Variational Stacked Local Attention Networks for Diverse Video Captioning

Tonmoay Deb   Akib Sadmanee   Kishor Kumar Bhaumik   Amin Ahsan Ali
M Ashraful Amin   A K M Mahbubur Rahman

Artificial Intelligence and Cybernetics (AGenCy) Lab, Independent University, Bangladesh

tonmoay.nsu@gmail.com, {1620274, 1621366, aminali, aminmdashraful, akmmrahman}@iub.edu.bd

Abstract

While describing spatio-temporal events in natural language, video captioning models mostly rely on the encoder's latent visual representation. Recent progress on the encoder-decoder model attends encoder features mainly in linear interaction with the decoder. However, growing model complexity for visual data encourages more explicit feature interaction for fine-grained information, which is currently absent in the video captioning domain. Moreover, feature aggregations methods have been used to unveil richer visual representation, either by the concatenation or using a linear layer. Though feature sets for a video semantically overlap to some extent, these approaches result in objective mismatch and feature redundancy. In addition, diversity in captions is a fundamental component of expressing one event from several meaningful perspectives, currently missing in the temporal, i.e., video captioning domain. To this end, we propose Variational Stacked Local Attention Network (VSLAN), which exploits low-rank bilinear pooling for self-attentive feature interaction and stacking multiple video feature streams in a discount fashion. Each feature stack's learned attributes contribute to our proposed diversity encoding module, followed by the decoding query stage to facilitate end-to-end diverse and natural captions without any explicit supervision on attributes. We evaluate VSLAN on MSVD and MSR-VTT datasets in terms of syntax and diversity. The CIDEr score of VSLAN outperforms current off-the-shelf methods by 7.8% on MSVD and 4.5% on MSR-VTT, respectively. On the same datasets, VSLAN achieves competitive results in caption diversity metrics.

1. Introduction

With the trailing success of Recurrent Neural Networks (RNN), sequence-to-sequence (seq2seq) [33] was introduced, which incorporates Long Short Term Memory (LSTM), a variant of RNN to generate captions from encoded visual features of Convolutional Neural Network (CNN). However, addressing the drawback of global features (by seq2seq) on missing important temporal context, attention mechanism [40] was proposed in spatio-temporal domain to improve the decoder during step-by-step word prediction. Later several improvements on attention mechanism [25, 5] have evolved for processing visual features to capture both spatial and temporal relation effectively.

Most existing works exploit attentive features from the encoder stage as hidden input to the decoder. Although it is a widely used norm, it is limited to perform in a single feature stream. Because the attention distribution is unimodal and weights all feature streams in an identical fashion. To tackle this issue, several studies from visual recognition [17] and question-answering [15] studied capturing richer feature semantics by facilitating better interaction. Bilinear Pooling [7] have been widely studied to be efficiently computed regardless of the number of interaction order with a cost of expensive computation. Recently, [16] proposed linearly mapped low-rank bilinear pooling using the Hadamard product, which solved this issue. Following that, this concept is adopted by several domains [21]. We similarly incorporate bilinear pooling as a building block of our local and global attention module. On top of that, we propose a novel self-attention architecture to attend specific feature sets directly from training loss. To our best knowledge, the concept of exploiting temporal structure, e.g., for video captioning, for generating fine-grained feature set with bilinear pooling, is yet unexplored.

Another research track focuses on visual feature sophistication with an intuition that richer features can lead to subtle clues for guiding the decoder. A straightforward approach is to capture features from different architectures and unify them into a fixed dimension. The networks can be frame-level 2D CNN and clip-level 3D CNN features. We argue that word prediction models can better differentiate between the candidate vocabularies when features from multiple origins are fed into the decoder. Also, the aggregation approach of the works above fuse features at a single stage, either by concatenation or with a linear layer. Yet, hierarchical dependence between the streams was ignored. In this scenario, a feature stream can hardly identify
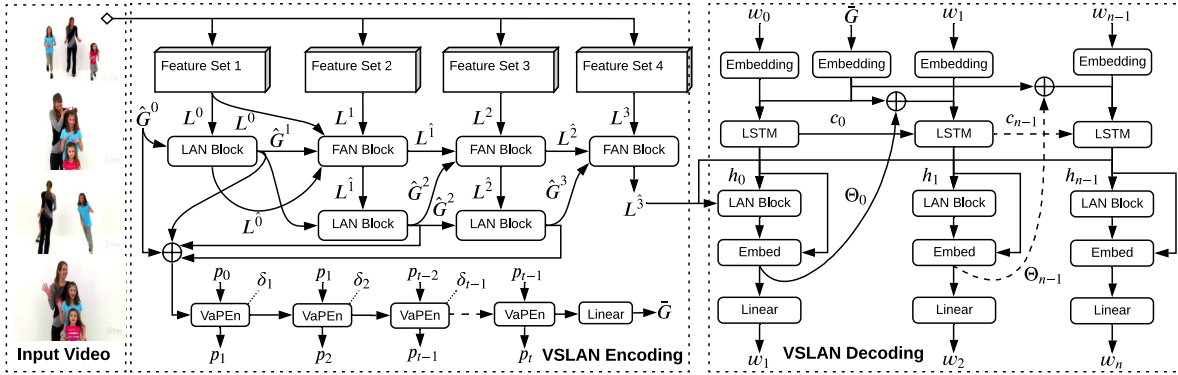
Figure 1. Overview of Variational Stacked Local Attention Network (VSLAN) comprising of Local Attention Network (LAN), Feature Aggregation Network (FAN), and Variational POS Encoder (VaPEn). The final Local feature set from FAN is utilized to query caption words at the decoding stage, done by another dedicated LAN block. VaPEn is built on top of the output of each FAN block. The final output from VaPEn's final node accounts for global POS feature, $\overline{G}$, further added with Θ_n for influencing the diversity on decoder.

attributes already addressed by the prior streams, resulting in redundancy. Given that feature, compression [36], and model distillation [19] are active research domains, unifying redundant features from multiple streams is worthwhile to explore. This work proposed a module that attends to specific feature sets from respective streams in a forward pass and learns to discount the redundant information end-to-end without explicit supervision.

Amid notable progress in video captioning, the main focus was on generating deterministic captions, i.e., single caption output from a video. Although one caption is expected to represent overall temporal dynamics, it is rarely true from the real-world perspective. Moreover, generating multiple-yet-relevant captions can semantically diversify the overall visual understandings from a video, which may also be useful in video query and retrieval. However, extracting probabilistic, i.e., diverse captions from videos, is also an unexplored problem to the best of our knowledge. Although some progress has been made for static images [2], the main challenge for videos remains in attending to a temporal window and encoding the respective features into a meaningful representation. To tackle this issue, we include diverse Parts of Speech (POS) generation to influence the final caption structure. Because POS information holds the basic pattern of grammar composition to control the final sentence structure [2, 35] and can be easily evaluated with existing metrics, e.g., BLEU [22]. Thus, we propose an approach to generate diverse POS from the fine-grained feature streams. The final representation from POS further works as a global feature of the caption decoder that also influences sentence diversity.

In summary, we propose Variational Stacked Local Attention Network (VSLAN), comprising of Local Attention Network (LAN), Feature Aggregation Network (FAN) modules to exploit the potential of visual features from

several off-the-shelf pre-trained models during captioning. The LAN block utilizes bilinear pooling to attend the related clips of a video during captioning, results in richer and highly attended information. While a new feature set arrives, the FAN block aggregates them so that it discounts previously learned information from preceding LAN blocks, which ensures compact visual features. The LAN-FAN blocks are propagated by aggregating specialized visual feature sets and leveraged into a decoder for captioning. We enforce diversity in sentences by proposing a Variational POS Encoder (VaPEn) using the fine-grained representation from each feature stream. In the decoding stage, the local features from the last encoder layer are used to query the next words, whereas the global features, i.e., stochastic VaPEn features, influence that query stream for diverse caption syntax. Figure 1 illustrates the primary view of VSLAN. Therefore, our key contributions are:

- We propose a novel, end-to-end video captioning architecture, VSLAN, which can attend to both local (in each feature stream) and global (in-between feature streams) spatio-temporal attributes without explicit supervision to capture richer semantics for decoding robust and diverse captions.

- We reformulate the existing feature aggregation methods by introducing a novel network to propagate the important information from the new feature streams in a discount fashion.

- We introduce diversity in video captioning by formulating and utilizing an end-to-end POS generator with temporal context, VaPEn, to influence decoder query.

- We perform extensive analysis to demonstrate the efficacy of our proposed architecture. VSLAN outperforms state-of-the-art methods by a notable margin on
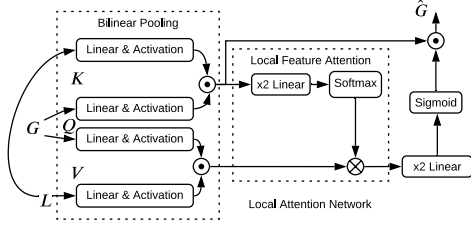
Figure 2. Overview of Local Attention Network (LAN).

Figure 3. Overview of Feature Aggregation Network (FAN).

two datasets and attains consistent diversity performance based on commonly used evaluation protocol.

2. Related Works

Sequence Learning and Attention approaches use encoder-decoder to learn a shared representation of visual and linguistic features. Recently, [42] proposed densely connected recurrent neural networks to facilitate attention and improve accuracy. The SibNet architecture by [19] builds on top of the encoder-decoder network, where two CNN branches are utilized to encode the videos. Another track of attention-based video captioning focuses on finding out salient region [37] in each video frame using attention and simultaneously learn the spatio-temporal representation over time. [38] did it more explicitly by storing attributes in a hierarchical memory and utilize while captioning. However, due to vanishing gradient issue by recurrent networks, these approaches fall short for long-tail videos. In our work, we overcome this limitation by designing the caption decoder utilizing key-query-value representation for each recurrent node and training a shareable LAN module.

Feature Aggregation: [32] used a hyperparameter to calculate the weighted sum of Optical Flow and RGB features at each step. [4, 10] enhanced encoder representation by including multimodal, e.g., audio, category features along with the visual stream. [35] designed cross-gating (CG) block to fuse RGB and Flow features toward captioning. Recently, to introduce feature invariance, [29] proposed encoding videos twice to avoid irrelevant temporal information. We argue that this strategy will fall short on online video captioning. Our approach self-attends each feature stream and learns redundancy while among multiple streams, results in fine-grained aggregated features.

Bilinear Pooling in image captioning has been recently explored by [21]. However, their inherent intuition and architecture are different from ours. For example, they applied squeeze-excitation [12] over their one of the key-query feature split. However, VSLAN consists of a straightforward element-wise multiplication between the key-query and the value's linearly pooled feature, extended with two linear layers and a sigmoid to connect the key-query representation. They utilize a learned decoder from the visual features for intra-modal (text and image) queries during decoding.
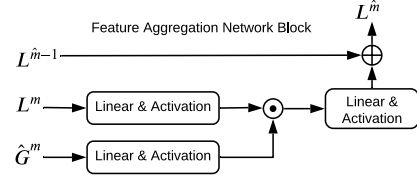
In contrast, we use a dedicated LAN decoder, shared in the decoding stage by learning from scratch.

Diversity in Captioning had initial success by deterministic Diverse Beam Search (DBS) [34]. [2] recently incorporated a stochastic latent space learning like variational RNN (VRNN) [6] and applied it for predicting words from images. Although there have been no explicit experiments on videos, [41] recently introduced an approach for generating captions from example sentences in addition to videos. Their model incorporates the syntax structure of the example sentence during caption generation. In this case, even controllable, it requires additional data to train and becomes unstable with unseen examples. Our diverse caption generation focuses more on inherent syntactic structure, i.e., POS generation, which attends to specific visual features and influences the decoder to generate diverse captions.

3. Variational Stacked Local Attention Nets

3.1. Video Encoder

Given $M = (L^1, L^2, ..., L^{M-1})$ a set of local feature set from a video, layer $L^m = (l_1^m, l_2^m, ..., l_N^m)$, where N is the video length and l_i^m is extracted feature from i-th clip, the goal is to generate a description $S = (w_1, w_2, ..., w_n)$ of n length, by sampling each word, w, from the vocabulary ϑ.

Local Attention Network (LAN): For Query $Q \in \mathbb{R}^q$, Key $K = (k_1, k_2, ..., k_N)$, and Value $V = (v_1, v_2, ..., v_N)$, where $k_i \in \mathbb{R}^k$ and $v_i \in \mathbb{R}^v$, we perform low-rank bilinear pooling on Q and K as:

$$\beta_i^K = \sigma\left(W_Q^K Q\right) \odot \sigma\left(W_K k_i\right) \quad (1)$$

Here, $W_Q^K \in \mathbb{R}^{z \times q}$ and $W_K \in \mathbb{R}^{z \times k}$ are embedding metrics, projects Q and K into a unified dimension $\mathbb{R}^z$. σ is the activation function. The projected Q performs element-wise multiplication, $\odot$, with each projected key k_i. Similarly, for Value V, the pooling

$$\beta_i^V = \sigma\left(W_Q^V Q\right) \odot \sigma\left(W_V v_i\right) \quad (2)$$

Here, $W_Q^V \in \mathbb{R}^{z \times q}$ and $W_V \in \mathbb{R}^{z \times v}$ projects Q and K into a unified dimension $\mathbb{R}^z$. $\beta^K = (\beta_1^K, \beta_2^K, ..., \beta_N^K)$ and $\beta^V = (\beta_1^V, \beta_2^V, ..., \beta_N^V)$ are bilinear Key, Value attention distribution for network layer m, holding $2^{m^{th}}$ order feature interaction properties.

To capture semantic information, β^K and β^V are propagated with two network streams to generate clip-wise, local attention, and global attention distribution, respectively. The local information $\beta_i^{s'}$ is captured as:

$$\beta_i^{s'} = \sigma\left(W_{\beta^V}^s \sigma\left(W_{\beta^V}^k \beta_i^K\right)\right) \tag{3}$$

Here, $W_{\beta^V}^k \in \mathbb{R}^{z' \times z}$ and $W_{\beta^V}^s \in \mathbb{R}^{1 \times z'}$. The fist linear layer projects the local feature into a richer representation dimension z', where the second lowers the dimension to 1 for probabilistic local feature $\beta^s = softmax\left(\beta^{s'}\right)$

Later, a weighted sum is performed on bilinear pooled value β^V with the distribution β^s as $\beta^l = \sum_{i=1}^{N} \beta_i^s \beta_i^V$. The attended local features represent key clips that are relevant to the captioning query Q. $\beta^l \in \mathbb{R}^{1 \times z}$ is further passed through two linear layers, followed by a sigmoid to generate fine-grained global information $\beta^g = \sigma\left(W_{\beta^l}^2\left(W_{\beta^l}^1 \beta^l\right)\right)$

Again, $W_{\beta^l}^1 \in \mathbb{R}^{x \times z}$ and $W_{\beta^l}^2 \in \mathbb{R}^{z \times x}$ are the projection metrics where $x < z$. The lower dimension of x squeezes global information into a latent representation and reconstructs the previous dimension accordingly. σ is sigmoid activation function. The concept of $W_{\beta^l}^1 \beta^l$ is similar to feature compression technique, concerning robust representation at the decoding stage. $\beta^g \in \mathbb{R}^{1 \times z}$ is further combined with key-query pairs, β_i^K, to compute stabilized global feature $\hat{G}$ with the network as $\hat{G} = \sum_{i=1}^{N} \beta_i^l \beta_i^K$

$\hat{G}$ is the enhanced representation using self-attention, which incorporates local information with explicit feature interaction compared to the regular self-attention that utilizes only first-order properties as mentioned in Figure 2. We exploit this pooling strategy over multiple layers using feature aggregation, described below.

Feature Aggregation Network (FAN): Each FAN block uses low-rank bilinear pooling to capture novel information from the new feature stream and integrates with the current.

For pre-trained CNN features $L^{\hat{m}-1}$, the global information is captured by $\hat{G}^m$ using LAN. When a new CNN feature set L^m appears, a low-rank biliear pooling operation is being performed between L^m and $\hat{G}^m$ as

$$\beta_i^L = \sigma\left(W_G^m \hat{G}^m\right) \odot \sigma\left(W_L^m L_i^m\right) \tag{4}$$

Here, for layer m, $W_G^m \in \mathbb{R}^{y \times z}$ and $W_L^m \in \mathbb{R}^{y \times v}$ projects $\hat{G}^m$ and L^m into a fixed dimension y to perform $\odot$ operation. β_i^L captures the relevant information from L^m by exploiting low-rank bilinear pooling using previous global feature $\hat{G}^m$. β_i^L is further passed through a linear layer with activation for $\beta_i^{L'} = \sigma\left(W_{L'}^m \beta_i^L\right)$. Here, $W_{L'}^m \in \mathbb{R}^{z \times y}$ projects features into z dimension. The final representation $L^{\hat{m}}$ from the FAN block is comprised of $LayerNorm$ over

pairwise sum of projected feature $\beta_i^{L'}$ and previous attentive local features $L^{\hat{m}-1}$

$L^{\hat{m}} \in \mathbb{R}^{N \times z}$ is a bridge between the preceding and current local information, respectively, further passed to LAN block for richer, more explicit attention distribution, resulting distilled feature representation. For the initial FAN block, we consider $L^{\hat{m}-1} = L^m = L^0$.

As discussed earlier, out of M feature sets, FAN discounts redundant information present in the prior feature propagation. During the training stage, the network distributes non-linear attentive weights to the representation that are more relevant to the caption. Following the recommendation from [1, 11], we exploit object-wise and temporal convolutional networks from the uniformly sampled frames and clips to represent subject and object properties, respectively. We set the visual features in L^m as mentioned in Figure 3, where LAN and FAN learn weights end-to-end from the caption loss without explicit guidance (fundamental mechanism of self-attention).

Variational POS Encoder (VaPEn): We generate stochastic POS sequences from the global visual features $\tilde{G} = \left[\hat{G}^0; \hat{G}^1; ...; \hat{G}^{M-1}\right] \in \mathbb{R}^{q=(M-1)*z}$. Here, $[.;.]$ denotes concatenation operation. $\tilde{G}$ is set at the intial hidden state to the encoder LSTM. We incorporate VRNN [6], which learns the latent distributions δ_t over time steps t of a recurrent network. During the generation stage, at first $\delta_t \sim \mathcal{N}(\mu_{0,t}, \sigma_{0,t})$ distribution is conditioned on the prior LSTM state variable s_{t-1} and decodes POS $p_t|\delta_t \sim \mathcal{N}(\mu_{\delta,t}, \sigma_{\delta,t})$ as output using the distribution at that time step. Modifying the original Variational Autoencoder, the recurrent generator prior distribution $r_\theta(p_{\leq T}|\delta_{\leq T}) = \prod_{t=1}^{T} r_\theta(p_t|\delta_{\leq t}, p_{<t}) r_\theta(\delta_t|p_{<t}, \delta_{<t})$. During the inference, δ_t is updated from the generated POS p_t by following posterior distribution $q_\varphi(\delta_{\leq t}, p_{\leq t}) = \prod_{t=1}^{T} q_\varphi(\delta_t|p_{\leq t}, \delta_{<t})$. Here θ and φ are the recurrent network, i.e., LSTM parameters, updated end-to-end. The overall objective function comprises reaching variational lower bound based on prior and encoder according to the below objective:

$$\mathbb{E}_{q_\varphi(\delta_{\leq T}|p_{\leq T})}\left[\sum_{t=1}^{T}\left(-KL(q_\varphi(\delta_t|\delta_{\leq t}, p_{\leq t})||r_\theta(\delta_t|\delta_{<t}, p_{<t}))\right.\right.$$
$$\left.\left. + \log r_\theta(p_t|\delta_{\leq t}, p_{<t})\right)\right]$$

Here $KL(q_\varphi||r_\theta)$ is Kullback–Leibler divergence between two distributions, q_φ and r_θ, measures nonsymmetric difference, aimed to maximize w.r.t the parameters. We refer readers to [6] for more details. The last hidden state, s_t is further propagated to the decoder as and shared across the decoding steps as stated in the next section.

3.2. Caption Decoder

The fundamental goal of a caption decoder is to generate a sequence of words conditioned on the encoder features and preceding words. The decoder is comprised of basic $LSTM$ network with a decoder LAN block, shared among all $LSTM$ nodes. With total M number of feature sets, $m = (M-1)^{th}$ layer information, $L^{\hat{m}}$ is passed to the decoding stage along with the mean projected pooled global information $\overline{G} = W_g s_t$. Here $W_g \in \mathbb{R}^{z \times q}$, projects the final POS features into z dimension. Particularly, for a timestep t, input to the decoder $LSTM$ is the embedding of word w_t, concatenated with element-wise sum of global feature $\overline{G}$ and bilinear-pooled representation Θ_{t-1} is:

$$h_t, c_t = LSTM\left([W_E w_t; \Theta_{t-1} \oplus \overline{G}], h_{t-1}, c_{t-1}\right) \quad (5)$$

Here, h_t and c_t are the hidden output and memory state from $LSTM$, W_E is the word-embedding matrix, and $\oplus$ is the summation operation. Θ_t is calculated by inserting h_t as Query Q to the LAN block with the Local Features $L^{\hat{m}}$ as $\Theta_t = W_A \left[LAN(h_t, L^{\hat{m}}); h_t\right]$

By exploiting LAN, Θ_t is the enhanced representation of $LSTM$ output, where it bridges the encoder information with the hidden states of $LSTM$ to bring the local visual information as keys and values attended by 2nd order interaction. After linear projection using W_A, Θ_t is passed using an embedding layer of vocabulary dimension, followed by a softmax function to predict the next word as:

$$w_{t+1} = p_\theta(w_{t+1} | w_{1:t}, \theta) = softmax\left(W_v \sigma\left(\Theta_t\right)\right) \quad (6)$$

The probabilistic word generation based on given parameters θ continues until a STOP token is received or generation reaches maximum caption length. Figure 1 depicts overall captioning architecture.

Entailment-based Caption Scoring: Textual entailment predicts the probability of a proposition to be true provided another proposition is true. Recent papers use CIDEr score as the reward function [35]. However, CIDEr uses undirected n-gram, assigning a high score even if there is a small-sized but critical error like negation, unrelated action, or object. According to [24], textual entailment ensures that the hypothesis gets a high reward only if it is logically inferred from the premise. Therefore we use the entailment scores as reward $r(w^s)$ in equation 8 to only calculate the loss on standalone RL training. We use RoBERTa [20] instead of the Decomposable Attention model (DA) [24] as DA assigns unreasonably high entailment scores to many non-matching sentence pairs. We present a comparative analysis of the CIDEr score and two entailment scores, DA & RoBERTa in Table 1. RoBERTa captures the wrong object and action better than DA.

Shared Learning Strategy: Given $S = (w_1, w_2, ..., w_n)$ as ground truth sequence for a clip, the goal is to learn

Ground Truth	Generated Caption	CIDEr	Entailment score	
			DA	RoBERTa
a person is playing a game	a man is playing a baseball		0.999	0.003
a young girl on stage	a boy is singing on stage		1	0.006
a man narrates a pokemon game	a person is playing a video game	0.24	0.971	0.753
a man is playing a video game	a man is talking about a river		0.999	0.012

Table 1. Comparison of CIDEr and entailment scores.

model parameters θ and minimize Cross-Entropy (XE) loss:

$$L^{XE}(\theta) = -\sum_{t=1}^{n} p(w_t | w_{1:t-1}, \theta) \quad (7)$$

However, according to recent studies [26], word-level cross-entropy training restricts sentence generation in a specific structure. For training VSLAN, we employ Self-critical Sequence Training [28] to minimize the expected reward, sentence-pair similarity score discussed in the previous section. The loss function:

$$\nabla_\theta L^{RL}(\theta) \approx -(r(w^s) - b)\nabla_\theta \log p_\theta(w^s) \quad (8)$$

Here, $r(w^s)$ is the reward for sampled sentence w^s with a baseline b for variance reduction. Standalone RL-based training does not warrant caption reliability [18], and fluency. We experiment with shared learning strategy, which employs both cross-entropy and reinforcement learning algorithm to calculate a shared loss $L^S(\theta)$:

$$L^S(\theta) = \eta L^{XE}(\theta) + (1 - \eta)L^{RL}(\theta) \quad (9)$$

Here, η is an adjustable hyperparameter. We evaluate VSLAN with L^{XE}, L^{RL}, and L^S to demonstrate the performance benchmark of the loss functions, respectively.

4. Experiments

4.1. Dataset Details and Preprocessing

We evaluate VSLAN on MS Research Video Description (MSVD/ YouTube2Text) [3] and MSR-VTT [39]. More details on these datasets and the preprocessing strategies can be found in the attached **supplementary material**.

4.2. Implementation Details

Feature Extraction: For object-level features, we utilize Faster R-CNN [27]. For a clip l_i, 30 frames are uniformly sampled and sorted based on the number of predicted boxes. The frame with the highest number of boxes is used to extract the regional features that represents the objects of that clip. We also used an off-the-shelf 2D feature descriptor, VGG16 [30], to extract the mean vector from 5 uniformly sampled frames from the clip. As the captions are mostly focused on the video activities, we leverage 3D CNNs i.e., C3D [31] and ResNext-101 [9]. Both architectures are trained on 3D kernel to capture the temporal information for each video clip. For both networks, we use pre-

Model	Backbone	B@4	M	C	R
Encoder-Decoder with Attention					
DenseLSTM	C3D+V	50.4	32.9	72.6	-
SAM	R	54	35.3	87.4	-
VRE	R	51.7	34.3	86.7	71.9
X-LAN	FR	52.1	31.1	79.9	72.2
Explicit Attribute Guidance					
GRU-EVE	IR+C3D+Y	47.9	35	78.1	71.5
HMM	C3D+IR	52.9	33.8	74.5	-
C-R	IR+R	<u>57</u>	<u>36.8</u>	<u>96.8</u>	-
Encoder-Decoder with Reconstruction					
RecNet$_{local}$	V+IR	52.3	34.1	80.3	69.8
MTL	V+G+IR	54.5	36	92.4	<u>72.8</u>
Multiple Visual Feature Aggregation					
HTM	C3D+V+R	54.7	35.2	91.3	72.5
SibNet	G	54.2	34.8	88.2	71.7
POS-CG	IR+M	53.9	34.9	91	72.1
VSLAN (ours) with Crossentropy (XE) Loss					
VSLAN$_{L^0}$	C3D	53.5	32.7	88.2	70.8
VSLAN$_{L^1}$	R	54.1	33.5	89.8	72.5
VSLAN$_{L^2}$	VGG16	48.7	31.2	84.6	68.3
VSLAN$_{L^3}$	FR	49.8	31.9	85.8	69.9
VSLAN$_{L^{all}}$ (w/o D. LAN)	R+C3D+V+FR	55.2	35.8	94.5	73.8
VSLAN$_{L^{all}}$ (w/ D. LAN)	R+C3D+V+FR	56.4	36.5	96.8	74.3
VSLAN (ours) with Reinforcement Learning (RL) Loss					
VSLAN$_{L^{all}}$ (w/o D. LAN)	R+C3D+V+FR	56.9	35.6	95.1	73.9
VSLAN$_{L^{all}}$ (w/ D. LAN)	R+C3D+V+FR	57.4	36.7	97.9	75
VSLAN (ours) with and Shared Loss (SL)					
VSLAN$_{L^{all}}$ (w/o D. LAN)	R+C3D+V+FR	57	36.1	97.3	74.2
VSLAN$_{L^{all}}$ (w/ D. LAN)	R+C3D+V+FR	**57.4**	**36.9**	**98.1**	**75.6**

Table 2. Performance comparison of VSLAN (last 10 rows) and other state-of-the-art models on MSVD, grouped by segments. Here, BLEU-4 (B@4), METEOR (M), CIDEr (C), and ROUGE$_L$ (R) are reported as percentage (%). V, G, Y, IR, M, R, FR corresponds to VGG16, GoogleNet, YOLO, Inception ResNet-v2, Optical Flow, ResNext-101, and Faster-RCNN.

trained models based on the Kinetics-400 dataset. We assume that, due to the inherent architectural difference, they produce distinguishable representations even trained on an identical dataset. The clip frames have been resized into 256×256 dimensions for network input. For each clip, outputs of the pre-$softmax$ layers are used as feature of the 3D CNN models. We initially set L^0 = ResNext-101, L^1 = C3D, L^2 = VGG16, and L^3 = Faster R-CNN to update the network parameters from coarse action representation to fine-grained object information. Moreover, we alternate the order of feature sets by holdout or shuffle and discuss result changes in the **supplementary material**.

Training Parameters: For LAN, we set the unified attention size $z = 1024$ and latent dimension $z' = 512$. For bilinear pooling in Equation 1 and 2, we use ELU as activation function, where ReLU for local feature attention at Equation 3. To squeeze information, we set $x = 256$, $y = 512$ for FAN. We transform feature dimension L^m and

$\hat{G}_m$ into similar key l, query q, value v dimension, 1024. For VaPen Encoding, the latent distribution δ_t dimension is set to 64 The decoder LSTM hidden dimension is set to 1024. We use Adam with 0.0001 learning rate. For training, we use 64 as batch size with early-stopping epochs. Both encoder and decoder parameters are uniformly distributed with gradient clip of size 10.0. Considering the learning uncertainty, before RL training, we pretrain the model with cross-entropy loss up to 10 epoch for faster convergence. Before training the decoder, we early train VaPEn with 50 epochs. This warm-up training is done to ease the complete model's training process by initializing the time-consuming latent distribution into a certain state. For shared learning, we set $\eta = 0.3$ empirically. The model is implemented on PyTorch [1] with NVIDIA Tesla V100 GPU. VSLAN (full) model consists of a total of 11.6 million parameters, and one forward pass takes around 10.5 seconds (averaged over 500 test iterations). In the **supplementary material**, we plot and analyze the convergence scenario of cross-entropy and reinforcement learning, followed by a discussion on the advantage of shared learning strategy for different η.

4.3. Experimental Setup

To compare our approach with existing works, we use four standard metrics, BLEU, METEOR, CIDEr, and ROUGE$_L$ available in MS-COCO toolkit [2]. For both datasets, the best-performing training model on the validation set is applied to the test set for comparison. During testing, we sample captions using beam search with size 5. For diversity evaluation, we sample 10 POS sequences using VaPEn. Similarly, 10 samples are extracted using DBS.

4.4. Performance Comparison

(1) **Encoder-Decoder with Attention:** DenseLSTM [42], SAM [37], X-LAN [21], and VRE [29]. (2) **Explicit Attribute Guidance:** GRU-EVE [1], HMM [38], and C-R [11]. (3) **Encoder-Decoder with Reconstruction:** RecNet$_{local}$ [36] and MTL [23]. (4) **Multiple Visual Feature Aggregation:** HTM [13], XlanV [14], SibNet [19], and POS-CG [35]. (5) **Multi Modal Feature Fusion:** M&M TGMM [4], Attention Fusion [10], V-ShaWei-GA [8], SMCG [19], and MGSA$_{+audio}$ [5].

MSVD Dataset: We outline the comparison of state-of-the-art methods with VSLAN in Table 2. The last 10 rows depict different variations of VSLAN. L^{all} stands for the complete VSLAN model utilizing 4 feature sets. At decoding, we experiment by excluding (w/o D. LAN) and including (w/ D. LAN) the LAN. The model is trained with the losses: Cross-Entropy, Reinforcement, and Shared Loss. The captions of the results are an average score of 10 VaPEn sample.

---

[1]https://pytorch.org
[2]https://github.com/tylin/coco-caption

Model	Backbone	B@4	M	C	R
Encoder-Decoder with Attention					
DenseLSTM	C3D+V	38.1	26.6	42.8	-
VRE	R	43.2	28.0	48.3	62.0
Explicit Attribute Guidance					
GRU-EVE	IR+C3D+Y	38.3	28.4	48.1	60.7
HMM	C3D+IR	39.9	28.3	40.9	-
Encoder-Decoder with Reconstruction					
RecNet$_{local}$	V+IR	39.1	26.6	42.7	59.3
MTL	V+G+IR	40.8	28.8	47.1	60.2
Multiple Visual Feature Aggregation					
SibNet	G	40.9	27.5	47.5	60.2
XlanV_{RL}	IR+I3D	41.2	28.6	54.2	61.5
POS-CG	IR+M	41.3	28.7	<u>53.4</u>	62.1
Multi Modal Feature Fusion					
M&M TGMM	IR+C3D+A	44.3	29.4	49.3	<u>62.1</u>
Attention Fusion	V+C3D+A	39.7	25.5	40	-
V-ShaWei-GA	C3D+A	**47.9**	<u>30.9</u>	-	-
MGSA$_{+audio}$	IR+C3D+A	45.4	28.6	50.1	-
VSLAN with Reinforcement Learning and Shared Loss					
VSLAN$_{L^1}$	R	39.8	26.5	43.7	58.1
VSLAN$_{L^{all}}$ (CONCAT)	R+C3D+V+FR	41.3	27.3	47.5	59.4
VSLAN$_{L^{all}}$ (w/o D. LAN)	R+C3D+V+FR	45.1	31.2	54.6	60.5
VSLAN$_{L^{all}}$ (w/ D. LAN)	R+C3D+V+FR	<u>46.5</u>	**32.8**	**55.8**	**62.4**

Table 3. Performance comparison of VSLAN (last 4 rows) and other state-of-the-art models on MSR-VTT dataset. The scores are reported as percentage (%). A denotes Audio features.

Model	MSVD			MSR-VTT		
	mBleu-4	Div-1	Div-2	mBleu-4	Div-1	Div-2
RecNet	0.86	0.22	0.23	0.84	0.2	0.21
POS-CG	0.79	0.24	0.25	0.76	0.22	0.23
X-LAN	0.83	0.22	0.25	0.8	0.24	0.28
SMCG+AllRec	0.74	0.28	0.31	0.61	0.27	0.29
VSLAN$_{no\ VaPEn}$ (w/o D. LAN)	0.82	0.21	0.23	0.81	0.23	0.26
VSLAN$_{no\ VaPEn}$ (w D. LAN)	0.77	0.26	0.29	0.75	0.26	0.29
VSLAN$_{full}$ (w/o DBS)	0.64	0.31	**0.36**	0.63	0.28	0.31
VSLAN$_{full}$ (w DBS)	**0.62**	**0.32**	0.36	**0.58**	**0.3**	**0.33**

Table 4. Comparison of related methods on diversity metrics.

The simplest version of our model, VSLAN$_{L^0}$ achieves an 88.2 CIDEr score and outperforms approaches based on straightforward attention with encoder and decoder. VSLAN$_{L^{all}}$ with XE loss reaches the highest baseline, C-R, with respect to all metrics. Here, we note that the C-R model explicitly guides the model based on external representation, i.e., visual relationship detector using language, where VSLAN solely relies on the video information as input. Similarly, GRU-EVE and HMM guide the model based on attribute information from object and action class labels, where we encourage VSLAN to learn from training captions directly. VSLAN$_{L^{all}}$ with LAN decoder, trained with shared loss, gains the best performance compared to approaches to date. Our closest competitor, POS-CG, utilizes training captions to extract POS information and then employ multiple visual features to predict POS, later used to generate the captions. We rather exploit only the visual information available and leave the pos-guided caption generation end-to-end, which gains $+7.8\%$ CIDEr score over POS-CG. Also, robustness of our VSLAN Encoder is identified by using a regular decoder on the average global feature, $\hat{G}$ without attention (w/o D. LAN), which outperforms attention and reconstruction methods by a wide margin.

MSR-VTT Dataset: This dataset is relatively complex with context and audio information. As mentioned earlier, we ignored modalities other than exploiting only visual data of MSR-VTT. For comparison, we train the best models on the MSVD dataset, mentioned in the last 4 rows of Table 3. GRU-EVE and HMM, even guide training with attributes, exploit only visual features, are outperformed by 7% using VSLAN with basic decoder (w/o D. LAN). Similarly, VSLAN with a basic decoder excels in reconstruction models, which uses a comparatively complex decoder. Similar performance is achieved against the multiple visual feature fusion models. However, recent state-of-the-art works are based on multi-modal, i.e., audio, category information with visual data. To verify our model robustness, we compare with state-of-the-art multi-modal models. We notice that our basic decoder ranks almost similar to the closest competitor, V-ShaWei-GA. However, the full model (w/ D. LAN) slightly improves the basic decoder. Our best performing metric is CIDEr, which outperforms POS-CG and MGSA$_{+audio}$ by 4.5% and 11.4%, respectively. We argue that rich audio information of the MSR-VTT leveraged better BLEU on V-ShaWei-GA. It is worth noting that our result does not necessarily require any sophisticated pretrained CNN models because of our inherent architecture's feature distillation process. We note that VSLAN trained with R has also achieved competitive performance with a concatenated feature only (CONCAT, w/o. FAN). Thus the steady improvement can be directly linked to our VSLAN architecture, which results in fine-grained representation. Extended VSLAN with category and audio info is expected to gain better results, which is out of our research scope.

Diversity Analysis: In Table 2 and 3, we see VSLAN's overall performance. However, these evaluations are limited to the efficacy of LAN and FAN block. Because, for these evaluations, we sample top-1 caption. So, the full potential of VaPEn is ignored. To this end, we evaluate the diversity of VSLAN's captions in Table 4. We use two protocols: mBleu-4 (lower is better), Div-n (higher is better). m-Bleu-4 performs a cross-match of 4-gram occurrences on the generated captions from one clip. These metrics ensure that the sentences are significantly different from each other. Div-n indicates a unique n-gram ratio between the generated captions. For more details regarding the metrics, we refer readers to [2]. We compare four versions of our model with recent algorithms, RecNet, POS-CG, and SMCG. Additionally, we use the pre-trained model of XlanV_{RL} on our

Figure 4. A detailed visual illustration of the diverse captions generated by our architecture. Each row of images represent sequential frames of a random test video. The heatmap depicts spatio-temporal attention distribution while generating each sentence (showed below the images). The color codes of each caption represents POS by VaPEn, in the bottom. For each, four reference ground truth are presented.

datasets to compare. VSLAN_{no VaPEn} is VSLAN without VaPEn, where we map $\tilde{G}$ to $\bar{G}$ using a linear layer. We can analyze that VSLAN_{no VaPEn} with decoder LAN achieves marginal performance compared to others. So, adding a decoder does not guarantee better diversity. However, when the VaPEn is included, the performance jumps and exceeds the compared methods. This phenomenon is due to the probabilistic property of the VaPEn encoder, which encourages the decoder to generate diverse captions. Moreover, we can see that DBS does not have a larger effect compared to VaPEn. Finally, we can see that the captions are syntactically correct (based on CIDEr scores) and diversified (based on the diversity evaluation).
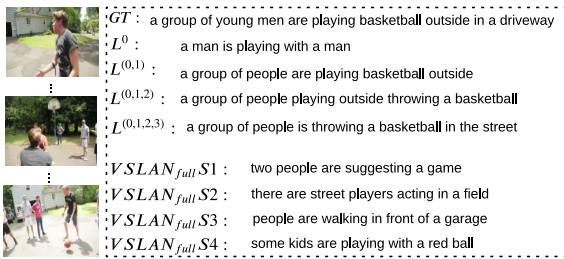

Figure 5. Qualitative evaluation on a random test video. GT is the ground truth, L^m are the indices of used feature sets.

Qualitative Analysis: Figure 5 shows a ground truth GT sentence and predicted captions for a random test video. The respective feature sets, L^m, are propagated through stacking the layers. We notice that L^0 misses basic object properties, where L^1 with new block corrects sentence structure. With $L^{(0,1,2)}$, additional subject and action features are attended, which results in a new verb, "throwing." Finally, the "street" attribute is captured by $L^{(0,1,2,3)}$, where Faster R-CNN was introduced. The result evolution depicts

the robustness of feature interaction while passing novel set using FAN block. Some samples from VSLAN are shown in the last 4 rows. Figure 4 organizes our complete model performance into one page. For each row of frames (belong to a single video), it can be seen that each sample of VaPEn influences the caption decoder to attend different spatio-temporal regions behind generating each diverse caption. For example, in row 2, "a man is standing" and "a boy is singing a song" belong to the nearby frame, whereas two attention locations influenced different meaningful yet relevant captions. Similarly, in the last row, we can see a failure case, where our method could not attend to appropriate locations due to lack of training. For each row sequence, we showed four ground truth captions as a reference. We have further evaluated the consistency of VSLAN based on two ablation studies. First, if the performance remains consistent upon shuffle of the feature orders. Second, if there is any performance gain if we use the existing model trained on video captioning dataset, followed by a cross-dataset experiment. We refer readers to the **supplementary material**.

5. Conclusion

This paper proposes VSLAN for generating diverse captions by exploiting explicit feature interaction and aggregating multiple feature streams in a discount fashion, followed by diverse POS prediction. Previous captioning models aggregated features from multiple extractors by either weighted concatenation or passing through linear layers to produce new representations. Also, they ignored diversity in caption generation. Whereas, VSLAN's learned attributes overcome the previous approaches' limitation on generating diverse captions by attending temporal states end-to-end, which achieved state-of-the-art performance.

References

[1] 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 *IEEE CVPR*, 2019.

[2] Jyoti Aneja, Harsh Agrawal, Dhruv Batra, and Alexander Schwing. Sequential latent spaces for modeling the intention during diverse image captioning. In *ICCV*, pages 4261–4270, 2019.

[3] David Chen and William B Dolan. Collecting highly parallel data for paraphrase evaluation. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 190–200, 2011.

[4] Shizhe Chen, Jia Chen, Qin Jin, and Alexander Hauptmann. Video captioning with guidance of multimodal latent topics. In *Proceedings of the 25th ACM international conference on Multimedia*, pages 1838–1846, 2017.

[5] Shaoxiang Chen and Yu-Gang Jiang. Motion guided spatial attention for video captioning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 8191–8198, 2019.

[6] Junyoung Chung, Kyle Kastner, Laurent Dinh, Kratarth Goel, Aaron C Courville, and Yoshua Bengio. A recurrent latent variable model for sequential data. *Advances in neural information processing systems*, 28:2980–2988, 2015.

[7] Yang Gao, Oscar Beijbom, Ning Zhang, and Trevor Darrell. Compact bilinear pooling. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 317–326, 2016.

[8] Wang-Li Hao, Zhaoxiang Zhang, and He Guan. Integrating both visual and audio cues for enhanced video caption. In *AAAI*, 2018.

[9] Kensho Hara, Hirokatsu Kataoka, and Yutaka Satoh. Can spatiotemporal 3d cnns retrace the history of 2d cnns and imagenet? In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 6546–6555, 2018.

[10] 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 *Proceedings of the IEEE international conference on computer vision*, pages 4193–4202, 2017.

[11] Jingyi Hou, Xinxiao Wu, Xiaoxun Zhang, Yayun Qi, Yunde Jia, and Jiebo Luo. Joint commonsense and relation reasoning for image and video captioning. In *AAAI*, pages 10973–10980, 2020.

[12] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7132–7141, 2018.

[13] Yaosi Hu, Zhenzhong Chen, Zheng-Jun Zha, and Feng Wu. Hierarchical global-local temporal modeling for video captioning. In *Proceedings of the 27th ACM International Conference on Multimedia*, pages 774–783, 2019.

[14] Yiqing Huang, Qiuyu Cai, Siyu Xu, and Jiansheng Chen. Xlanv model with adaptively multi-modality feature fusing for video captioning. In *Proceedings of the 28th ACM International Conference on Multimedia*, pages 4600–4604, 2020.

[15] Jin-Hwa Kim, Jaehyun Jun, and Byoung-Tak Zhang. Bilinear attention networks. In *Advances in Neural Information Processing Systems*, pages 1564–1574, 2018.

[16] Jin-Hwa Kim, Kyoung Woon On, Woosang Lim, Jeonghee Kim, Jung-Woo Ha, and Byoung-Tak Zhang. Hadamard Product for Low-rank Bilinear Pooling. In *The 5th International Conference on Learning Representations*, 2017.

[17] Shu Kong and Charless Fowlkes. Low-rank bilinear pooling for fine-grained classification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 365–374, 2017.

[18] Chia-Wei Liu, Ryan Lowe, Iulian Serban, Mike Noseworthy, Laurent Charlin, and Joelle Pineau. How NOT to evaluate your dialogue system: An empirical study of unsupervised evaluation metrics for dialogue response generation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2122–2132, Austin, Texas, Nov. 2016. Association for Computational Linguistics.

[19] Sheng Liu, Zhou Ren, and Junsong Yuan. Sibnet: Sibling convolutional encoder for video captioning. MM '18, page 1425–1434, New York, NY, USA, 2018. Association for Computing Machinery.

[20] Y. Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, M. Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. *ArXiv*, abs/1907.11692, 2019.

[21] Yingwei Pan, Ting Yao, Yehao Li, and Tao Mei. X-linear attention networks for image captioning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10971–10980, 2020.

[22] 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 of the Association for Computational Linguistics*, pages 311–318, 2002.

[23] Ramakanth Pasunuru and Mohit Bansal. Multi-task video captioning with video and entailment generation. In *ACL*, 2017.

[24] Ramakanth Pasunuru and Mohit Bansal. Reinforced video captioning with entailment rewards. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 979–985, Copenhagen, Denmark, Sept. 2017. Association for Computational Linguistics.

[25] Yunchen Pu, Martin Renqiang Min, Zhe Gan, and Lawrence Carin. Adaptive feature abstraction for translating video to text. In *AAAI*, pages 7284–7291, 2018.

[26] Marc'Aurelio Ranzato, S. Chopra, M. Auli, and W. Zaremba. Sequence level training with recurrent neural networks. *CoRR*, abs/1511.06732, 2016.

[27] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pages 91–99, 2015.

[28] Steven J Rennie, Etienne Marcheret, Youssef Mroueh, Jerret Ross, and Vaibhava Goel. Self-critical sequence training for

image captioning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7008–7024, 2017.

[29] Xiangxi Shi, Jianfei Cai, Shafiq Joty, and Jiuxiang Gu. Watch it twice: Video captioning with a refocused video encoder. In *Proceedings of the 27th ACM International Conference on Multimedia*, pages 818–826, 2019.

[30] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[31] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. Learning spatiotemporal features with 3d convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, pages 4489–4497, 2015.

[32] Subhashini Venugopalan, Marcus Rohrbach, Jeffrey Donahue, Raymond Mooney, Trevor Darrell, and Kate Saenko. Sequence to sequence-video to text. In *Proceedings of the IEEE international conference on computer vision*, pages 4534–4542, 2015.

[33] Subhashini Venugopalan, Huijuan Xu, J. Donahue, Marcus Rohrbach, R. Mooney, and Kate Saenko. Translating videos to natural language using deep recurrent neural networks. In *HLT-NAACL*, 2015.

[34] Ashwin K Vijayakumar, Michael Cogswell, Ramprasath R Selvaraju, Qing Sun, Stefan Lee, David Crandall, and Dhruv Batra. Diverse beam search: Decoding diverse solutions from neural sequence models. *arXiv preprint arXiv:1610.02424*, 2016.

[35] 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 *Proceedings of the IEEE International Conference on Computer Vision*, pages 2641–2650, 2019.

[36] Bairui Wang, Lin Ma, Wei Zhang, and Wei Liu. Reconstruction network for video captioning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7622–7631, 2018.

[37] Huiyun Wang, Youjiang Xu, and Yahong Han. Spotting and aggregating salient regions for video captioning. In *Proceedings of the 26th ACM international conference on Multimedia*, pages 1519–1526, 2018.

[38] Junbo Wang, Wei Wang, Yan Huang, Liang Wang, and Tieniu Tan. Hierarchical memory modelling for video captioning. In *Proceedings of the 26th ACM international conference on Multimedia*, pages 63–71, 2018.

[39] Jun Xu, Tao Mei, Ting Yao, and Yong Rui. Msr-vtt: A large video description dataset for bridging video and language. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5288–5296, 2016.

[40] Li Yao, Atousa Torabi, Kyunghyun Cho, Nicolas Ballas, Christopher Pal, Hugo Larochelle, and Aaron Courville. Describing videos by exploiting temporal structure. In *Proceedings of the IEEE international conference on computer vision*, pages 4507–4515, 2015.

[41] Yitian Yuan, Lin Ma, Jingwen Wang, and Wenwu Zhu. Controllable video captioning with an exemplar sentence. In *Proceedings of the 28th ACM International Conference on Multimedia*, pages 1085–1093, 2020.

[42] Yongqing Zhu and Shuqiang Jiang. Attention-based densely connected lstm for video captioning. In *Proceedings of the 27th ACM international conference on multimedia*, pages 802–810, 2019.