate text descriptions for video data directly. The difficulties of video captioning mainly lie in the modeling of temporal dynamics and the fusion of multiple modalities.

Encoder-decoder structures are widely used in video captioning [Shen *et al.*, 2017; Aafaq *et al.*, 2019; Pei *et al.*, 2019; Wang *et al.*, 2019; Gan *et al.*, 2017]. In general, the encoder learns multiple types of features from raw video data. The decoder utilizes these features to generate words. Most encoder-decoder structures are built upon the long short-term memory (LSTM) unit, however, LSTM has two main drawbacks. First, LSTM-based decoder does not allow a parallel prediction of words at different time steps, since its hidden state is computed based on the previous one. Second, LSTM-based encoder has insufficient capacity to capture the long-range temporal correlations.

Figure 1: An example of redundancy between video frames.

To tackle these issues, [Chen *et al.*, 2018] and [Zhou *et al.*, 2018] proposed to replace LSTM with transformer for video understanding. Specifically, [Chen *et al.*, 2018] used multiple transformer-based encoders to encode video features and a transformer-based decoder to generate descriptions. Similarly, [Zhou *et al.*, 2018] utilized transformer for dense video captioning, [Zhou *et al.*, 2018] utilized a transformer-based encoder to detect action proposals and described them simultaneously with a transformer-based decoder. Different from LSTM, the self-attention mechanism in transformer correlates the features at any two time steps, enabling the global association of features. However, the vanilla transformer is limited in processing video features with much temporal redundancy like the example in Fig. 1. In addition, the cross-modal interaction between different modalities is ignored in the existing transformer-based methods.

Motivated by the above observations, we propose a novel method named sparse boundary-aware transformer (SBAT) to improve the transformer-based encoder and decoder architectures for video understanding. In the encoder, we employ sparse attention mechanism to better capture the global and local dynamic information by solving the redundancy between consecutive video frames. Specifically, to capture the global temporal dynamics, we divide all the time steps into n chunks according to the gradient values of attention logits and select n time steps with top-n gradient values. To capture the local temporal dynamics, we implement self-attention between r adjacent time steps. In the decoder, we also employ the boundary-aware strategy for encoder-decoder multihead attention. In addition, we implement cross-modal sparse attention following the self-attention layer to align multimodal features along temporal dimension. We conduct extensive empirical studies on two benchmark video captioning datasets. The quantitative, qualitative and ablation experimental results comprehensively reveal the effectiveness of

[*] Corresponding author.

our proposed methods.

The main contributions of this paper are three-folded:

(1) We propose the sparse boundary-aware transformer (SBAT) to improve the vanilla transformer. We use boundary-aware pooling operation following the preliminary scores of multihead attention and select the features of different scenarios to reduce the redundancy.

(2) We develop a local correlation scheme to compensate for the local information loss brought by sparse operation. The scheme can be implemented synchronously with the boundary-aware strategy.

(3) We further propose a cross-modal encoding scheme to align the multimodal features along the temporal dimension.

2 Related Work

As a popular variant of RNN, LSTM is widely used in existing video captioning methods. [Venugopalan *et al.*, 2015] utilized LSTM to encode video features and decode words. [Yao *et al.*, 2015] integrated the attention mechanism into video captioning, where the encoded features are given different attention weights according to the queries of decoder. [Hori *et al.*, 2017] further proposed a two-level attention mechanism for video captioning. The first level focuses on different time steps, and the second level focuses on different modalities. [Long *et al.*, 2018] and [Jin *et al.*, 2019b] detected local attributes and used them as supplementary information. [Jin *et al.*, 2019a] introduced cross-modal correlation into attention mechanism. Recently, [Chen *et al.*, 2018] proposed to replace LSTM with transformer in video captioning models. However, directly using transformer for video captioning has several drawbacks, i.e., the redundancy of video features and the lack of multimodal interaction modeling. In this paper, we propose a novel approach called sparse boundary-aware transformer (SBAT) to address these problems.

3 Transformer-based Video Captioning

Transformer [Vaswani *et al.*, 2017] is originally proposed for machine translation. Due to the effectiveness and scalability, transformer is employed in many other tasks including video captioning. A simple illustration of transformer-based video captioning model is shown in Fig. 2(a). The encoder and decoder both consist of multihead attention blocks and feed-forward neural network.

3.1 Encoder

Different from the uni-modal inputs of machine translation, the inputs of video captioning are typically multimodal. As shown in Fig. 2(a), two separate encoders process image and motion features, respectively. We use $I \in \mathbb{R}^{T_i \times d}$ and $M \in \mathbb{R}^{T_m \times d}$ to denote the image and motion features, respectively. Here we take the process of image encoding as an example. The self-attention layer is formulated as

$$\mathrm{SelfAttention}(I) = \mathrm{MultiHead}(I, I, I) \tag{1}$$

$$\mathrm{MultiHead}(I, I, I) = \mathrm{Cat}(\mathrm{head_1}, ..., \mathrm{head_h})W_1 \tag{2}$$

where "Cat" denotes concatenation operation, $W_1 \in \mathbb{R}^{d \times d}$ is a trainable variable. Multihead attention is a special variant of attention, where each head is calculated as

$$\mathrm{head_i} = \mathrm{Attention}(IW_i^Q, IW_i^K, IW_i^V) \tag{3}$$

where W_i^Q, W_i^K, and $W_i^V \in \mathbb{R}^{d \times \frac{d}{h}}$ are also trainable variables, "Attention" denotes scaled dot-product attention:

$$\mathrm{Attention}(Q, K, V) = \mathrm{softmax}(\frac{QK^{\mathbf{T}}}{\sqrt{d}})V \tag{4}$$

where d is dimension of Q and K. We adopt residual connection and layer normalization after the self-attention layer:

$$x = \mathrm{LayerNorm}(I + \mathrm{SelfAttention}(I)) \tag{5}$$

Every self-attention layer is followed by a feed-forward layer (FFN) that employs non-linear transformation:

$$\mathrm{FFN}(x) = \max(0, xW_2 + b_2)W_3 + b_3 \tag{6}$$

$$I^{'} = \mathrm{LayerNorm}(x + \mathrm{FFN}(x)) \tag{7}$$

where $W_2 \in \mathbb{R}^{d \times 4d}$, $b_2 \in \mathbb{R}^{4d}$, $W_3 \in \mathbb{R}^{4d \times d}$, and $b_3 \in \mathbb{R}^d$ are trainable variables. The encoded image features $I^{'}$ is the output of an encoder block. The encoded motion features $M^{'}$ are calculated in the same way.

3.2 Decoder

The decoder block consists of self-attention layer, enc-dec attention layer, and feed-forward layer. In the self-attention layer, the word embeddings of different time steps associate with each other, and we take the output features as queries. In the enc-dec multihead attention layer, the query first associates image and motion features to get two context vectors respectively, then generates the words. The feed-forward layer in decoder is the same as Eqns. 6 and 7. We also adopt residual connection and layer normalization after all the layers of the decoder.

Specifically, we use $E \in \mathbb{R}^{T_e \times d}$ to denote the embeddings of target words. To predict the word y_{t_e} at time step t_e, the self-attention layer is formulated as

$$E^{'}_{<t_e} = \mathrm{LayerNorm}(E_{<t_e} + \mathrm{SelfAttention}(E_{<t_e})) \tag{8}$$

where $E_{<t_e} \in \mathbb{R}^{(t_e-1) \times d}$ denotes the word embeddings of time steps less than t_e. The enc-dec attention layer is:

$$I_{t_e} = \mathrm{MultiHead}(E^{'}_{t_e\text{-}1}, I^{'}, I^{'}) \tag{9}$$

$$M_{t_e} = \mathrm{MultiHead}(E^{'}_{t_e\text{-}1}, M^{'}, M^{'}) \tag{10}$$

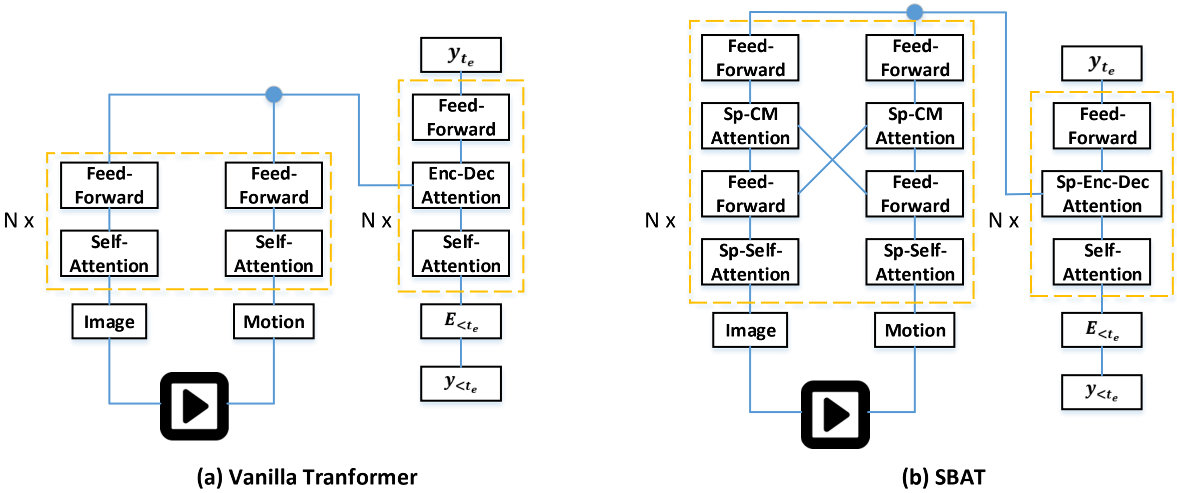
Following [Hori *et al.*, 2017], we employ a hierarchical attention layer for I_{t_e} and M_{t_e}:

$$V_{t_e} = \mathrm{LayerNorm}(E^{'}_{t_e\text{-}1} + \mathrm{MultiHead}(E^{'}_{t_e\text{-}1}, C_{t_e}, C_{t_e})) \tag{11}$$

$$C_{t_e} = [I_{t_e}, M_{t_e}] \tag{12}$$

$V^{'}_{t_e} = \mathrm{LayerNorm}(V_{t_e} + \mathrm{FFN}(V_{t_e}))$ denotes the output of feed-forward layer. We calculate the probability distributions of words as:

$$Pr(y_{t_e} | y_{<t_e}, I^{'}, M^{'}) = \mathrm{softmax}(W_p V^{'}_{t_e} + b_p) \tag{13}$$



(a) Vanilla Tranformer

(b) SBAT

Figure 2: (a) is the overall framework of Vanilla Transformer. It consists of multihead attention mechanism and feed-forward neural network. The features of different modalities are processed separately and the queries of decoder associate these features to generate words. N denotes the number of stacked blocks. (b) is the architecture of SBAT. It introduces sparse boundary-aware strategy (Sp) into all the multihead attention blocks in encoder and decoder. In addition, we learn cross-modal interaction after the first feed-forward layer in an encoder block.

where $I^{'}$ and $M^{'}$ denote the encoded video features. The optimization goal is to minimize the cross-entropy loss function defined as accumulative loss from all the time steps:

$$L = -\sum_{t_e=1}^{T_e} \log Pr(y^*_{t_e}|y^*_{<t_e}, I^{'}, M^{'}) \tag{14}$$

where $y^*_{t_e}$ denotes the ground-truth word at time step t_e.

4 Sparse Boundary-Aware Attention

Considering the redundancy of video features, it is not appropriate to compute attention weights using vanilla multihead attention. To solve the problem, we introduce a novel sparse boundary-aware strategy into the multihead attention. In Section 4.1, we introduce the sparse boundary-aware strategy. In Section 4.2, we provide the analysis of sparse boundary-aware strategy. In Section 4.3, we introduce the local correlation attention which compensates for the local information loss. In Section 4.4, we introduce an aligned cross-modal encoding scheme based on SBAT.

4.1 Sparse Boundary-Aware Pooling

We employ sparse boundary-aware strategy following the scaled dot-product attention logits. Specifically, the original logits are calculated as follows:

$$P = \frac{QK^{\mathbf{T}}}{\sqrt{d}} \tag{15}$$

where $Q \in \mathbb{R}^{T_q \times d}$ and $K \in \mathbb{R}^{T_k \times d}$ denote the query and key, respectively; d represents the dimension of Q and K. We utilize $P_{i,j}$ to represent the associated result of $Q_i \in \mathbb{R}^d$ and $K_j \in \mathbb{R}^d$. The discrete first derivative of P in the second dimension is obtained as follows:

$$P^{'}_{i,j} = \begin{cases} |P_{i,j}| & j = 0 \\ |P_{i,j} - P_{i,j-1}| & j \neq 0 \end{cases} \tag{16}$$

For time step i of the query, we choose top-n values in $P^{'}_i$, since the boundary of two scenarios always has high gradient value.

$$\mathcal{H}(P,n)_{i,j} = \begin{cases} P_{i,j} & P^{'}_{i,j} \geq c_i \\ -\infty & P^{'}_{i,j} < c_i \end{cases} \tag{17}$$

where c_i is the n-th largest value of $P^{'}_i$. We implement $\mathrm{softmax}$ function for the processed $\mathcal{H}(P,n)$.

Furthermore, to keep the time steps with large original logits, we define $P^*_{i,j}$ to replace $P^{'}_{i,j}$:

$$P^*_{i,j} = \alpha P^{'}_{i,j} + (1-\alpha)P_{i,j} \tag{18}$$

$P^{'}_{i,j}$ is a special variant of $P^*_{i,j}$ when $\alpha = 1$.

4.2 Theoretical Analysis of Boundary-Aware Pooling

Suppose we randomly choose one time step of $Q \in \mathbb{R}^{T_q \times d}$ as query $q \in \mathbb{R}^d$, the query q associates $K \in \mathbb{R}^{T_k \times d}$ at all the time steps. The logits of scaled dot-product attention are $[p_1, p_2, ..., p_{T_k}] \in \mathbb{R}^{T_k}$. We calculate the attention weight of each time step as:

$$a_\beta = \frac{\exp(p_\beta)}{\sum_{t_k=1}^{T_k} \exp(p_{t_k})} \tag{19}$$

To the best of our knowledge, there are about 3-5 scenarios on average in a ten-second video clip at a coarse granularity, like the example in Fig. 1. One-second clip usually contains 25 frames. Therefore, most frames in the same scenario are redundant. Existing methods sample the video to a fixed number of frames or directly reduce the frame rate. Although such methods are effective to some extent, there is still much redundancy in the scenarios that have a large number of time steps. The total attention weights of the scenarios with fewer time steps may be influenced. Specifically, we divide T_k time steps into two groups. The scenario one occupies T_1 time steps, the remaining T_2 time steps belong to scenario two. Suppose that the features of different time steps in the same scenario are the same, we obtain the total weights of two scenarios as follows:

$$A_{s_\gamma} = \frac{T_\gamma \exp(p_{s_\gamma})}{\sum_{o=1}^{2} T_o \exp(p_{s_o})}, \gamma \in \{1, 2\} \tag{20}$$

where A_{s_γ} denotes the total weight of scenario γ, p_{s_γ} denotes the associated logit. Suppose the query is related to scenario two ($p_{s_1} < p_{s_2}$) and $T_1 > T_2$, the ratio of T_1 to T_2 may influence the total attention weights (A_{s_1} and A_{s_2}) of two scenarios.

More concretely, we assume that T_1 is $0.75T_k$ and T_2 is $0.25T_k$. A_{s_1} and A_{s_2} are calculated as:

$$A_{s_1} = \frac{3\exp(p_{s_1})}{3\exp(p_{s_1}) + \exp(p_{s_2})} \tag{21}$$

$$A_{s_2} = \frac{\exp(p_{s_2})}{3\exp(p_{s_1}) + \exp(p_{s_2})} \tag{22}$$

if we apply sparse boundary-aware pooling strategy ($P^{'}_{i,j}$) for the logits and sample one time step in each scenario. Both A_{s_1} and A_{s_2} are transformed and the weight of scenario two obviously increases.

$$A^{'}_{s_1} = \frac{\exp(p_{s_1})}{\exp(p_{s_1}) + \exp(p_{s_2})} < A_{s_1} \tag{23}$$

$$A^{'}_{s_2} = \frac{\exp(p_{s_2})}{\exp(p_{s_1}) + \exp(p_{s_2})} > A_{s_2} \tag{24}$$

However, when the query is related to scenario one ($p_{s_1} > p_{s_2}$). It is not appropriate to reduce the proportion of scenario one. Therefore, we define $P^{*}_{i,j}$ to replace $P^{'}_{i,j}$ and select not only the boundaries of scenarios, but also the time steps with large original logits. Specifically, the number of selected steps is n, we sample two boundaries in the two scenarios and the remaining $n-2$ time steps belong to scenario one. $A^{'}_{s_1}$ is obtained as:

$$A^{'}_{s_1} = \frac{(n-1)\exp(p_{s_1})}{(n-1)\exp(p_{s_1}) + \exp(p_{s_2})} \tag{25}$$

for the increase from A_{s_1} in Eqn. 21 to $A^{'}_{s_1}$ in Eqn. 25, we just need to ensure that $n - 1 > 3$.

When the video clip has more than two scenarios, we also divide them into two groups. One has the scenarios with larger logits, the other has the remaining scenarios. The above analysis of two scenarios is approximately applicable in this situation.

4.3 Local Correlation

Since we employ sparse boundary-aware strategy for the attention logits, the local information between consecutive frames is ignored. We develop a local correlation scheme based on the multihead attention to compensate for the information loss. Formally, the original logits P are obtained following Eqn. 16. The correlation scheme is

$$\mathcal{H}_{\text{corr}}(P, n)_{i,j} = \begin{cases} P_{i,j} & |i-j| \leq r \\ -\infty & |i-j| > r \end{cases} \tag{26}$$

where r denotes the maximum distance of two frames and the correlation size is $2r$. In practice, the local correlation and boundary-aware correlation are utilized simultaneously.

4.4 Cross-Modal Scheme

Existing methods deal with different modalities separately in the encoder and ignore the interaction between different modalities. Here, we propose an aligned cross-modal scheme based on sparse boundary-aware attention. We divide the video into a fixed number of video chunks and then extract image and motion features from these chunks at the same intervals. Therefore, the feature vectors at the same step are extracted from the same video chunk. We directly apply our sparse boundary-aware attention to the aligned features. When the query is image modality, the key is motion modality, vice versa. Taking the former situation as an example, we compute the results of vanilla and boundary-aware cross-modal attentions as follows:

$$\text{CM-Attention}(I, M) = \text{MultiHead}(I, M, M) \tag{27}$$

$$\text{Sp-CM-Attention}(I, M) = \text{Sp-MultiHead}(I, M, M) \tag{28}$$

where CM denotes cross-modal.

5 Video Captioning with SBAT

We introduce the encoder-decoder structure combined with our sparse boundary-aware attention for video captioning. As shown in Fig. 2(b), we replace all the vanilla multihead attention blocks with boundary-aware attention blocks, except for the self-attention block for target word embeddings. Different from the original structure, an additional cross-modal attention layer is adopted following the self-attention layer in the encoder. In the decoder, we also introduce the boundary-aware attention into the enc-dec attention layer, but we set α to 0 in Eqn. 18 and do not use local correlation.

6 Experimental Methodology

6.1 Datasets and Metrics

We evaluate SBAT on two benchmark video captioning datasets, MSVD [Chen and Dolan, 2011] and MSR-VTT [Xu *et al.*, 2016]. Both the datasets are provided by Microsoft Research, and a series of state-of-the-art methods have been proposed based on these datasets in recent years. MSVD contains 1970 video clips and each video clip is about 10 to 25 seconds long and annotated with about 40 English sentences. MSR-VTT is larger than MSVD with 10000 YouTube video clips in total and each clip is annotated with 20 English sentences. We follow the commonly used protocol in the previous work and evaluate methods under four standard metrics including BLEU, ROUGE, METEOR, and CIDEr.

Method	MSVD				MSR-VTT			
	BLEU4	ROUGE	METEOR	CIDEr	BLEU4	ROUGE	METEOR	CIDEr
Vanilla Transformer	51.4	69.7	34.6	86.4	40.9	60.4	28.5	48.9
SBAT (w/o CM)	52.4	71.2	35.0	87.0	42.0	60.8	28.5	50.1
SBAT (w/o Local)	**53.5**	**72.3**	35.2	88.9	41.9	61.0	28.4	50.5
SBAT (Sample)	51.3	71.9	35.2	88.6	42.3	61.0	28.7	51.0
SBAT	53.1	**72.3**	**35.3**	**89.5**	**42.9**	**61.5**	**28.9**	**51.6**

Table 1: Evaluation results of our proposed methods. Note that we reproduce the results of Vanilla Transformer (TVT [Chen et al., 2018]). Due to different learning rate strategy, our implementation achieves better performances than the original TVT on MSR-VTT.

6.2 Data Preprocessing

We extract image features and motion features of video data. For image features, we sample video data to 80 frames and use the pre-trained Inception-ResNet-v2 [Szegedy *et al.*, 2017] model to obtain the activations from the penultimate layer. For motion features, we divide the raw video data into video chunks centered on the sampled frames and use the pre-trained I3D [Carreira and Zisserman, 2017] model to obtain the activations from the last convolutional layer. We implement a mean-pooling operation along the temporal dimension to get the motion features. On MSR-VTT, we also employ glove embeddings of the auxiliary video category labels to facilitate feature encoding.

6.3 Experimental Details

The hidden size is set to 512 for all the multihead attention mechanisms. The numbers of heads and attention blocks are 8 and 4, respectively. The value of α is set to 0.8 in the encoder and 0 in the decoder. In the training phase, we use Adam [Kingma and Ba, 2014] algorithm to optimize the loss function. The learning rate is initially set to 0.0001. If the CIDEr on validation set does not improve over 10 epochs, we change the learning rate to 0.00002. The batch size is set to 32. In the testing phase, we use the beam-search method with a beam-width of 5 to generate words. We use the pre-trained word2vec embeddings to initialize the word vectors. Each word is represented as a 300-dimension vector.

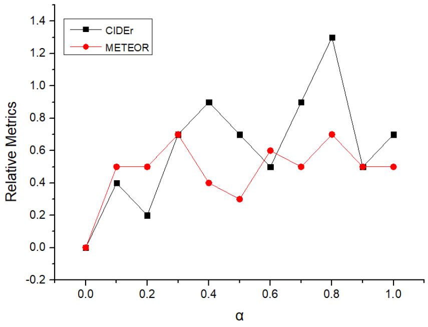


Figure 3: Effect of α on MSR-VTT. We show the relative results on METEOR and CIDEr. Specifically, we set $\alpha = 0$ as the baseline.

7 Experimental Results

7.1 Impact of Sparse Boundary-Aware Attention

We first evaluate the effectiveness of different variants of SBAT, as shown in Table 1. Vanilla Transformer and SBAT denote the models in Fig. 2(a) and (b). SBAT (w/o CM) denotes the model without aligned cross-modal attention. SBAT (w/o Local) denotes the model without local correlation in the encoder. SBAT (Sample) denotes the model with equidistant sampling for all the time steps, rather than our boundary-aware operation.

In Table 1, Vanilla Transformer achieves relatively bad results on both datasets. However, when we adopt boundary-aware or equidistant sampling strategies in the multihead attentions, the performances are obviously improved. SBAT with boundary-aware attention, local correlation, and aligned cross-modal interaction achieves promising results under all the metrics. The comparison between SBAT (w/o CM) and SBAT shows that the cross-modal interaction provides useful cues for generating words. The comparison between SBAT (w/o Local) and SBAT shows that the local correlation can make up the loss of local information. Comparing SBAT and SBAT (Sample), although equidistant sampling reduces the feature redundancy to some extent, the ratio between different scenarios is not considered, while SBAT solves this problem effectively.

7.2 Comparison of $P^{'}$ and P

To evaluate the impact of P^{*} and find an appropriate ratio between $P^{'}$ and P, we adjust the value of α in Eqn. 18 based on SBAT. The experimental results are shown in Fig. 3. Note that we only adjust the value of α in the encoder, and the value of α in the decoder is always 0. We observe that P^{*} with $\alpha = 0.8$ achieves the best performances on both METEOR and CIDEr. In addition, only using original logits P ($\alpha = 0$) shows the worst performances, indicating that our proposed boundary-aware strategy $P^{'}$ is a significant boost for the transformer-based video captioning model.

7.3 Comparison with State-of-the-art

Table 2 shows the results of different methods on MSVD and MSR-VTT. For a fair comparison, we compare SBAT with the methods which also use image features and motion features. The comparison methods include TVT [Chen *et al.*, 2018], MGSA [Chen and Jiang, 2019], Dense Cap [Shen *et al.*, 2017], MARN [Pei *et al.*, 2019], GRU-EVE [Aafaq *et al.*, 2019], POS-CG [Wang *et al.*, 2019], SCN [Gan *et al.*, 2017]. In Table 2, SBAT shows better or competitive performances compared with the state-of-the-art methods. On MSR-VTT, SBAT outperforms TVT, MGSA, Dense Cap, MARN, GRU-EVE, POS-CG on all the metrics. In partic-

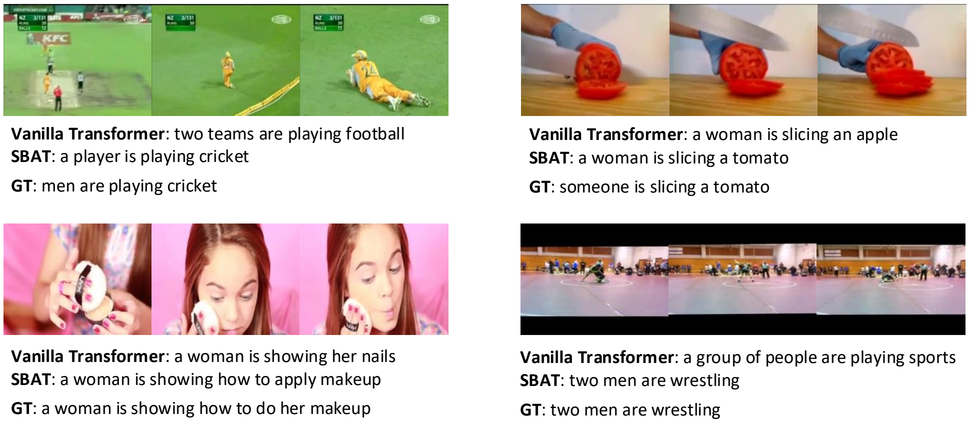

Vanilla Transformer: two teams are playing football
SBAT: a player is playing cricket
GT: men are playing cricket
Vanilla Transformer: a woman is slicing an apple
SBAT: a woman is slicing a tomato
GT: someone is slicing a tomato
Vanilla Transformer: a woman is showing her nails
SBAT: a woman is showing how to apply makeup
GT: a woman is showing how to do her makeup
Vanilla Transformer: a group of people are playing sports
SBAT: two men are wrestling
GT: two men are wrestling

Figure 4: Some qualitative results of the video clips on the test sets of MSR-VTT and MSVD. We provide the ground-truth description and the generated descriptions of Vanilla Transformer and SBAT for each video clip.

Dataset	Method	B	R	M	C
MSR-VTT	TVT	40.1	61.1	28.2	47.7
	MGSA	42.4	-	27.6	47.5
	Dense Cap	41.4	61.1	28.3	48.9
	MARN	40.4	60.7	28.1	47.1
	GRU-EVE	38.3	60.7	28.4	48.1
	POS-CG	42.0	61.1	28.1	49.0
	SBAT	**42.9**	**61.5**	**28.9**	**51.6**
MSVD	TVT	**53.2**	-	35.2	86.8
	SCN	51.1	-	33.5	77.7
	MARN	48.6	71.9	35.1	**92.2**
	GRU-EVE	47.9	71.5	35.0	78.1
	POS-CG	52.5	71.3	34.1	88.7
	SBAT	53.1	**72.3**	**35.3**	89.5

Table 2: Evaluation results of video captioning, where B, R, M, C denote BLEU4, ROUGE, METEOR, CIDEr, respectively.

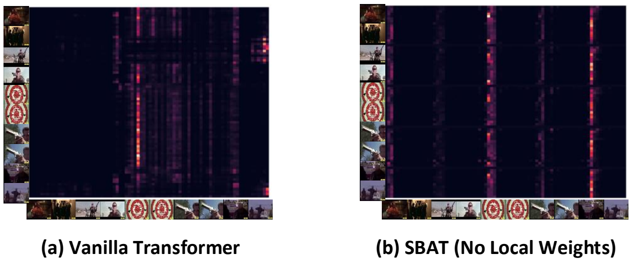


Figure 5: Visualization of attention mechanism. (a) and (b) denote Vanilla Transformer and SBAT, respectively. x and y axes both denote continuous video frames. The generated descriptions of two methods are both "a man is shooting a gun".

ular, SBAT achieves 51.6% on CIDEr, making an improvement of 2.6% over POS-CG. On MSVD, SBAT outperforms SCN, GRU-EVE, POS-CG on all the metrics and has a better overall performance than TVT and MARN.

7.4 Visualization of Attention Mechanism

To further illustrate the effectiveness of SBAT, we conduct a case study and visualize the attention distributions of SBAT and Vanilla Transformer. In Fig. 5, we take a video clip for example. Note that we only visualize the weights of image modality for convenience, and we do not show the local attention weights. Fig. 5(a) shows that the attention weights of Vanilla Transformer are dispersed and Vanilla Transformer has a poor ability to detect the boundary of different scenarios. While Fig. 5(b) shows that (1) SBAT has more sparse attention weights than Vanilla Transformer; (2) SBAT accurately detects the scenario boundaries.

7.5 Qualitative Results

Fig. 4 shows several qualitative examples. We compare the descriptions generated by Vanilla Transformer, SBAT, and ground truth (GT). With the help of redundancy reduction and a better usage of global and local information, SBAT generates more accurate descriptions that are close to GT.

8 Conclusion

In this paper, we have proposed a new method called sparse boundary-aware transformer (SBAT) for video captioning. Specifically, we have proposed sparse boundary-aware strategy for improving the attention logits in vanilla transformer. Combined with local correlation and cross-modal encoding, SBAT can effectively reduce the feature redundancy and capture the global-local video information. The quantitative, qualitative, and ablation experiments on two benchmark datasets have demonstrated the advantage of SBAT.

Acknowledgments

This work is supported in part by Science and Technology Innovation 2030 –"New Generation Artificial Intelligence" Major Project No.(2018AAA0100904), National Key R&D Program of China (No. 2018YFB1403600), NSFC (No. 61672456, 61702448, U19B2043), Artificial Intelligence Research Foundation of Baidu Inc., the funding from HIKVision and Horizon Robotics, and ZJU Converging Media Computing Lab.

References

[Aafaq *et al.*, 2019] Nayyer Aafaq, Naveed Akhtar, Wei Liu, Syed Zulqarnain Gilani, and Ajmal Mian. Spatio-temporal dynamics and semantic attribute enriched visual encoding for video captioning. In *CVPR*, 2019.

[Antol *et al.*, 2015] Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C Lawrence Zitnick, and Devi Parikh. Vqa: Visual question answering. In *ICCV*, 2015.

[Carreira and Zisserman, 2017] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *CVPR*, 2017.

[Chen and Dolan, 2011] David L Chen and William B Dolan. Collecting highly parallel data for paraphrase evaluation. In *ACL*, 2011.

[Chen and Jiang, 2019] Shaoxiang Chen and Yu-Gang Jiang. Motion guided spatial attention for video captioning, 2019.

[Chen *et al.*, 2018] Ming Chen, Yingming Li, Zhongfei Zhang, and Siyu Huang. Tvt: Two-view transformer network for video captioning. In *ACML*, 2018.

[Gan *et al.*, 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.

[Hori *et al.*, 2017] 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*, 2017.

[Jin *et al.*, 2019a] Tao Jin, Siyu Huang, Yingming Li, and Zhongfei Zhang. Low-rank hoca: Efficient high-order cross-modal attention for video captioning. *arXiv preprint arXiv:1911.00212*, 2019.

[Jin *et al.*, 2019b] Tao Jin, Yingming Li, and Zhongfei Zhang. Recurrent convolutional video captioning with global and local attention. *Neurocomputing*, 2019.

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

[Li *et al.*, 2019] Xiangpeng Li, Jingkuan Song, Lianli Gao, Xianglong Liu, Wenbing Huang, Xiangnan He, and Chuang Gan. Beyond rnns: Positional self-attention with co-attention for video question answering. In *AAAI*, 2019.

[Long *et al.*, 2018] Xiang Long, Chuang Gan, and Gerard de Melo. Video captioning with multi-faceted attention. *TACL*, 2018.

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

[Pei *et al.*, 2019] Wenjie Pei, Jiyuan Zhang, Xiangrong Wang, Lei Ke, Xiaoyong Shen, and Yu-Wing Tai. Memory-attended recurrent network for video captioning. In *CVPR*, 2019.

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

[Szegedy *et al.*, 2017] Christian Szegedy, Sergey Ioffe, Vincent Vanhoucke, and Alexander A Alemi. Inception-v4, inception-resnet and the impact of residual connections on learning. In *AAAI*, 2017.

[Vaswani *et al.*, 2017] 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*, 2017.

[Venugopalan *et al.*, 2015] Subhashini Venugopalan, Marcus Rohrbach, Jeffrey Donahue, Raymond Mooney, Trevor Darrell, and Kate Saenko. Sequence to sequence-video to text. In *ICCV*, 2015.

[Wang *et al.*, 2019] Bairui Wang, Lin Ma, Wei Zhang, Wenhao Jiang, Jingwen Wang, and Wei Liu. Controllable video captioning with pos sequence guidance based on gated fusion network. In *ICCV*, 2019.

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

[Yao *et al.*, 2015] Li Yao, Atousa Torabi, Kyunghyun Cho, Nicolas Ballas, Christopher Pal, Hugo Larochelle, and Aaron Courville. Describing videos by exploiting temporal structure. In *ICCV*, 2015.

[You *et al.*, 2016] Quanzeng You, Hailin Jin, Zhaowen Wang, Chen Fang, and Jiebo Luo. Image captioning with semantic attention. In *CVPR*, 2016.

[Zhou *et al.*, 2018] Luowei Zhou, Yingbo Zhou, Jason J Corso, Richard Socher, and Caiming Xiong. End-to-end dense video captioning with masked transformer. In *CVPR*, 2018.