Sparse Frame Grouping Network with Action Centered for Untrimmed Video Paragraph Captioning

Guorui Yu[1], Yimin Hu[1], Yuejie Zhang[1], Rui Feng[1], Tao Zhang[2], Shang Gao[3]
[1]Sch. of Comp. Sci, Shanghai Key Laboratory of Intelligent Information Processing, Fudan University
[2]Sch. of Inf. Manag. & Eng, Shanghai Key Laboratory of Financial Information Technology, Shanghai University of Finance and Economics, [3]Sch. of Inf. Tech, Deakin University
21210240409@m.fudan.edu.cn, {20210240283, yjzhang, fengrui}@fudan.edu.cn, taozhang@mail.shufe.edu.cn, shang.gao@deakin.edu.au

Abstract

Generating paragraph captions for untrimmed videos without event annotations is challenging, especially when aiming to enhance precision and minimize repetition at the same time. To address this challenge, we propose a module called Sparse Frame Grouping (SFG). It dynamically groups event information with the help of action information for the entire video and excludes redundant frames within pre-defined clips. To enhance the performance, an Intra-Contrastive Learning technique is designed to align the SFG module with the core event content in the paragraph, and an Inter-Contrastive Learning method is employed to learn action-guided context with reduced static noise simultaneously. Extensive experiments are conducted on two benchmark datasets (ActivityNet Captions and YouCook2). Results demonstrate that SFG outperforms the state-of-the-art methods on all metrics.

1 Introduction

Video Paragraph Captioning for untrimmed video aims to describe multiple events in the video with three or four sentences. Initially, the focus was on describing each event with a single sentence in a long video (J. Lei and Mohit, 2020) (J. Mun and Han, 2019), until (Y. Song and Jin, 2021) proposed to directly generate paragraph captions without event ground truths. This refined approach to paragraph captioning is more practical but also more challenging. Firstly, without event labels, it becomes difficult to precisely locate the transitions between different events and to summarize the complete story from a holistic perspective. Secondly, most prior studies generated one sentence for each event, whereas paragraph captioning without event ground truths requires generating sentences that are two or three times longer. This naturally exacerbates issues of redundancy and repetition. Our work focuses on video paragraph captioning without event annotations as shown in Fig. 1.

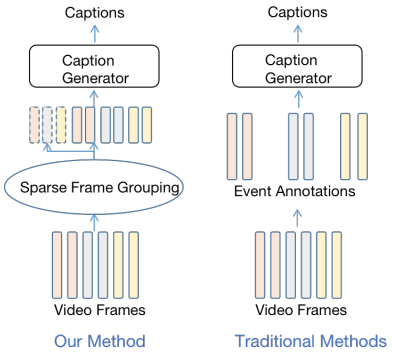

Figure 1: Comparison of SFG with traditional methods.

As a common frame sampling strategy in most existing works, random sampling method randomly selects one frame every one or two seconds. The hypothesis of previous works is that the frames in one second are highly similar, so random sampling does not result in significant information loss. However, actions often occur rapidly, such as within a one-second time frame, and usually lead changes to events. Random sampling may miss crucial action frames that occur quickly and retain many similar frames as noise. According to (J. Lei and Mohit, 2022), one or two key frames are sufficient to capture important content, and the "redundant" frames can be neglected in an action clip. In this way, we can learn the visual content from a small number of frames that contain critical information.

Therefore, our objective is to learn event distributions guided by action frames, and utilize them as estimations for event locations, enabling us to construct comprehensive paragraph stories by incorporating these momentary yet important actions. Considering the impracticality of including all frames (often averaging hundreds) into the analysis, it becomes essential to find an effective way for frame selection and information grouping. Our work specifically focuses on action-driven video paragraph captioning. It tackles the captioning task by utilizing sparse frames and refined video an-

chors with action visual representation. No event annotation or any other additional information is used except for the video and paragraph captions, thereby ensuring the practicality of our method.

We propose our vanilla model based on the Transformer architect (J. Lei and Mohit, 2020). A new grouping module, named Sparse Frame Grouping (SFG), is introduced. It uses a set of designed anchors as queries to select and group useful frames. The objective is to extract two types of content from the video: event distributions and action changes between adjacent frames. The idea is to progressively estimate the actual event distributions by grouping visually similar video frames using these designed anchors.

At each encoder layer, SFG is first applied to every two frames and selects one informative frame, where anchors are random frames. As a result, a sequence of sparse and important video frames are obtained, instead of using the randomly sampled ones. This process is referred to as clip-level SFG. Next, SFG is employed to group event context across the entire video. Video anchors are initialized using triplets of ground truths during the training phase. The objective is to ensure that the anchors capture the action-driven event distributions in the video, while the triplets provide a concise representation of the main action content in the sentences. This process is referred to as video-level SFG. At this stage, the video anchors are updated by grouping the newly selected frames. The remaining steps follow the standard transformer computation. In the decoder, we modify the cross-attention mechanism by attending video anchors in the video-level SFG additionally. The weighted anchors, along with the weighted video frames, guide the decoder to generate each word at each decoding step.

To train SFG without using event annotations, we introduce an Intra- and Inter-Contrastive Learning technique. The objective is to ensure that the video anchors are close to their relevant event contents in the video. This process naturally involves contrastive learning between anchors and paragraph. The challenge is that the embedding of the entire paragraph contains excessive noise for the video anchors which are intended to represent the center of events. What we aim to learn is a summary of the event context within the paragraph, while disregarding those static descriptions that are not relevant to the actions in our video anchors. We modify the positive contrastive pair from visual-paragraph to visual-triplets, while the negative pairs consist of unmatched pairs within the batch. For each sentence in the paragraph, triplet (s, v, o) (Sec. 3.2) is extracted and the embeddings of all triplets are averaged. This contrastive learning approach is referred to as Intra-Contrastive Learning, and its purpose is to minimize the distance between the overall video anchors and the paragraphs.

Besides, it is crucial to differentiate the video anchors from other visual contents in the video. The Inter-Contrastive Learning method takes the left video representation as a negative visual part. It requires the distance between the video anchors and the triplet embeddings is at least smaller than the distance from the negative visual part to the triplet embeddings. We conduct experiments on ActivityNet Caption and YouCook2 datasets. The results demonstrate that our model outperforms the state-of-the-art (SOTA) methods when evaluated using the event ground truths on both datasets.

The contributions of this work are as follows:

(1) We propose a Sparse Frame Grouping (SFG) module to estimate event locations and catch action-driven visual content. It operates at both the short clips and the whole video levels without relying on event annotations, thereby enhancing the practicality of video paragraph captioning.

(2) We build an Intra- and Intro-contrastive loss for SFG training. It helps the model focus more on learning event distribution rather than other noisy contextual information, reducing redundancy and improving precision of the captions.

(3) Our model achieves SOTA results on benchmark datasets ActivityNet and YouCook2, outperforming even the methods using event annotations.

2 Related work

Dense Video Captioning As a multi-modal task, it has attracted much attention. Popular categorization of dense video captioning falls into three: predicting event location and sentences, solely predicting sentences with event annotations, and predicting paragraphs without event annotations. In this paper, our focus falls into the third category. (L. Zhou and Xiong, 2018) is the first adopting transformer to decode sentences and create differentiable masks for the consistency between proposals and captions. (J. Mun and Han, 2019) used a pointer network to select possible event proposals from candidates and to recurrently generate sen-

tence for each event. (T. Wang and Luo, 2021) firstly tackled this task as a parallel multitask. Localization and caption generation are predicted at the same time using a DETR scheme. (P. Jae Sung and Anna, 2019) adopted adversarial inference to evaluate the paragraph quality. (J. Lei and Mohit, 2020) proposed a memory module in decoder to remove redundant visual and text information dynamically. These prior studies all rely on event annotations. (Y. Song and Jin, 2021) focused on reducing redundancy using an adding and erasing module at each decoding step without event annotations. However, little attention have been paid to video redundancy itself at the encoder stage. In our work, we aim to address this issue by tackling it from the initial frame selection stage and capturing the core action content of the video.

Video Action Understanding Action localization in video is an important task for video understanding. (Yeung et al., 2016) and (Zhou et al., 2018) utilized reinforce learning to dynamically determine the next frame. (Z. Yuan and Gangshan, 2021) proposed to sample motion from cumulative distribution. (B. Korbar and Torresani, 2019) proposed to understand video only through some salient clips which were ranked and selected for further understanding. (Gowda et al., 2021) selected single frame and utilized high level feature for video classification. The previous works seldom combine the two modalities (visual and text) together. Our work utilizes both to better select key frames.

3 Methods

The proposed Sparse Frame Grouping Network is shown in Fig. 2. It consists of three modules: clip-level SFG, video-level SFG, and Intra- and Inter-Contrastive Learning. Together with the vanilla transformer, these modules generate action-driven and event annotation-free paragraphs. In this section, the vanilla transformer is first described, followed by the introduction to the three modules.

3.1 Vanilla Model

Given an untrimmed video $V = \{v_1, ..., v_n\}, v_i \in R^{d_v}$, where v_i is the i-th video frame representation (referred to as frame hereafter), n is the total number of frames in V, and d_v is the dimension of video frame. The video paragraph captioning task aims to describe the content of V with a paragraph $Y = \{y_1, ..., y_k\}, y_j \in R^{d_w}$, where y_j is the j-th word in the k length paragraph, and d_w is the dimension of word embeddings. Here we build our vanilla model based on the transformer structure, following the approach described in (L. Zhou and Xiong, 2018) and (J. Lei and Mohit, 2020). The encoder transformer E is used to learn the video representation V_e from video input V and the decoder transformer D generates the captions conditioned on V_e. Specifically, we first adopt N-layer transformer encoder to model the long range interactions:

$$V_{i+1} = FFN(V_i + MultiHead(V_i, V_i, V_i)) \quad (1)$$

where V_i is the output of i-th transformer layer. $FFN()$ and $MultiHead()$ denote feed-forward network and multi-head attention as in (L. Zhou and Xiong, 2018). For the decoder, we first use the masked multi-head self attention (L. Zhou and Xiong, 2018) to model the hidden states of the last layer, then we introduce cross attention to fuse the text modality and the visual modality. For cross attention, we use the text embeddings at each decoding step as the query, and V_n from the last layer of the encoder as key and value, so that each word can be generated based on the previously predicted words and the attended video contents. The vanilla model is trained typically by the Maximum Likelihood Estimation (MLE) loss:

$$L_{mle} = -\frac{1}{T}\sum_{t=0}^{T} \log p(y_t^* | y_{<t}^*, V_e) \quad (2)$$

$$L_{mle} = -\frac{1}{T}\sum_{t=0}^{T} \log p(y_t^* | y_{<t}^*, V_e) \quad (3)$$

where V_e is conditioned video representation. y_t^* is the predicted word at the t-th step and T is the length of caption. Without event segment annotations, there are three challenges in video paragraph captioning. Firstly, directly decoding the entire video usually leads to confusion in recognizing clear contexts and decoding each into word output. Secondly, the random frame sampling strategy results in the loss of essential key information, affecting the decoder's ability to catch action changes and decode precise words. Lastly, the MLE loss tends to generate high frequency and repetitive words in previous steps.

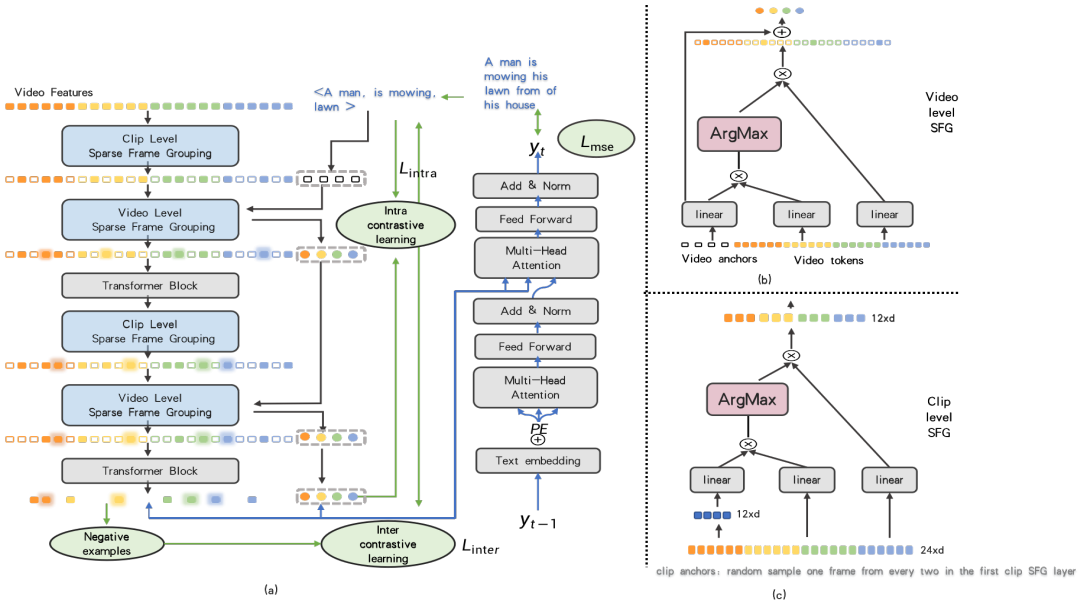

Figure 2: Overview of SFG Network. (a) Architect of the network. SFG is applied at both the clip and video level in every encoder layer. The decoder dynamically generates words based on previous video anchors and selected video frames. Intra- and Inter-Contrastive Learning is introduced for video anchor training. (b) Detail of SFG.

3.2 Progressive Grouping Transformer

As shown in Fig. 2(a), based on the vanilla transformer (Sec.3.1), instead of directly feeding all the video frames into the transformer encoder, we incorporate the clip-level sparse frame grouping (SFG) module and video-level grouping module prior to each transformer layer, denoted as SFG_{clip} and SFG_{video}. We employ dynamic tokens, known as anchors, to selectively choose useful frames from the input or the previous transformer layer. The objective of these modules is to obtain a set of sparse video frames and a collection of video anchors that can learn event distributions dynamically.

Formally, we take the l-th transformer layer as an example and have the video frames V^l as the input. Firstly, we randomly select one frame from every two frames as anchors for SFG_{clip} as $V^{l_1} = SFG_{clip}(V^l, A^l_{clip})$, where A^l_{clip} represents anchors at the clip level. V^{l_1} is the half length of the input frames. To help SFG learn the action-driven event context, we initialize the video level frames via triplet embeddings. Assuming that triplets highly summarize the main contents of the sentence in captions, we extract one triplet (L. Wang and Svetlana, 2016) from each sentence and project it to the same latent space as the video frames through linear projection. E.g., one triplet from a sentence is $< s, v, o >$, which can be embedded as $\{y_s, y_{v1}, ..., y_{vk}, y_o\}$. y_s is subject, $y_{v1}, ..., y_{vk}$ is one verb or verb phrase, and y_o is object. We sum the word embeddings in the triplet and project them to the video representation space: $a^l_i = Linear(\sum y_{triplet})$, $a^l_i \in R^{d_v}$ is one anchor among the video anchors. By initializing the left video anchors and setting the number of anchors to be equal to the number of sentences in the captions, the anchors can effectively group the content with action-related visual features and learn the event distributions through different anchors. We get the updated video anchors A^{l+1}_{video} and the selected video frames via:

$$I^l, A^{l+1}_{video} = SFG_{video}(V^{l_1}, A^l_{video}) \quad (4)$$

$$V^{l+1} = Transfromer\_Layer(V^{l_1}), \quad (5)$$

where I^l is the outcome of $ArgMax$, which is the index of center frames at this stage. We adopt the video frames selected by SFG_{clip} and forward them to l-th transformer layer. With the stacked encoder transformer layers and SFG modules, we get a set of sparse video frames with fewer redundant frames and the most representative anchors. These anchors already group the core content of the events in the video.

3.3 Sparse Frame Grouping Block

As shown in Fig. 2(b), SFG utilizes the dynamic anchors to select the center frames for each anchor

and group action-driven event content around the updated anchors.

Formally, we take anchors $A = \{a_1, ..., a_m\}, a_i \in R^{d_v}$ as the query, and the video frames V as the key and value pairs to compute the weights of video frames. To select target frames, we change the original SoftMax to $ArgMax$ and weight matrix to one-hot matrix. Then we select the center frames and obtain the updated anchors $\hat{A}$ through the $ArgMax$ function:

$$\hat{A} = A + ArgMax(A \times V) \times V, \quad (6)$$

where A represents the anchors discussed in Sec. 3.2. However, the one-hot matrix from $ArgMax$ is not differentiable. We further use Gumbel SoftMax (E. Jang and Ben, 2016) and transform Eq. 6 as follows:

$$\hat{A} = A + \frac{exp(A \times V + \beta)}{\sum exp(A \times V + \beta)} \times V, \quad (7)$$

where $\hat{A}$ represents the updated anchors. β is sampled from the $Gumble(0, 1)$ distribution to ensure that SFG can merge the video event information progressively during the training phase.

3.4 Intra- and Inter-Video Contrastive Learning

We obtain a sparse video representation V_e with significantly fewer frames compared to the raw video and a set of event context anchors after encoding. To effectively train the proposed video anchors, especially in the absence of event annotations, we introduce an Intra- and Inter-Video Contrastive Learning technique. This technique enables the learning of event context, which is mainly action context-driven. Specially, we first employ intra-video contrastive learning to minimize the distance between the representation of dynamic anchors A and the text representation. However, the ground truth information may contain noise. For example, grouping frames should not include an excessive number of static or descriptive words such as color, position and background. We aim to extract some confined context from the captions that includes action description while excluding the "unimportant" words. Therefore, triplets are used (Sec.3.2). Assuming that short triplets highly summarize the main contents of the video, we denote the positive pair as $< \overline{A}, \overline{T} >$, where $\overline{A}$ is the average of anchor representation and $\overline{T}$ is the average of triplet embeddings. The unmatched pairs in the batch are negative pairs. The Intra-contrastive loss $L_{intra} = L_{a \to t} + L_{t \to a}$ is as follows:

$$L_{a \to t} = \frac{1}{T_b} \sum_{i=1}^{T_b} \log \frac{exp(\overline{A_i} \cdot \overline{T_i})/\tau}{\sum_{j=1}^{T_b} exp(\overline{A_i} \cdot \overline{T_j})/\tau}, \quad (8)$$

$$L_{t \to a} = \frac{1}{T_b} \sum_{i=1}^{T_b} \log \frac{exp(\overline{T_i} \cdot \overline{A_i})/\tau}{\sum_{j=1}^{T_b} exp(\overline{T_i} \cdot \overline{A_j})/\tau}, \quad (9)$$

where T_b is the batch size. We use dot production to measure the similarity between A and T. The loss aims to ensure that all the anchors capture similar contextual information as the possible events described in the captions. However, to reduce redundancy in the generated captions, we also need to secure that the video anchors learn less static noise. Therefore, we further apply Inter-Contrastive Learning to guarantee that the video anchors are at least closer to the triplet embeddings compared to the surrounding visual features. For negative pairs, we select the frames that are not chosen by SFG_{video} in the last transformer encoder layer, which is $A_n = (1 - I^L) \times V$. Thus the negative visual pair becomes $< \overline{A_n}, \overline{T} >$. The positive pair in the Inter-Constrastive Learning is the same as that in the Intra-Contrastive Learning. The Inter-contrastive loss is as below:

$$L_{inter} = max(\overline{A^p} \cdot \overline{T} - \overline{A^n} \cdot \overline{T} + \sigma, 0), \quad (10)$$

where σ is a margin constant. This inter-video loss helps distinguish the representation of video anchors from the other parts of the video. The total training loss is:

$$Loss = L_{mle} + \lambda_1 L_{intra} + \lambda_2 L_{inter}, \quad (11)$$

where λ_1 and λ_2 are loss weights over zero.

4 Experiments

4.1 Datasets and Implementation

Datasets and Evaluation Metrics: All the experiments are conducted on two widely used benchmark datasets, i.e., ActivityNet Captions(K. Ranjay and Juan, 2017) and YouCook2(L. Zhou and J, 2018). For fair comparison, we use four mainstream evaluation metrics, including BLEU-4 (B@4), METEOR, CIDER and R@4. A higher score for any metric indicates better performance in captioning. Besides, we evaluate repetition using R@4, following the previous work in (P. Jae Sung and Anna, 2019). A lower R@4 score indicates

better performance, suggesting less repetition in the paragraph caption.

Implementation Details: For visual features, we utilize the video features extracted by ResNet-200 and BNInception, as provided by (L. Zhou and Xiong, 2018). The length of video anchors is set to 10 for ActivityNet Captions and 14 for YouCook2. In the inference phase, random frames are used to initialize the video anchors for SFG_{video}. For text features, words are initialized using Glove embeddings. Text is limited to 60 words. Empirically, the loss weight λ_1 is set to 1 and λ_2 to 0.5. The batch size is 32. Greedy decoding is adopted which is comparable to beam search.

4.2 Comparison with SOTA Methods

In Tab. 1, we compare our model with SOTA methods on ActivityNet Captions. These methods can be divided into two categories: (1) inferring without event annotations, including TDPC (CVPR21) (Y. Song and Jin, 2021); (2) with annotations, including VTrans (CVPR18) (L. Zhou and Xiong, 2018), MFT (ECCV18) (Y. Xiong and Lin, 2018), GVD (CVPR19) (L. Zhou and Marcus, 2019), GVDsup (CVPR19) (L. Zhou and Marcus, 2019), AdvInf (CVPR19) (P. Jae Sung and Anna, 2019), Transformer-XL (ACL19) (Z. Dai and Ruslan, 2019), MART (ACL20) (J. Lei and Mohit, 2020), PDVC (ICCV21) (T. Wang and Luo, 2021). It is worth noting that GVD, GVDsup and AdvInf adopt additional object features to align the descriptions and objects in the video. Naturally stronger feature leads to impressive performance. However, our model outperformas them, regardless of whether the object features are integrated or event annotations are used.

Without the event ground truths, our model still achieves the best performance on B@4, Meteor and CIDER with a relative improvement of 3.7%, 6.5% and 5.9% respectively compared to PDVC (on ActivityNet ae-val). The uplift is achieved even with fewer frames and no event annotations. While TDPC also utilizes key frames, it relies on the pretrained model to reconstruct the common representation space between text and video. It should be the strong prior knowledge in our model that makes the generation more precise. Additionally, TDPC tends to generate shorter sentences, which may result in lower METEOR score for AdvInf. This could be attributed to TDPC encoding all frames directly and overlooking event distributions in the video. TDPC may learn the fragmented information scattered throughout the frames. In contrast, our model makes more efforts to address this problem, resulting in more action-driven and organized descriptions. The uplift in METEOR score serves as evidence of a preference for longer sentences in our approach.

Tab. 2 shows the performance on YouCook2 validation set. Our method achieves the SOTA performance, demonstrating a significant improvement over the other methods across all the metrics. Also, YouCook2 contains a large number of events, which highlights the importance of event distributions for caption generation.

4.3 Ablation Studies

To verify the effectiveness of the proposed grouping module and loss, we conduct ablation studies on ActivityNet.

Effects of SFG As shown in Tab. 3, $Vanilla$ is the baseline with MLE loss. $Vanilla + SFG_{clip}$ refers to only SFG_{clip} is used in every layer of the encoder transformer. $Vanilla + SFG_{video} + L_{intra} + L_{inter}$ is the complete module for our video-level SFG. The last row is the full model. The experiments report that: (1) SFG_{clip} contributes to a reduction in repetition, as indicated by the improvement in R@4. R@4 is improved by 38% and by 59% in rows 1 & 2 and rows 5 & 6. This suggests that the dynamic frame sampling performed by SFG_{clip} not only benefits the selection of informative frame, but also results in a sparse video representation, reducing the redundancy from the video side; (2) for SFG_{video} in row 5, there are considerable improvements in the precision and repetition. This indicates the video anchors indeed learn to estimate the video content, reducing the static noise and, consequently the redundancy.

Effects of Intra- and Inter-Contrastive Loss In Tab. 3, $Vanilla + SFG_{video} + L_{intra}$ refers to only SFG_{video} is used in every encoder layer with $Loss_{intra}$. $Vanilla + SFG_{video} + L_{inter}$ is to verify whether the distinction of anchors alone is sufficient to summarize the event distribution in the video, without the supervision of triplets. From the results, it can be concluded that $Loss_{intra}$ has a greater impact on improving the model precision (B@4, METEOR and R@4), while $Loss_{inter}$ significantly reduces R@4. Specifically, when comparing with row 2, row 3 ($Loss_{intra} + SFG_{video}$) has improvements in B@4, METEOR and CIDER

Table 1: Performance Comparison on ActivityNet. *ae-val*. V, F, O: visual, flow and obj features. *: utilizing stronger features. The italic number means they are significant in hypothesis testing,

Model	Event Annotation		Features	*ae-val*				*ae-test*			
	Train	Infer		B@4↑	METEOR↑	CIDER↑	R@4↓	B@4↑	METEOR↑	CIDER↑	R@4↓
HSE(B. Zhang and Sha, 2018)	Y	Y	V	9.84	13.78	18.78	13.22	-	-	-	-
VTrans(L. Zhou and Xiong, 2018)	Y	Y	V+F	9.75	15.64	22.16	7.79	9.31	15.54	21.33	7.45
Tansformer-XL(Z. Dai and Ruslan, 2019)	Y	Y	V+F	10.39	15.09	21.67	8.54	10.25	14.91	21.71	8.79
MART(J. Lei and Mohit, 2020)	Y	Y	V+F	10.33	15.68	23.42	5.18	9.78	15.57	22.16	5.44
PDVC(T. Wang and Luo, 2021)	Y	Y	V+F	11.80	15.93	27.27	-	-	-	-	-
GVD(L. Zhou and Marcus, 2019)	Y	Y	V+F+O	11.04	15.71	21.95	8.76	10.50	15.60	21.60	-
GVDsup(L. Zhou and Marcus, 2019)	Y	Y	V+F+O	11.30	16.41	22.94	7.04	10.70	16.10	22.20	-
AdvInf(P. Jae Sung and Anna, 2019)	Y	Y	V+F+O	10.04	16.60	20.97	5.76	-	-	-	-
MFT(Y. Xiong and Lin, 2018)	Y	N	V+F	8.45	14.75	14.15	17.59	-	-	-	-
PDVC(T. Wang and Luo, 2021)	Y	N	V+F	10.24	15.80	20.45	-	-	-	-	-
TDPC*(Y. Song and Jin, 2021)	N	N	V+F	-	-	-	-	12.20	16.10	27.36	2.63
Vanilla	N	N	V+F	11.23	15.34	25.23	7.38	10.87	15.21	25.12	8.23
Ours	N	N	V+F	**12.24**	***16.97***	***28.89***	***2.51***	**12.21**	**16.62**	**28.54**	**2.53**

Table 2: Performance Comparison on YouCook2.

Model	B@4↑	METEOR↑	CIDER↑	R@4↓
VTrans(Z. Dai and Ruslan, 2019)	7.62	15.65	32.26	7.83
Tansformer-XL(Z. Dai and Ruslan, 2019)	6.56	14.76	26.35	6.30
MART(J. Lei and Mohit, 2020)	8.00	15.90	35.74	4.39
Vanilla	7.54	15.32	32.46	5.89
Ours	**8.53**	**16.24**	**39.27**	**4.12**

Table 3: Ablation Studies.

Index	Model	B@4	METEOR	CIDER	R@4
1	$Vanilla$	11.23	15.34	25.23	7.38
2	$Vanilla$+SFG_{clip}	11.39	15.49	25.90	5.34
3	$Vanilla$+SFG_{video}+L_{intra}	11.42	15.90	26.76	5.89
4	$Vanilla$+SFG_{video}+L_{inter}	11.58	16.00	27.06	4.67
5	$Vanilla$+SFG_{video}+L_{intra}+L_{inter}	11.71	16.30	28.21	3.98
6	blue$Vanilla$+SFG_{clip}+SFG_{video}+L_{intra}	11.83	16.41	28.54	3.90
7	blue$Vanilla$+SFG_{clip}+SFG_{video}+L_{inter}	11.80	16.54	28.67	3.20
8	$Vanilla$+SFG_{clip}+SFG_{video}+L_{intra}+L_{inter}	**12.24**	**16.97**	**28.89**	**2.51**

by 0.3%, 3.0% and 3.3% respectively, except for R@4 which is increased. We speculate that the increase in R@4 may be due to the absence of SFG_{clip}, but the precision benefits from the supervision of $Loss_{intra}$, which is similar to the representation of triplets in the captions. As for $Loss_{inter}$, it boosts all the metrics, especially R@4, suggesting that $Loss_{inter}$ helps video anchors exclude many static visual content and reduce repetition.

Effect of Anchor Number During the training phase, the initialization of the video anchors are only related to triplets from captions. However, during the testing phase, we currently select one frame at regular intervals from the video as an anchor. The difference is that the tokens in SFG are predefined in terms of quantity, and a random frame is used as the initial value. We conduct further analysis to determine the optimal number of anchors for the clip-level and video-level SFGs. As shown in Tab. 5, clip-level refers to setting one anchor for every 2/3 video frames, while video-level refers to setting 8/10 anchors in video-level SFG. It can be seen that selecting one video frame from every 2 frames for clip-level SFG and using 10 anchors in video-level SFG yield the best performance. When comparing 3 frames to 2 frames at the clip-level, the precision decreases, but R@4 is improved by 20% (in rows 1 & 2). This represents a trade-off between repetition and precision. For video anchors, using 10 anchors can cover most events in ActivityNet Captions while 8 anchors may miss some. Also, we test different ways of select anchors in the testing phase, with randomly selecting frames as anchors using a random seed or uniformly selecting over total frames. The results are as Tab. 4 and it can be observed that the random sampling method is not as effective as uniform sampling.

Table 4: Effects of Anchor Position in Testing Phase.

anchor position	B@4↑	METEOR↑	CIDER↑	R@4↓
random sample	10.3	15.8	28.56	3.12
uniform sample	12.24	16.97	28.89	2.51

Table 5: Effects of Frame Num. at Clip- and Video-levels.

Clip Level	Video Level	B@4↑	METEOR↑	CIDER↑	R@4↓
2	8	11.69	15.45	27.35	4.93
3	8	11.54	15.23	27.21	4.12
2	10	12.24	16.97	28.89	2.51
3	10	11.86	16.01	27.54	2.49

Table 6: Effects of Anchors & Fusion in Decoder.

Decoder	B@4↑	METEOR↑	CIDER↑	R@4↓
w/o anchors	11.95	16.34	28.56	3.12
add	12.12	16.01	28.23	3.09
concatenate	12.24	16.97	28.89	2.51

Figure 3: Qualitative Results1.

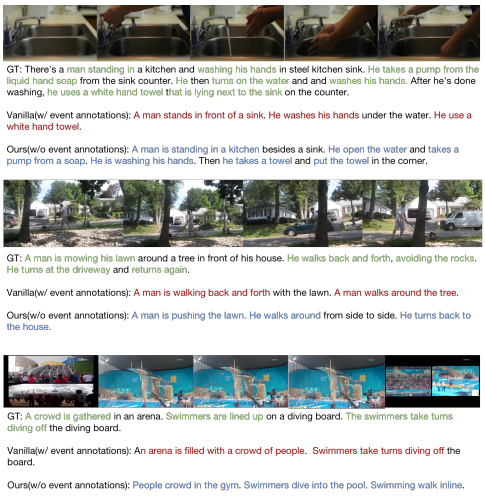

Figure 4: Qualitative Results2.

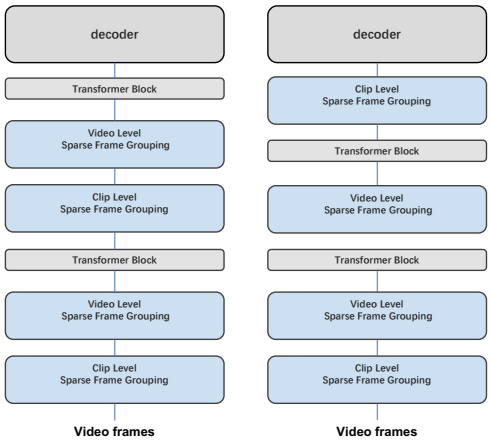

Figure 5: Different Model Structures.

5 Qualitative results

Fig. 4 displays example captions produced by our model on ActivityNet Captions. We compare them with *GT* (the Ground Truth) and those generated by the *Vanilla* models. The triplets in all paragraphs are highlighted for comparison. Compared to *Vanilla*, our model tends to generate more precise and action-driven descriptions, as evident in the examples. Particularly in the third case, where the surroundings are relatively static and only the athlete is performing different actions, our model is able to differentiate between these actions in the neighbouring frames.

For event distributions, in the first case, there are three events, and the second is momentary compared with the other two. The vanilla model fails to describe the second event even with annotation information. Our model, however, captures the action "is taken by" and decodes it to "give" in the generated description.

Effects of Fusion Ways in Decoder To utilize the dynamic anchors for generation guidance, we use the anchor attention in the decoder cross attention. Formally, the word embedding at the t-th step is used as a query to attend to the anchors, and the weighted anchors are fused by concatenating them with the visual representation as V. This V is then utilized to guide the decoding of the next step word in the sequence. We verify the effectiveness of anchors in the decoder and compare different fusion ways including *concatenating* and *adding*. As shown in Tab. 6, attending to the anchors indeed improves the relevance of the generated captions. Additionally, the results indicate that concatenation is more effective compared to addition.

Effects of Different Model Structures We also adjust the placements of the clip-level and video-level SFG modules. In the left encoder structure shown in Fig. 5, the order of the modules is SFG_{clip}, SFG_{video} and transformer layer. The block (SFG_{clip}, SFG_{video}, transformer layer) is stacked twice in the structure. The right structure in Fig. 5 considers the information lost in the second SFG_{clip}, puts SFG_{video} first in the second layer

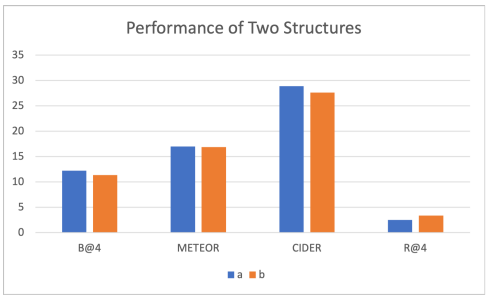

Figure 6: Performance of Model Structures. a and b represent the left and right structures in Fig. 5, respectively.

of transformer, and conducts SFG_{clip} after the last layer of transformer.

Comparing the two structures, the left outperforms the right. This suggests that even though SFG_{clip} discards some frames before SFG_{video}, the key information is retained, allowing SFG_{video} to learn the event contributions. Also frame sampling helps remove noise, which benefits SFG_{video} grouping center context of events. Fig. 6 shows the performance of the two structures.

6 Conclusions

In this work, we present a network with Sparse Frame Grouping (SFG) enhanced Transformer via Contrastive Learning for event annotation-free video paragraph captioning. The SFG module is designed to select and group informative frames and learn event distributions without relying on event annotations. Additionally, an Intra- and Inter-Contrastive Learning technique is proposed to train the SFG, enabling it to learn the overall event context in the video and distinguish it from the static context. Experimental results on two benchmark datasets demonstrate that SFGTransformer achieves SOTA performance. In the future, we plan to extend the method to longer video data such as movie data to further investigate the influence of key action visual information. And we also investigate training method without triplets to simplify the pre-process of data.

Limitations

Video paragraph captioning requires semantic consistency between the generated captions and the video content. Although the "action-guided frame grouping" method provides some guidance for frame selection and grouping, there are still potential instances of semantic inconsistency. This is especially true when there are multiple related yet distinct actions in the video, which may result in less accurate or consistent captions in terms of semantic coherence.

Acknowledgements

This work was supported by National Science and Technology Innovation 2030 - Major Project (No. 2021ZD0114001; No. 2021ZD0114000), National Natural Science Foundation of China (No. 61976057; No. 62172101), the Science and Technology Commission of Shanghai Municipality (No. 21511101000; No. 22DZ1100101), and the Fundamental Research Funds for the Central Universities of China (No. 2023110139). Yuejie Zhang, Rui Feng and Tao Zhang are corresponding authors.

References

D. Tran B. Korbar and L. Torresani. 2019. Scsampler: Sampling salient clips from video for efficient action recognition. In *ICCV*, pages 6232–6242.

H. Hu B. Zhang and F. Sha. 2018. Cross-modal and hierarchical modeling of video and text. In *ECCV*, pages 374–390.

S. Gu E. Jang and P. Ben. 2016. Categorical reparameterization with gumbel-softmax. *ICLR*.

Shreyank N Gowda, Marcus Rohrbach, and Laura Sevilla-Lara. 2021. Smart frame selection for action recognition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 1451–1459.

B. Tamara L J. Lei and B. Mohit. 2022. Revealing single frame bias for video-and-language learning. *arXiv preprint arXiv:2206.03428*.

Y. Shen D. Yu B. Tamara J. Lei, L. Wang and B. Mohit. 2020. Mart: Memory-augmented recurrent transformer for coherent video paragraph captioning. In *ACL*, pages 2603–2614.

Z. Ren N. Xu J. Mun, L. Yang and B. Han. 2019. Streamlined dense video captioning. In *CVPR*, pages 6588–6597.

R. Frederic F. Li K. Ranjay, H. Kenji and C. Juan. 2017. Dense-captioning events in videos. In *ICCV*, pages 706–715.

Y. Li L. Wang and L. Svetlana. 2016. Learning deep structure-preserving image-text embeddings. In *CVPR*, pages 5005–5013.

C. Xu L. Zhou and C. Jason J. 2018. Towards automatic learning of procedures from web instructional videos. In *AAAI*.

J. Corso R. Socher L. Zhou, Y. Zhou and C. Xiong. 2018. End-to-end dense video captioning with masked transformer. In *CVPR*, pages 8739–8748.

X. Chen C. Jason J L. Zhou, K. Yannis and R. Marcus. 2019. Grounded video description. In *CVPR*, pages 6578–6587.

D. Trevor P. Jae Sung, R. Marcus and R. Anna. 2019. Adversarial inference for multi-sentence video description. In *CVPR*, pages 6598–6608.

Z. Lu F. Zheng R. Cheng T. Wang, R. Zhang and P. Luo. 2021. End-to-end dense video captioning with parallel decoding. In *ICCV*, pages 6847–6857.

S. Chen Y. Song and Q. Jin. 2021. Towards diverse paragraph captioning for untrimmed videos. In *CVPR*, pages 11245–11254.

B. Dai Y. Xiong and D. Lin. 2018. Move forward and tell: A progressive generator of video descriptions. In *ECCV*, pages 468–483.

Serena Yeung, Olga Russakovsky, Greg Mori, and Li Fei-Fei. 2016. End-to-end learning of action detection from frame glimpses in videos. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2678–2687.

Y. Yang C. Jaime G L. Quoc Z. Dai, Z. Yang and S. Ruslan. 2019. Transformer-xl: Attentive language models beyond a fixed-length context. In *ACL*, pages 2978–2988.

W. Limin Z. Yuan, T. Zhan and W. Gangshan. 2021. Mgsampler: An explainable sampling strategy for video action recognition. In *CVPR*, pages 1513–1522.

Kaiyang Zhou, Yu Qiao, and Tao Xiang. 2018. Deep reinforcement learning for unsupervised video summarization with diversity-representativeness reward. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32.