VideoXum: Cross-Modal Visual and Textural Summarization of Videos

Jingyang Lin, *Graduate Student Member, IEEE*, Hang Hua, Ming Chen, Yikang Li, Jenhao Hsiao, Chiuman Ho, and Jiebo Luo, *Fellow, IEEE*

***Abstract*—Video summarization aims to distill the most important information from a source video into either an abridged video clip or a textual narrative. Existing methods often treat the generation of video and text summaries as independent tasks, thus neglecting the semantic correlation between visual and textual summarization. In other words, these methods only study a single modality as output without considering coherent video and text as outputs. In this work, we first introduce a novel task: cross-modal video summarization. This task seeks to transfer a long video into a condensed video clip and a semantically aligned textual summary, collectively referred to as a cross-modal summary. We then establish VideoXum (X refers to different modalities), a new large-scale human-annotated video benchmark for cross-modal video summarization. VideoXum is reannotated based on ActivityNet Captions with diverse open-domain videos. In the current version, VideoXum provides 14 K long videos, with a total of 140 K pairs of aligned video and text summaries. Compared to existing datasets, VideoXum offers superior scalability while preserving a comparable level of annotation quality. To validate the dataset's quality, we provide a comprehensive analysis of VideoXum, comparing it with existing datasets. Further, we perform an extensive empirical evaluation of several state-of-the-art methods on this dataset. Our findings highlight the impressive generalization capability of the vision-language encoder-decoder framework yields on VideoXum. Particularly, we propose VTSUM-BLIP, an end-to-end framework, serving as a strong baseline for this novel benchmark. Moreover, we adapt CLIPScore for VideoXum to measure the semantic consistency of cross-modal summaries effectively.**

***Index Terms*—Cross-modal video summarization, video captioning, video summarization.**

I. Introduction

VIDEO summarization, which is known as generating a concise summary that conveys the primary parts of a full-length video, is a profound challenge for video analysis. Practical automatic video summarization systems have a great potential impact on numerous applications, e.g., movie trailer generation [1] and narrative generation [2]. Typical approaches of video summarization extract essential clips or frames from a given long video [3], [4], [5]. Alternatively, the principal video content can also be summarized in natural language, e.g., video captioning [6], [7], [8]. However, previous works treat either visual or textual summarization as separate tasks and thus ignore the semantic correlation between these two modalities of summarization. Therefore, these methods lack the ability to generate aligned visual textual summaries. An earlier attempt [9] seeks to simultaneously generate visual and textual summaries from long videos. Still, the generated visual textual summaries in this work are not guaranteed to be semantically aligned since the two tasks were treated as separate, and there were no paired video and text summarization data for training or testing.

In this study, we first introduce a novel cross-modal video summarization task, which involves generating visual and textual summaries with semantic coherence. To facilitate this new task, we propose **VideoXum**, an enriched large-scale dataset for cross-modal video summarization. The dataset is built on ActivityNet Captions [8], a large-scale public video captioning benchmark consisting of 200 distinct activity categories. These activity classes belong to 5 different top-level video topics: "Eating and Drinking", "Sports, Exercises, and Recreation", "Socializing, Relaxing, and Leisure", "Personal Care", and "Household". To ensure consistent annotations, we hire workers to annotate ten shortened video summaries for each long source video according to the corresponding captions. Consequently, VideoXum contains 14 K long videos with 140 K pairs of aligned video and text summaries. Our goal is to extend the traditional single-modal video summarization task to a cross-modal video summarization task. Fig. 1 presents this novel task termed V2X-SUM (Video-to-X Summarization), where X denotes the modality of generated summaries. According to the target modality, we categorize the V2X-SUM task into three subtasks:

Video-to-Video Summarization (V2V-SUM): This task requires models to identify key segments from a source video and produce an abridged version.

Video-to-Text Summarization (V2T-SUM): In this task, models need to summarize the main content of the source video into a brief text description.

Video-to-Video&Text Summarization (V2VT-SUM): This task requires models to achieve V2V-SUM and V2T-SUM tasks simultaneously. Moreover, the semantics of these two modalities of summaries should be well aligned.

Manuscript received 31 July 2023; revised 8 October 2023; accepted 12 November 2023. Date of publication 29 November 2023; date of current version 21 March 2024. The Associate Editor coordinating the review of this manuscript and approving it for publication was Wu Liu. *(Jingyang Lin and Hang Hua contributed equally to this work.) (Corresponding author: Jiebo Luo.)*

Jingyang Lin, Hang Hua, and Jiebo Luo are with the Department of Computer Science, University of Rochester, Rochester, NY 14627 USA (e-mail: jlin81@ur.rochester.edu; hhua2@ur.rochester.edu; jluo@cs.rochester.edu).

Ming Chen, Yikang Li, Jenhao Hsiao, and Chiuman Ho are with the OPPO US Research Center, Palo Alto, CA 94303 USA (e-mail: cmelf0819@gmail.com; lyk010632@gmail.com; mhsiao.pro@gmail.com; chiuman100@gmail.com).

The project page is https://videoxum.github.io/.

Digital Object Identifier 10.1109/TMM.2023.3335875

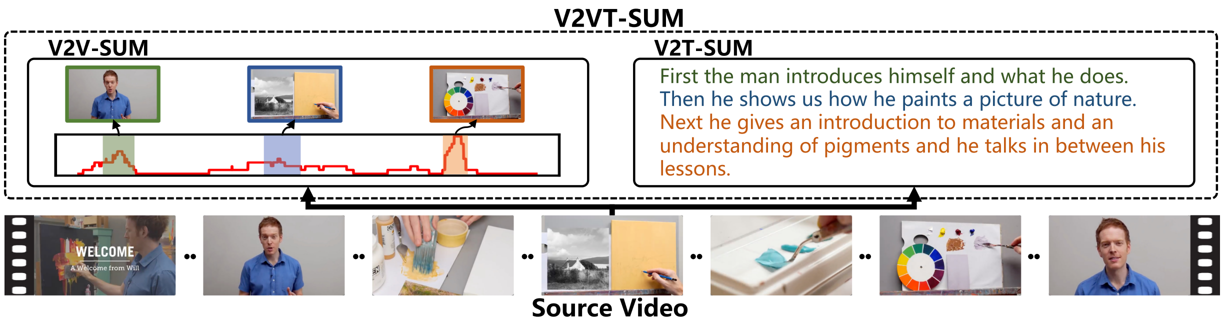

Fig. 1. Illustration of the V2X-SUM tasks. A full-length source video (*bottom*) can be summarized into a shortened video and a text narrative (*top*). This task requires semantic alignment between the video and text summaries.

Compared with single-modal summarization tasks, cross-modal video summarization comes with its own challenges. We summarize three primary challenges for this new task. First, the scarcity of large-scale, diverse, and well-annotated cross-model video summarization benchmarks presents a significant hurdle for researchers in promoting the corresponding techniques. Second, from the perspective of optimization, it is nontrivial to ensure the stability of the training process that accommodates both tasks concurrently. Specifically, a stable training process could facilitate learning of each single-modal task, thereby improving the overall performance. Third, either assuring or evaluating the semantic coherence between the generated video and text summaries is challenging.

To establish strong baseline models for this emerging task, we propose VTSUM-BLIP, an end-to-end cross-modal video summarization model. To leverage the strong capability of vision-language pretrained (VLP) models for vision understanding and language modeling, we employ BLIP [10] as our foundational backbone. This VLP encoder-decoder architecture provides a superior initialization, which is crucial for stable and effective optimization in machine learning models [11]. Inspired by efficient video encoding techniques [12], [13], [14], [15], we design an efficient hierarchical video encoding strategy, incorporating a frozen encoder, a temporal modeling module, and a context aggregation module to encode long videos. The video encoder is followed by different task-specific decoders for video and text summarization. The modularized design enables us to perform more complex downstream tasks without changing the structure of the pretrained backbone. Existing multimodal-based video summarization works [9], [16] follow a pipeline where a summary is first generated in one modality, and then this generated summary is served as a prompt to improve the summary in another modality. Such methods may suffer from bias accumulation issues since they do not consider the semantic coherence of summaries in two modalities. In contrast, our proposed VTSUM-BLIP enables joint training of the video and text summarization decoders in parallel. In other words, the predictions of these two decoders avoid sequential dependency between the two modalities of summaries. Furthermore, the video and text summarization decoders collaboratively influence the shared parameters during training, allowing the framework to learn the semantic coherence between two tasks.

Our proposed framework achieves promising performance on VideoXum, as well as other existing single-modal video summarization datasets (i.e., TVSum [3], SumMe [4], and ActivityNet Captions [8]). Inspired by the CLIPScore [17] and its video-text variant [18], we adapt these metrics for the VideoXum and propose VT-CLIPScore for evaluating the semantic coherence of cross-modal summaries. The empirical results show the consistency of the proposed metric with human evaluation.

Our main contributions can be summarized as follows:

- We introduce **VideoXum**, an enriched large-scale dataset, to bridge the modality gap between the video and text summarization. The dataset contains 14 K long videos with corresponding human-annotated video and text summaries. We conduct comprehensive experimental analyses to verify the rationality of our proposed new dataset.
- Based on VLP encoder-decoder architecture, we propose an end-to-end cross-modal video summarization framework – VTSUM-BLIP to establish strong baseline models for this novel task. The models achieve promising results on VideoXum and the new state of the art on several existing single-modal video summarization datasets.
- We propose an evaluation metric VT-CLIPScore on the VideoXum benchmark to evaluate cross-modal semantic consistency. The empirical results show the high consistency of our proposed metric with human evaluation.

II. Related Work

A. *Video Summarization*

Video summarization datasets (e.g., SumMe [4], TVSum [3], and YouTube [5]) have enabled the development of state-of-the-art video summarization methods [16], [19], [20], [21], [22]. Among these models, vsLSTM [19] first attempted to learn frame importance by modeling the temporal dependency among frames using LSTM [23] units. The model can be combined with a determinantal point process (DPP) to improve the diversity of generated video summary. Following vsLSTM, several other approaches were proposed to model the temporal dependency, e.g., H-RNN [24], HSA-RNN [25], DASP [26]. Another solution models the spatiotemporal structure of the video to learn frame importance, such as MerryGoRoundNet [27], and CRSum [28]. Adversarial learning-based methods [29], [30] can also perform well. Recently, multimodal-based video summarization method [16] leverages generated text summaries to promote predictions of frame-level scores for video summaries. Different from multimodal-based video summarization, the cross-modal video summarization task requires simultaneously producing both visual and textual summaries from a source video, which goes beyond generic video summarization. Moreover, it ensures semantic coherence between these two modalities.

B. Video Captioning

Video Captioning aims to describe a video with text, which requires the capability of understanding actions and events. Existing benchmarks (e.g., MSVD [31], YouCook [6], MSR-VTT [7], and ActivityNet Captions [8]) have helped to promote the ability of language models to generate reasonable captions for video. Benefiting from these human-annotated datasets, many novel approaches are proposed. Attention-based methods [32], [33] employ attention mechanisms to help the model in associating relevant frames since not every frame in a video is equally important. DENSE [8] is an early attempt at dense video captioning, which detects events with an event proposal module and associates them with LSTM. Wang et al. [34] develop a bidirectional process to encode context for detecting event proposals. Moreover, Masked Transformer [35] proposes a differentiable masking scheme to ensure consistency between event proposal and caption generation modules.

C. Multimodal Pretraining

Large language models (LLMs) [36], [37], [38], [39] have revolutionized NLP research in recent years. Following the large-scale pretraining models in the field of NLP, numerous works [40], [41], [42], [43] on exploring the combination of vision and language (VL) pretraining have achieved great success. Since then, image-text pretraining has become a default approach to tackling VL tasks [44], [45], [46], [47]. In addition, the introduction of Vision Transformers [48] enables vision and language modalities to be jointly modeled by Transformers in a more scalable fashion [49], [50], [51], [52]. According to the encoding strategies for image and language modalities, VL models can be categorized into fusion encoder [53], [54], [55], [56], dual encoder [57], and a combination of both [58], [59], [60]. Several video-language pretrained models have also shown strong performance on video captioning and other video tasks, such as HERO [12], VideoBERT [61], and UniVL [62]. In this work, cross-modal video summarization requires models with strong video understanding and language modeling capabilities. Therefore, this new task provides a practical scenario to assess the superiority of multimodal pretrained models.

III. Dataset

In this section, we introduce the proposed VideoXum dataset. The dataset is reannotated by a limited number of workers, including 14,001 long videos with 140,010 video and text summaries pairs. We describe the process of dataset collection and annotation strategy. We also provide several quantitative and qualitative analyses of the proposed dataset. Finally, we compare the VideoXum with existing single-modal video summarization datasets.

A. Dataset Curation

Dataset Collection: The VideoXum dataset is built based on ActivityNet Captions [8], a high-quality public video captioning benchmark. There are three primary reasons to build upon ActivityNet Captions. First, the dataset contains 20 K real-life Youtube videos with diverse content, in terms of rich topics, different photographic devices, multiple view angles, and so on. Each video in this dataset is annotated with a series of dense temporal segments, and each segment corresponds to a concrete sentence description, offering diverse patterns essential for video understanding and generation tasks. Second, the dataset contains numerous lengthy videos in Fig. 2(a), which introduces more challenges to the cross-modal video summarization task. Third, as described in Section I, the well-annotated sentence narratives are natural summaries of the source videos. Therefore, the content and length of videos in the ActivityNet Captions dataset largely meet our requirements and provide an ideal foundation for constructing our cross-modal video summarization benchmark. To maintain our focus on long videos, we filter out videos shorter than 10 seconds.

Dataset Reannotation: For each video, we expect the total length of its video summary to be bounded to 15% of the source video, along with a semantically aligned text summary. ActivityNet Captions [8] already contains video captions with temporal segments for long videos. Therefore, we concatenate the caption sentences as a text summary for the long source video. However, the annotated video spans, which cover an average of 94.6% of the source videos, are too long to be regarded as a video summary by themselves since video summaries need to be much more concise. Therefore, we reannotate the video spans and obtain an abridged version of video segments for better aligning with the sentence captions.

Due to the inherently subjective nature of summarizing a long video (this conclusion is also reflected by the human performance on V2V-SUM in Table III), it is hard to obtain perfect ground truth labels for this task. Following previous works [3], [63], [64], we required ten different workers to annotate video summary spans corresponding to a same text description. For each given caption, we obtained ten shortened spans. During the evaluation, we compared the prediction with all ten annotations and then obtain the average score for each video. To further ensure consistent annotations, we hired 40 workers in total to reannotate all 140,010 summarised video spans over a period of two months. On average, each worker reannotated about 15 videos per hour. To maintain high-quality annotations, we regularly reviewed the reannotated video spans and provided feedback to workers. Every 24 hours, we randomly evaluated 15% of an annotation batch for accuracy. If the acceptance rate of the sampled annotations reached 90%, we considered the entire annotation batch as passed; otherwise, we asked workers to reannotate the batch.

This reannotation pipeline aims to obtain an abridged version (ideally bounded to 15%) of videos for better aligning with the sentence captions. Therefore, we filter the initial ActivityNet dataset using the length compression ratio of video with 20% as the threshold. The video length compression ratio is calculated

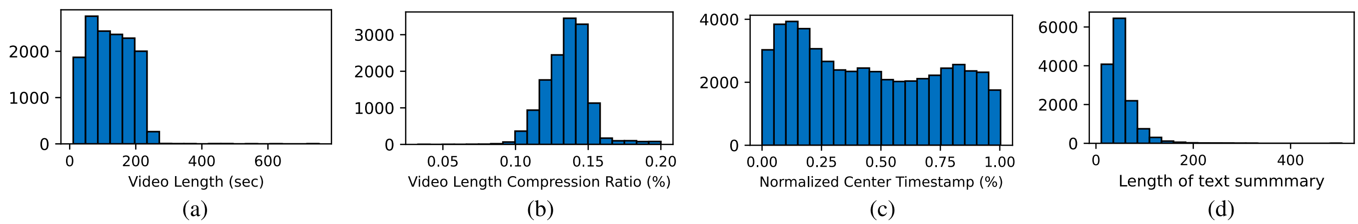

Fig. 2. Statistical information of VideoXum dataset: (a) distribution of video length; (b) distribution of video length compression ratio; (c) distribution of normalized center timestamp; (d) distribution of length of text summary.

TABLE I
COMPARISON WITH EXISTING SINGLE-MODAL VIDEO-TO-VIDEO SUMMARIZATION AND VIDEO-TO-TEXT SUMMARIZATION DATASETS

Dataset	Domain	# Videos	Avg Ratio (%) VideoSum	Avg Len TextSum	Supported Task V2V-Sum	V2T-Sum	V2VT-Sum
MSVD [31]	open	1,970	-	8.7	✗	✓	✗
YouCook [6]	cooking	88	-	15.9	✗	✓	✗
MSR-VTT [7]	open	7,180	-	18.6	✗	✓	✗
ActivityNet [8]	open	20,000	-	40	✗	✓	✗
SumMe [4]	3 categories	25	15% <	-	✓	✗	✗
Youtube [5]	5 categories	50	15% <	-	✓	✗	✗
TVSum [3]	10 categories	50	15% <	-	✓	✗	✗
VideoXum (ours)	open	14,001	13.6%	49.9	✓	✓	✓

as Ratio$(S, V) = \frac{|S|}{|C|}$, where $|S|$ denotes the length of summary, $|C|$ denotes the length of source video. Finally, 14,001 long videos remain in our dataset.

Dataset Split: We split the dataset into training, validation, and test sets. The split strategy also guarantees that all three data splits preserve the same distribution of video length. In particular, the dataset is divided into 8,000, 2,001, and 4,000 videos in the training, validation, and test sets, respectively.

B. Dataset Statistics

Fig. 2 presents the statistical information of the VideoXum dataset. As shown in Fig. 2(a), it shows that the length of the videos ranges from 10 to 755 seconds, with 99.9% of them under 300 seconds. The average length is 124.2 seconds, and the median length is 121.6 seconds. For the video summarization task, most video summary lengths are shorter than 15% of the source video length. Fig. 2(b) shows that the average length compression ratio is 13.6%, with a median ratio of 13.7%, and a maximum ratio of 20%. Moreover, we investigate the distribution of the center timestamps of important clips. All the center timestamps are normalized to fall within the range of $[0, 1.0]$ according to the original video length. Fig. 2(c) suggests that the important clips are generally uniformly distributed throughout the video, with a mild peak at the beginning. Therefore, the VideoXum dataset does not suffer from temporal bias issues [63]. For the text summarization task, each video is summarized into a narrative paragraph that describes multiple events. On average, each narrative paragraph contains 49.9 words. Fig. 2(d) indicates that most (98%) text summaries are shorter than 128 words, which guides us to set the maximum text generation length as 128.

TABLE II
COMPARISON OF F1 SCORE ON *HUMAN ANNOTATION*

VideoXum (ours) F1-Avg	F1-Max	SumMe F1-Avg	F1-Max	TVSum F1-Avg	F1-Max
36.2	59.5	31$^\sharp$	54$^\sharp$	54$^\sharp$	78$^\sharp$

F1-avg denotes the averaged F1 score across all reference summaries. F1-max represents the maximum F1 score. Symbol ♯ denotes the results directly quoted from [65].

C. Comparison With Existing Single-Modal Video Summarization Datasets

In Table I, we compare the proposed VideoXum dataset with existing *single-modal* video-to-video and video-to-text[1] summarization datasets. The main difference between VideoXum and other existing datasets is that VideoXum contains aligned human-annotated video and text summaries, while others only have single-modal summaries for source videos. Compared with the existing video summarization benchmarks (e.g., SumMe [4] and TVSum [3]), the amount of data in the VideoXum dataset is significantly larger. In addition, VideoXum contains open-domain videos with more diverse scenarios than other datasets. To ensure the quality of human annotation, we evaluate annotated data using a leave-one-out strategy [65]. Table II shows that our annotation quality is comparable with existing benchmarks.

IV. METHODOLOGY

A. Problem Formulation

We formulate the problem of cross-modal video summarization as a multi-task learning problem, including V2V-SUM and

[1]In this paper, we regard the video captioning task as video-to-text summarization task

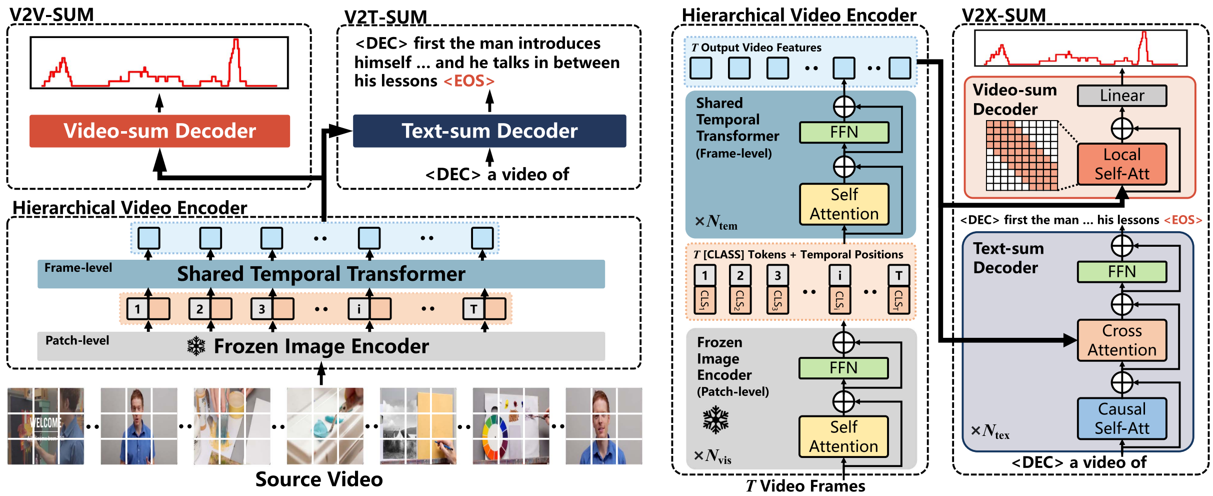

Fig. 3. Overview of our VTSUM-BLIP framework (*left*). It consists of a hierarchical video encoder (*middle*), video-sum decoder, and text-sum decoder (*right*). For V2V-SUM, the video-sum decoder employs a temporal Transformer and local self-attention module to aggregate the local context. For V2T-SUM, the text-sum decoder is a pretrained BLIP text decoder.

V2T-SUM. Given a video $\mathcal{V} = \{\boldsymbol{v}_i\}_{i=1}^{T}$, T is the number of frames in the video and $\boldsymbol{v}_i$ denotes the i-th frame in the temporal order. Our goal is to learn a shared video encoder $f(\cdot;\theta)$ followed by two task-specific decoders, including a video summarization (video-sum) decoder $g_v(\cdot;\theta_v)$ and a text summarization (text-sum) decoder $g_t(\cdot;\theta_t)$. In particular, the notations of θ, θ_v, and θ_t represent the learnable parameters of the shared video encoder, video-sum decoder, and text-sum decoder, respectively. We first feed the input video $\mathcal{V}$ into the shared video encoder to produce the video features $\tilde{\mathcal{Z}}$:

$$\tilde{\mathcal{Z}} = f\left(\mathcal{V}, \mathcal{E}_{\text{temp}}; \theta\right), \tag{1}$$

where $\mathcal{E}_{\text{temp}}$ is the temporal position embedding for the video frames. Given the video features $\tilde{\mathcal{Z}}$, the model generates a video summary $\mathcal{V}_{\text{sum}}$ and a text summary $\mathcal{T}_{\text{sum}}$. In particular, we can formulate visual and narrative outcomes as follows:

$$\mathcal{V}_{\text{sum}} = g_v\left(\tilde{\mathcal{Z}}; \theta_v\right), \tag{2}$$

$$\mathcal{T}_{\text{sum}} = g_t\left(\tilde{\mathcal{Z}}, \mathcal{T}_{\text{prompt}}; \theta_t\right), \tag{3}$$

where $\mathcal{T}_{\text{prompt}}$ denotes a prompt sequence.

B. Cross-Modal Video Summarization

Our proposed VideoXum benchmark requires models with strong video understanding and language modeling capabilities. To this end, we employ the large vision-language pretrained (VLP) model BLIP [10] as our backbone. For efficient video encoding, we propose a hierarchical video encoder to capture spatiotemporal features. Followed by the video encoder, video-sum and text-sum decoders are designed for V2V-SUM and V2T-SUM tasks, respectively. The overall framework termed VTSUM-BLIP is shown in Fig. 3.

Hierarchical Video Encoder: The hierarchical video encoder $f(\cdot;\theta)$ aims to address the challenge of efficiently extracting spatiotemporal visual features from a long video. Drawing the inspiration from efficient video encoding [12], [13], [14], [15] and long document summarization [66], we formulate the BLIP image encoder into a hierarchical architecture for long video encoding without changing the structure of the encoder. This enables us to efficiently obtain rich video features at both video frame and image patch levels. Specifically, given a video $\mathcal{V} = \{\boldsymbol{v}_i\}_{i=1}^{T}$ with T-frame, the frozen image encoder projects each video frame $\boldsymbol{v}_i$ into the representation space and produce T visual tokens $\mathcal{Z} = \{\boldsymbol{z}_i\}_{i=1}^{T}$. Next, we use temporal position embedding $\mathcal{E}_{\text{temp}} = \{\boldsymbol{e}_i\}_{i=1}^{T}$ with the shared **T**emporal **T**ransformer (TT) to model the temporal information for the video sequence. In this way, we can obtain spatiotemporal visual features $\tilde{\mathcal{Z}} = \{\tilde{\boldsymbol{z}}_i\}_{i=1}^{T}$.

To better understand the hierarchical video encoder, we break it down into two key components:

- *Frozen Image Encoder:* Following the previous works [14], [43], [67], we freeze the parameters of the pretrained BLIP encoder, which can help to improve the training time and GPU memory efficiency for encoding long videos. In detail, we first convert input images into several patches as the input tokens for the N_{vis}-layer BLIP encoder. The patch embedding is prepended with a `[CLS]` token in the representation space. Next, we take all output of the `[CLS]` tokens as the representation of the input frames. We can compress the input video at the frame level through the hierarchical encoding strategy and generate the representation $\mathcal{Z}$.
- *Shared Temporal Transformer:* After obtaining a sequence of the video frame representation $\mathcal{Z} = \{\boldsymbol{z}_i\}_{i=1}^{T}$, we add these temporal position embeddings $\mathcal{E}_{\text{temp}} = \{\boldsymbol{e}_i\}_{i=1}^{T}$ to $\mathcal{Z}$, and feed them into the shared temporal Transformer (TT) for temporal modeling and get the spatiotemporal visual features $\tilde{\mathcal{Z}} = \{\tilde{\boldsymbol{z}}_i\}_{i=1}^{T}$ in (1):

$$\begin{aligned} \boldsymbol{z}_i^{(0)} &= \boldsymbol{z}_i + \boldsymbol{e}_i, \\ \boldsymbol{z}_i^{(l)} &= \mathrm{TT}^{(l)}\left(\boldsymbol{z}_1^{(l-1)}, \ldots, \boldsymbol{z}_T^{(l-1)}\right), \ l = 1, \ldots, N_{tem}, \\ \tilde{z}_i &= \boldsymbol{z}_i^{(N_{tem})}, \end{aligned} \tag{4}$$

where l indicates the l-th block of the temporal Transformer, and N_{tem} denotes total block number of the temporal Transformer.

Video-Sum Decoder: Inspired by long document encoding technique [13], the video-sum decoder $g_v(\cdot;\theta_v)$ employs a **C**ontext **A**ggregation (CA) module that captures context from neighboring frames with local self-attention. In particular, we first define a fixed-size slice window at each temporal position and then construct a binary local attention map $M^{LA} \in \{0,1\}^{T\times T}$ with a given window size ε. For example, Fig. 3 (*right*) presents a local attention map with a window size $\varepsilon = 7$. Next, we compute the local attention features $\mathcal{A}_{loc}$:

$$\mathcal{A}_{loc} = \left(\text{softmax}\left(\frac{QK^T}{\sqrt{d}}\right) \odot M^{LA}\right) V, \tag{5}$$

where $\odot$ is element-wise multiplication, and queries Q, keys K, and values V are d-dimensional features generated from temporal-aware visual features $\tilde{\mathcal{Z}}$. Finally, we feed local attention-enhanced features into a linear classifier to obtain the predictions of the frame-level importance scores $\{p_i\}_{i=1}^T$. Our training objective for video summarization is an averaged binary cross-entropy loss, as following:

$$\mathcal{L}_v = -\frac{1}{T}\sum_{i=1}^{T}\left(\hat{y}_i \cdot \log(p_i) + (1-\hat{y}_i)\cdot\log(1-p_i)\right), \tag{6}$$

where $\hat{y}_i \in \{0,1\}$ denotes whether the i-th frame is a key frame, and p_i indicates the predicted importance score of the i-th frame. Finally, we select the top 15% of frames to attain a video-sum result $\mathcal{V}_{\text{sum}}$ in (2) from a long video.

Text-Sum Decoder: The pretrained BLIP text decoder is a strong baseline for text generation. The text summarization decoder $g_t(\cdot;\theta_t)$ contains N_{tex} stacked Transformer decoder blocks with cross-attention modules. During the decoding process, the text decoder takes a prompt sequence $\mathcal{T}_{\text{prompt}}$, and the video features $\tilde{\mathcal{Z}} = \{\tilde{z}_i\}_{i=1}^T$ from the video encoder as inputs and then generate the final text summary $\mathcal{T}_{\text{sum}}$ in (3). The training objective of text summarization is negative log-likelihood (NLL), which can be expressed in the equation as:

$$\mathcal{L}_t = -\sum_{i=1}^{N_{tex}} \log P\left(w_i | w_0, w_1, \ldots, w_{i-1}, \tilde{\mathcal{Z}}\right), \tag{7}$$

where w_i denotes the i-th word in the sentence, N_{tex} is the length of output sequence.

C. Overall Objective

Following the multi-task learning paradigm, the overall objective of our proposed framework is calculated as the integration of video-sum loss $\mathcal{L}_v$ and text-sum loss $\mathcal{L}_t$:

$$\mathcal{L} = \lambda_v\mathcal{L}_v + \lambda_t\mathcal{L}_t, \tag{8}$$

where λ_v and λ_t are the weights of different summary tasks.

V. Experiments

In this section, we first introduce the baseline models and experimental setup for the proposed VideoXum dataset. Then, we present the evaluation metrics and human evaluation strategy. In addition, we report several baseline models' performances under different settings and present a comprehensive experimental analysis to prove the effectiveness of our proposed method.

A. Baseline Models

We introduce all the baseline models listed in Table III:

Frozen-BLIP: refers to inference over the test set using a frozen BLIP model without training. We take this zero-shot setting performance as a lower bound for our benchmark.

VSUM-BLIP (Base): is a baseline model to perform video-to-video summary. It consists of a frozen BLIP encoder and a learnable video-sum decoder.

TSUM-BLIP (Base): is a video-to-text summary baseline. We employ the vanilla BLIP model with a frozen encoder.

VTSUM-BLIP (Base): combines the VSUM-BLIP and TSUM-BLIP modules of the model. It is comprised of a shared frozen encoder and two task-specific decoders.

Temporal Transformer (TT): is a crucial module to achieve the hierarchical encoding for videos while incorporating temporal information into a video sequence. Specifically, we use several Transformer layers combined with temporal positional embedding to model the temporal information.

Context Aggregation (CA): is a plug-and-play module to model the video frame representations for the V2VSum task. Compared with the baseline models, this mechanism enhances the local context information for video representations and could help reduce the redundancy of video summaries.

B. Experimental Setup and Implementation Details

Data Preprocessing: Video frames of all *train/val/test* sets are first resized using bi-linear resampling to 224 pixels along the shorter side. Next, a 224×224 center crop is applied to the resized frames. This is a common preprocessing method. For each training batch, we add padding to all video sequences to make them the same length, enabling the videos to be processed in parallel and speeding up the training process. In addition, the padding tokens are masked out during the self-attention calculation. Based on data statistics in Fig. 2(a), we set the maximum video length to 512, and frames exceeding the maximum length are truncated. For each text summary, we concatenate (dense) sentence captions in a video to construct a narrative paragraph [68], [69], [70]. According to data statistics in Fig. 2(d), we set the maximum generation length to 128 in the text summarization task.

Model Architecture: We employ ViT-B/16 [48] as the image encoder backbone with $N_{\text{vis}} = 12$ layers. The N_{tem}-layer Temporal Transformer (TT) follows the image encoder, where N_{tem} is 1. The temporal positional embeddings $\varepsilon_{\text{temp}}$ in (1) are also learnable. The video-sum decoder contains a Context Aggregation (CA) module capturing local context and a binary linear classifier. The CA module constructs a binary local attention map with window size $\epsilon = 5$. For the text-sum decoder, we adopt a variant of Transformer with $N_{\text{tex}} = 12$ layers, which replaces the bidirectional self-attention module with a causal self-attention module [48]. In addition, the prompt $\mathcal{T}_{\text{prompt}}$ of the text-sum decoder in (3) is set as "[DEC] a video of".

TABLE III
PERFORMANCE OF THE BASELINE MODELS ON THE VIDEOXUM DATASET *TEST* SET FOR THREE DIFFERENT V2XSUM TASKS

Method	V2V-SUM			V2T-SUM				V2VT-SUM
	F1 score	Kendall	Spearman	BLEU@4	METEOR	ROUGE-L	CIDEr	VT-CLIPScore
Frozen-BLIP	16.1	0.008	0.011	0.0	0.4	1.4	0.0	19.5
Single-Modal Video Summarization								
VSUM-BLIP (Base)	21.7	0.131	0.207	-	-	-	-	-
+ Temporal Transformer	22.1	0.168	0.222	-	-	-	-	-
+ Context Aggregation	22.2	0.172	0.228	-	-	-	-	-
+ TT + CA	23.1	0.185	0.246	-	-	-	-	-
TSUM-BLIP (Base)	-	-	-	5.5	11.7	24.9	18.6	-
+ Temporal Transformer	-	-	-	5.6	11.8	24.9	20.9	-
Cross-Modal Video Summarization								
Two-stage Manner	19.4	0.107	0.143	5.1	11.2	24.0	15.6	28.2
VTSUM-BLIP (Base)	21.7	0.131	0.207	5.5	11.7	24.9	18.6	28.4
+ Temporal Transformer	22.4	0.176	0.233	5.7	12.0	24.9	22.4	28.9
+ Context Aggregation	22.2	0.172	0.228	5.5	11.7	24.9	18.6	28.6
+ TT + CA	**23.5**	**0.196**	**0.258**	**5.8**	**12.2**	**25.1**	**23.1**	**29.4**
Human	33.8	0.305	0.336	5.2	14.7	25.7	24.2	38.0

The F1 score, BLEU@4, METEOR, ROUGE-L, CIDEr, and VT-CLIPScore are shown in %.

Weight Initialization: To initialize the weights of our model, we employ a state-of-the-art VLP model called BLIP [10]. The image encoder and the text-sum decoder are initialized by pre-trained $\text{BLIP}_{\text{CapFilt-L}}$. Additionally, the Temporal Transformer and video-sum decoder are randomly initialized.

Optimization: Due to limited computational resources, we finetuned all of the parameters in our proposed VTSUM-BLIP model, except for the image encoder. We adopt the AdamW [71] optimizer with an initial learning rate of 2×10^{-5} to optimize the model, and the $\beta_1 = 0.9, \beta_2 = 0.999$. The batch size is 64, and weight decay is 5×10^{-2}. The learning rate follows a cosine decay schedule [72] with the minimum learning rate of 0.0. We train the VTSUM-BLIP framework for 56 epochs with a batch size of 64 on 4 A100 GPUs. In addition, the weights of video-sum loss $\mathcal{L}_v$ and text-sum loss $\mathcal{L}_t$ are $\lambda_v = 15.0$ and $\lambda_t = 1.0$, respectively.

C. Evaluation

Video Summary Evaluation: Following previous works [16], [22], [65] for video summarization evaluation, we adopt the F1 score, Kendall's τ [73], and Spearman's ρ [74] as our automatic evaluation metrics.

Text Summary Evaluation: To evaluate the quality of generated text summaries for video **text summary**, we adopt several metrics for video captioning evaluation [75] including: BLEU [76], METEOR [77], ROUGE-L [78], CIDEr [79].

Video-text Semantic Consistency Evaluation: Apart from independently evaluating single-modal summaries, we also evaluate *semantic consistency* of text and video summaries. Inspired by previous works [17], [18], [80], we adapt CLIPScore for VideoXum benchmark and introduce a new evaluation metric – VT-CLIPScore for evaluating the text and video semantic consistency. Specifically, we finetune the vanilla CLIP model [57] on VideoXum dataset with contrastive learning strategies. It is worth noting that adapting CLIPScore to our proposed benchmark is necessary since there is a domain gap between the CLIP pretraining data and our VideoXum data. Therefore, finetuning the CLIP model on our data makes the evaluation score more reliable. Similar attempts of finetuning evaluation models (i.e., BERTScore [81] and Sentence-BERT [82]) also support the necessity of the VT-CLIPScore.

To facilitate reimplementation, we use the AdamW optimizer with an initial learning rate of 2×10^{-6} and a weight decay of 5×10^{-2}. We finetune the CLIP model for 50 epochs with a batch size of 16 on 4 GPUs. The empirical results in Table VII show that our proposed VT-CLIPScore is sensitive enough to the semantic change of video and text. Moreover, the results in Table VI indicate the high consistency of our proposed automatic evaluation metric with human evaluation.

D. Results on VideoXum

We conduct experiments on VideoXum using different baseline models. Table III shows the empirical results of the models on VideoXum. By comparing VSUM-BLIP (Base), TSUM-BLIP (Base), and VTSUM-BLIP (Base) with Frozen-BLIP, BLIP models show better results after finetuning on specific tasks. The comparison between the end-to-end VTSUM-BLIP (Base) and the Two-stage Manner [9] (i.e., first V2V-Sum and then V2T-Sum) demonstrates the superiority of the end-to-end framework since errors originating in the V2V-SUM stage could negatively influence the V2T-SUM stage. In all three tasks, the model with TT can help model the video sequence better, indicating that temporal information is necessary. The CA module can enhance the local information awareness of the model, which can help improve the performance of the V2V-SUM task. The performance gains of TT on V2V-SUM are more significant than that on V2T-SUM, one of the possible reasons is that the text decoder is a well-generalized model trained on a large corpus and is insensitive to the subtle changes of the input features [84], [85], [86]. A Base BLIP model combined with TT and CA achieves the SOTA results in our proposed three tasks. From the overall performance, our proposed multitask framework can benefit both V2V and V2T-SUM tasks. In addition, Table III shows the human performance on VideoXum.

Fig. 4. Two example results of the generated video and text summaries across different baseline models. Red (both line and box) indicates the results of the ground truth. Green indicates the results of the VTSUM-BLIP (Base). Blue indicates the results of VTSUM-BLIP (+TT+CA).

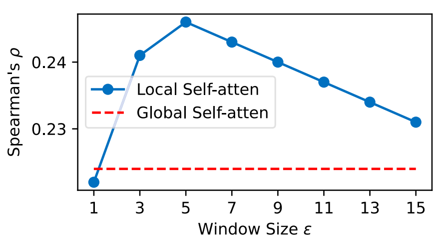

Fig. 5. Impact of local window size ε for Context Aggregation (CA) module.

The result is obtained on human-annotated reference summaries using a leave-one-out strategy [65], which measures the average consistency of human annotators. Although humans outperform all baseline models in most evaluation metrics on different tasks (except for BLEU@4), our proposed VTSUM-BLIP archives quite competitive results, especially on the V2T-SUM task. Fig. 4 visualizes some examples of generated video and text summaries, showing the effectiveness of TT and CA modules. Additionally, more visualization results are present in Fig. 7.

E. Experimental Analysis

Method Comparisons on Existing Benchmarks: To further evaluate the effectiveness of our proposed model, we conduct experiments on two well-known video summarization datasets, i.e., TVSum and SumMe. The results in Table IV show that the BLIP-VSUM (Base) achieves competitive results against several strong baselines. Moreover, our proposed mechanisms of Temporal Transformer and Context Aggregation can further improve video summarization performance. We also verify the ability of video-to-text summarization of our model on ActivityNet Captions. As we can see from Table V, our proposed model outperforms all the strong baseline models by a large margin in multiple evaluation metrics (2.9 in BLEU@4, 1.0 in METEOR, 7.4 in ROUGE-L, and 19.0 in CIDEr).

TABLE IV
COMPARISON WITH STATE-OF-THE-ART METHODS ON THE TVSUM AND SUMME DATASETS

Method	TVSum		SumMe	
	Kendall	Spearman	Kendall	Spearman
dppLSTM [19]	0.042	0.055	-	-
DR-DSN [20]	0.020	0.026	-	-
Sumgraph [21]	0.094	0.138	-	-
CLIP-it [16]	0.108	0.147	-	-
Standard ranker [22]	0.176	0.230	0.011	0.014
VSUM-BLIP (Base)	0.160	0.207	0.154	0.191
+ Temporal Transformer	0.182	0.239	0.266	0.330
+ Context Aggregation	0.185	0.243	0.268	0.332
+ TT + CA	**0.200**	**0.261**	**0.295**	**0.365**

TABLE V
COMPARISON WITH STATE-OF-THE-ART METHODS ON THE ACTIVITYNET CAPTIONS DATASET

Method	ActivityNet Captions			
	BLEU@4	METEOR	ROUGE-L	CIDEr
DENSE [8]	1.6	8.9	-	-
DVC-D-A [84]	1.7	9.3	-	-
Wang *et al.* [34]	2.3	9.6	19.1	12.7
Bi-LSTM+TempoAttn [35]	2.1	10.0	-	-
Masked Transformer [35]	2.8	11.1	-	-
Support-Set [70]	1.5	6.9	17.8	3.2
TSUM-BLIP (Base)	5.5	12.1	25.1	19.7
+ Temporal Transformer	**5.7**	**12.1**	**25.2**	**22.2**

TABLE VI
HUMAN EVALUATION OF THE BASELINE MODELS ON THE VIDEOXUM DATASET

Method	V2V-SUM	V2T-SUM		Cross-Modal
	Accuracy	Accuracy	Fluency	Consistency
VTSUM-BLIP (Base)	3.1	4.1	4.3	3.2
+ Temporal Transformer	3.5	4.2	4.2	3.2
+ Context Aggregation	3.2	4.1	4.3	3.1
+ TT + CA	**3.8**	**4.4**	**4.4**	**3.4**

Human Evaluation: We conducted a human evaluation of video/text summaries on 50 random samples, assessed by workers for quality and consistency across video and text summaries, scoring 1-5 (5 best). We report the average score in Table VI. We can conclude from the table that our proposed model can generate more fluent and accurate text summaries for long videos. The proposed temporal Transformer and Context Aggregation can help generate accurate and consistent video summaries. Following [87], we compute the Kappa coefficient of different workers, and the value is $0.49 \in (0.41, 0.60)$, which means that the consistency is moderate.

Comparison between CLIPScore and VT-CLIPScore: Although we can apply a pretrained CLIP model without any adaptation on our dataset to evaluate the semantic consistency of the video and text summaries, the similarity score may be

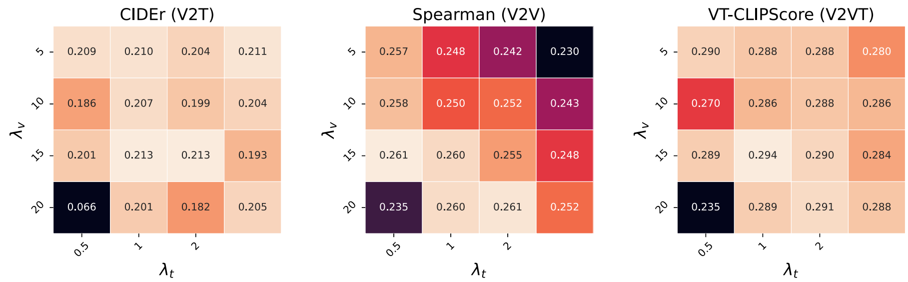

Fig. 6. Impact of multi-task weights λ_v and λ_t.

TABLE VII
RESULTS OF CROSS-MODALITY SIMILARITY UNDER DIFFERENT SEMANTIC CHANGES

Method	**Cross-Modal Similarity (Cosine Similarity)**	
	Positive pairs	Negative pairs
CLIPScore [17]	14.5	0.3
VT-CLIPScore	38.0	0.2

insensitive to the semantic change of generated cross-modal summaries. In Table VII, we compare the vanilla CLIPScore [17] and the finetuned VT-CLIPScore to measure the similarity of different video and text summarization pairs. The positive pairs refer to the paired video and text summaries. The negative pair includes unpaired video and text summaries. From the results, we can observe that CLIPScore provides a solid foundation for the finetuned VT-CLIPScore. Moreover, adapting the vanilla CLIP model to our proposed task is necessary since there is a domain gap between the CLIP pretraining data and our VideoXum data. The finetuned CLIP model on our data makes the evaluation score more reliable. Table VII shows that finetuning on VideoXum makes the similarity scores more reflective or informative in measuring the semantic consistency of cross-modal summaries.

F. Extended Discussion

Complexity analysis: In terms of computational complexity, the V2V-Sum, V2T-Sum, and V2VT-Sum models require 0.06, 1.15, and 1.18 GPU hours for training on an A100 GPU, respectively. Regarding the model complexity, they have parameter sizes of 435.6 MB, 564.8 MB, and 567.1 MB.

Impact of multi-task weights in (8): To determine the impact of multi-task weights (i.e., λ_v and λ_t), we perform an ablation study on the λ_v and λ_t using the VideoXum *val* set. Fig. 6 shows that the peak point appears when $\lambda_v = 15.0$ and $\lambda_t = 1.0$ as mentioned in Section V-B.

Impact of Local Window Size ε: For the V2V-SUM task, the local window size ε controls the context range of local self-attention. For $\varepsilon = 1$, the local self-attention module degrades to a multilayer perceptron (MLP). As ε increases to $T\,(> 1)$, the local self-attention module upgrades to a global/regular self-attention module. Fig. 5 shows the optimal performance occurs at $\varepsilon = 5$; below this, limited context hinders performance, while above it, excess context introduces irrelevant frames, reducing efficacy. Therefore, the performance of the V2V-SUM task is improved by carefully selecting appropriate local context information.

Impact of Temporal Transformer Layers: We conduct an ablation study on the number of Temporal Transformer layers N_{tem} using VideoXum *val* set. Table VIII indicates that altering N_{tem} does not significantly affect the performance for all three tasks. Therefore, we set N_{tem} to 1.

Analysis for Human Performance on V2T-SUM: Human performance on our proposed three tasks can be regarded as an upper bound of each task. Table III presents human performance outperforms our proposed model by a large margin on V2V-SUM and V2VT-SUM. However, on V2T-SUM, human performance does not exhibit a significant advantage over our model (especially on BLEU@4). To better understand this phenomenon, we examine human-annotated captions and their corresponding references, where "Human" indicates the human predictions on VideoXum *test* set and "Reference" denotes the corresponding ground truth. Both "Human" and "Reference" are human-annotated text summaries from ActivityNet Captions [8] validation sets. In particular, we present a representative example below:

```
 ================================
 *Human*: Two children stand in front of
a mat. They throw something onto the mat.
They take turns jumping across the mat.
They pick up the item they threw on it.
 ================================
 *Reference*: Two young children are
standing in line indoors in what looks
like a living room. The little girl is
standing closest to the hopscotch mat and
she throws her toy onto the mat and then
begins jumping until she meets the end of
the mat then turns around and heads back
to the point she started and her turn is
over. The little boy goes next, and he
throws the toy onto the mat and begins
jumping to the end of the mat, then turns
around and jumps back towards his start-
ing point.The little girl steps in front
of the boy and gets into motion to start
another turn on the hopscotch mat.
 ================================
```

Both captions describe two children playfully interacting on a mat, but "Reference" provides a more vivid and detailed picture

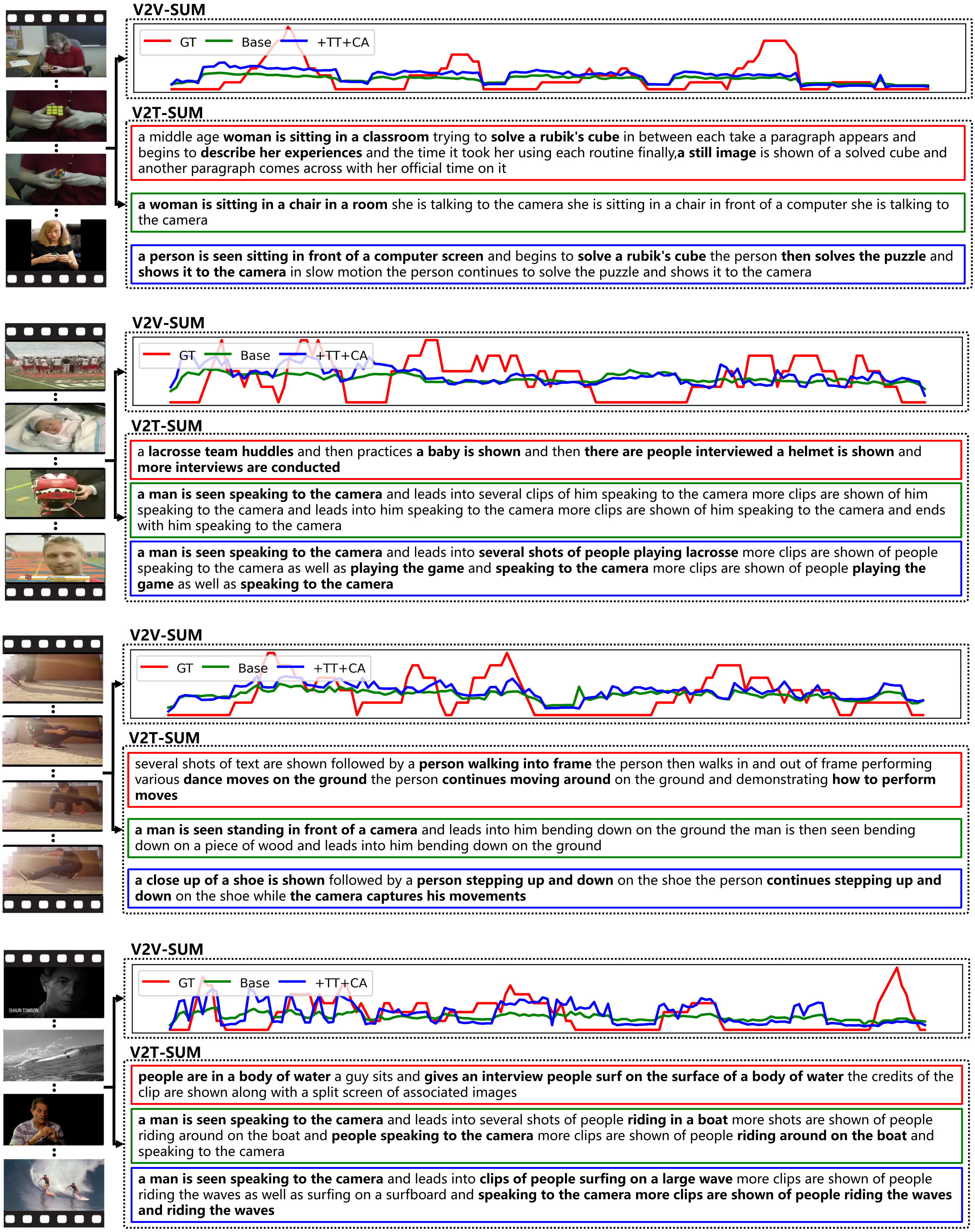

Fig. 7. Additional example results of the generated video and text summaries across different baseline models. Red (both line and box) indicates the results of the ground truth. Green indicates the results of the VTSUM-BLIP (Base). Blue indicates the results of VTSUM-BLIP (+TT+CA).

TABLE VIII
IMPACT OF TEMPORAL TRANSFORMER (TT) LAYERS N_{TEM} ON THE VIDEOXUM DATASET *VAL* SET. F1 SCORE, BLEU@4, METEOR, ROUGE-L, CIDEr, AND VT-CLIPSCORE ARE SHOWN IN %

Method	V2V-SUM			V2T-SUM				V2VT-SUM
	F1 score	Kendall	Spearman	BLEU@4	METEOR	ROUGE-L	CIDEr	VT-CLIPScore
VTSUM-BLIP (Base)	21.9	0.157	0.208	5.2	11.3	24.3	18.3	28.4
+ 1-layer TT	22.6	0.176	0.232	5.2	11.5	24.3	20.2	28.9
+ 2-layer TT	22.7	0.176	0.232	5.2	11.4	24.3	20.2	28.9
+ 3-layer TT	22.7	0.174	0.230	5.1	11.4	24.3	20.5	29.0

of the scene. The comparison shows that summarizing a long video is inherently subjective, leading to varying text descriptions of the same content among different individuals. Therefore, it explains why human performance does not exhibit a significant advantage over VTSUM-BLIP model.

VI. CONCLUSION

In this study, we first propose a new video and text summarization task along with an enriched dataset VideoXum. In this task, we jointly consider the generic video summarization task and video-to-text summarization task. Furthermore, we propose a strong baseline framework VTSUM-BLIP to address the challenges for our proposed task. The empirical results show that our proposed framework achieves promising performance on VideoXum. In addition, we adapt CLIPScore on the VideoXum benchmark and introduce a new metric VT-CLIPScore to evaluate cross-modal video summarization semantic consistency, which shows high consistency with human evaluation on multiple experimental results. For future studies, there are several promising directions on this benchmark. There is plenty of room to explore the strategy of associating V2V and V2T summarization tasks for better performance and efficiency. The proposed VideoXum dataset provides a foundation that could be significantly expanded through GPT-4 [88], for generating video instruction-following data and thereby promoting the development of a general-purpose video assistant. For the evaluation metric, the adapted CLIP model for measuring video-text similarity is a practical compromise for the lack of large video-text pretrained models. It also suggests the need for a more reliable metric for video-text coherence measurement. In addition, more advanced visual/video encoders and large language models (LLMs) [88], [89] could be integrated into the proposed framework to benefit the results of cross-model summarization.

REFERENCES

[1] G. Irie, T. Satou, A. Kojima, T. Yamasaki, and K. Aizawa, “Automatic trailer generation,” in *Proc. ACM Int. Conf. Multimedia*, 2010, pp. 839–842.

[2] I. Grabska-Gradzińska, L. Nowak, W. Palacz, and E. Grabska, “Application of graphs for story generation in video games,” in *Proc. Australas. Comput. Sci. Week Multiconf.*, 2021, pp. 1–6.

[3] Y. Song, J. Vallmitjana, A. Stent, and A. Jaimes, “TVSum: Summarizing web videos using titles,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 5179–5187.

[4] M. Gygli, H. Grabner, H. Riemenschneider, and L. Van Gool, “Creating summaries from user videos,” in *Proc. 13th Eur. Conf. Comput. Vis.*, 2014, pp. 505–520.

[5] S. E. F. De Avila, A. P. B. Lopes, A. da Luz Jr., and A. de Albuquerque Araújo, “VSUMM: A mechanism designed to produce static video summaries and a novel evaluation method,” *Pattern Recognit. Lett.*, vol. 32, pp. 56–68, 2011.

[6] L. Zhou, C. Xu, and J. Corso, “Towards automatic learning of procedures from web instructional videos,” in *Proc. AAAI Conf. Artif. Intell.*, 2018, Art. no. 930.

[7] J. Xu, T. Mei, T. Yao, and Y. Rui, “MSR-VTT: A large video description dataset for bridging video and language,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 5288–5296.

[8] R. Krishna, K. Hata, F. Ren, L. Fei-Fei, and J. Carlos Niebles, “Dense-captioning events in videos,” in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 706–715.

[9] B.-C. Chen, Y.-Y. Chen, and F. Chen, “Video to text summary: Joint video summarization and captioning with recurrent neural networks,” in *Proc. Brit. Mach. Vis. Conf.*, 2017.

[10] J. Li, D. Li, C. Xiong, and S. Hoi, “BLIP: Bootstrapping language-image pre-training for unified vision-language understanding and generation,” in *Proc. Int. Conf. Mach. Learn.*, 2022, pp. 12888–12900.

[11] B. Zoph et al., “Rethinking pre-training and self-training,” in *Proc. Adv. Neural Inf. Process. Syst.*, 2020, pp. 3833–3845.

[12] L. Li et al., “HERO: Hierarchical encoder for video language omni-representation pre-training,” in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2020, pp. 2046–2065.

[13] I. Beltagy, M. E. Peters, and A. Cohan, “Longformer: The long-document transformer,” 2020, *arXiv:2004.05150*.

[14] C. Ju, T. Han, K. Zheng, Y. Zhang, and W. Xie, “Prompting visual-language models for efficient video understanding,” in *Proc. Eur. Conf. Comput. Vis.*, 2022, pp. 105–124.

[15] B. Ni et al., “Expanding language-image pretrained models for general video recognition,” in *Proc. Eur. Conf. Comput. Vis.*, 2022, pp. 1–18.

[16] M. Narasimhan, A. Rohrbach, and T. Darrell, “Clip-it! Language-guided video summarization,” in *Proc. Adv. Neural Inf. Process. Syst.*, 2021, pp. 13988–14000.

[17] J. Hessel, A. Holtzman, M. Forbes, R. Le Bras, and Y. Choi, “CLIPScore: A reference-free evaluation metric for image captioning,” in *Proc. Conf. Empirical Methods Nat. Lang. Process.*, 2021, pp. 7514–7528.

[18] C. Wu et al., “GODIVA: Generating open-domain videos from natural descriptions,” 2021, *arXiv:2104.14806*.

[19] K. Zhang, W.-L. Chao, F. Sha, and K. Grauman, “Video summarization with long short-term memory,” in *Proc. 14th Eur. Conf. Comput. Vis.*, 2016, pp. 766–782.

[20] K. Zhou, Y. Qiao, and T. Xiang, “Deep reinforcement learning for unsupervised video summarization with diversity-representativeness reward,” in *Proc. AAAI Conf. Artif. Intell.*, 2018, Art. no. 929.

[21] J. Park, J. Lee, I.-J. Kim, and K. Sohn, “SumGraph: Video summarization via recursive graph modeling,” in *Proc. 16th Eur. Conf. Comput. Vis.*, 2020, pp. 647–663.

[22] Y. Saquil, D. Chen, Y. He, C. Li, and Y.-L. Yang, “Multiple pairwise ranking networks for personalized video summarization,” in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2021, pp. 1718–1727.

[23] A. Graves and A. Graves, “Long short-term memory,” in *Supervised Sequence Labelling With Recurrent Neural Networks*, Berlin, Germany: Springer, 2012, pp. 37–45.

[24] B. Zhao, X. Li, and X. Lu, “Hierarchical recurrent neural network for video summarization,” in *Proc. ACM Int. Conf. Multimedia*, 2017, pp. 863–871.

[25] B. Zhao, X. Li, and X. Lu, “HSA-RNN: Hierarchical structure-adaptive RNN for video summarization,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 7405–7414.

[26] Z. Ji, F. Jiao, Y. Pang, and L. Shao, “Deep attentive and semantic preserving video summarization,” *Neurocomputing*, vol. 405, pp. 200–207, 2020.

[27] S. Lal, S. Duggal, and I. Sreedevi, "Online video summarization: Predicting future to better summarize present," in *Proc. IEEE Winter Conf. Appl. Comput. Vis.*, 2019, pp. 471–480.

[28] Y. Yuan, H. Li, and Q. Wang, "Spatiotemporal modeling for video summarization using convolutional recurrent neural network," *IEEE Access*, vol. 7, pp. 64676–64685, 2019.

[29] T.-J. Fu, S.-H. Tai, and H.-T. Chen, "Attentive and adversarial learning for video summarization," in *Proc. IEEE Winter Conf. Appl. Comput. Vis.*, 2019, pp. 1579–1587.

[30] Y. Zhang, M. Kampffmeyer, X. Zhao, and M. Tan, "DTR-GAN: Dilated temporal relational adversarial network for generic video summarization," in *Proc. ACM Turing Celebrations Conf. - China*, 2019, pp. 1–6.

[31] D. Chen and W. B. Dolan, "Collecting highly parallel data for paraphrase evaluation," in *Proc. 49th Annu. Meetings Assoc. Comput. Linguistics*, 2011, pp. 190–200.

[32] L. Yao et al., "Describing videos by exploiting temporal structure," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 4507–4515.

[33] C. Yan et al., "STAT: Spatial-temporal attention mechanism for video captioning," *IEEE Trans. Multimedia*, vol. 22, no. 1, pp. 229–241, Jan. 2020.

[34] J. Wang, W. Jiang, L. Ma, W. Liu, and Y. Xu, "Bidirectional attentive fusion with context gating for dense video captioning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 7190–7198.

[35] L. Zhou, Y. Zhou, J. J. Corso, R. Socher, and C. Xiong, "End-to-end dense video captioning with masked transformer," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 8739–8748.

[36] T. Brown et al., "Language models are few-shot learners," in *Proc. Adv. Neural Inf. Process. Syst.*, 2020, pp. 1877–1901.

[37] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proc. Conf. North Amer. Chap. Assoc. Comput. Linguist.*, 2019, pp. 4171–4186.

[38] M. Lewis et al., "BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension," in *Proc. 58th Annu. Meet. Assoc. Comput. Linguist.*, 2020, pp. 7871–7880.

[39] C. Raffel et al., "Exploring the limits of transfer learning with a unified text-to-text transformer," *J. Mach. Learn. Res.*, vol. 21, pp. 5485–5551, 2020.

[40] W. Kim, B. Son, and I. Kim, "ViLT: Vision-and-language transformer without convolution or region supervision," in *Proc. Int. Conf. Mach. Learn.*, 2021, pp. 5583–5594.

[41] W. Wang et al., "MINILM: Deep self-attention distillation for task-agnostic compression of pre-trained transformers," in *Proc. Adv. Neural Inf. Process. Syst.*, 2020, pp. 5776–5788.

[42] H. Xue et al., "Probing inter-modality: Visual parsing with self-attention for vision-language pre-training," in *Proc. Adv. Neural Inf. Process. Syst.*, 2021, pp. 4514–4528.

[43] P. Zhang et al., "VinVL: Revisiting visual representations in vision-language models," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 5579–5588.

[44] A. F. Biten et al., "Scene text visual question answering," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 4291–4301.

[45] T.-Y. Lin et al., "Microsoft COCO: Common objects in context," in *Proc. 13th Eur. Conf. Comput. Vis.*, 2014, pp. 740–755.

[46] M. Regneri et al., "Grounding action descriptions in videos," *Trans. Assoc. Comput. Linguist.*, vol. 1, pp. 25–36, 2013.

[47] A. Singh et al., "Towards VQA models that can read," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 8317–8326.

[48] A. Dosovitskiy et al., "An image is worth 16x16 words: Transformers for image recognition at scale," in *Proc. Int. Conf. Learn. Representations*, 2021.

[49] J.-B. Alayrac et al., "Flamingo: A visual language model for few-shot learning," in *Proc. Adv. Neural Inf. Process. Syst.*, 2022, pp. 23716–23736.

[50] J. Wang et al., "GIT: A generative image-to-text transformer for vision and language," *Trans. Mach. Learn. Res.*, vol. 2022, 2022.

[51] J. Yu et al., "Coca: Contrastive captioners are image-text foundation models," *Trans. Mach. Learn. Res.*, vol. 2022, 2022.

[52] L. Yuan et al., "Florence: A new foundation model for computer vision," 2021, *arXiv:2111.11432*.

[53] L. H. Li, M. Yatskar, D. Yin, C.-J. Hsieh, and K.-W. Chang, "VisualBERT: A simple and performant baseline for vision and language," 2019, *arXiv:1908.03557*.

[54] J. Lu, D. Batra, D. Parikh, and S. Lee, "ViLBERT: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2019, pp. 13–23.

[55] W. Su et al., "VL-BERT: Pre-training of generic visual-linguistic representations," in *Proc. Int. Conf. Learn. Representations*, 2020.

[56] H. Tan and M. Bansal, "LXMERT: Learning cross-modality encoder representations from transformers," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2019, pp. 5099–5110.

[57] A. Radford et al., "Learning transferable visual models from natural language supervision," in *Proc. Int. Conf. Mach. Learn.*, 2021, pp. 8748–8763.

[58] H. Bao et al., "VLMo: Unified vision-language pre-training with mixture-of-modality-experts," in*Proc. Adv. Neural Inf. Process. Syst.*, 2022, pp. 32897–32912.

[59] Y. Du, Z. Liu, J. Li, and W. X. Zhao, "A survey of vision-language pre-trained models," in *Proc. Int. Joint Conf. Artif. Intell.*, 2022, pp. 5436–5443.

[60] A. Singh et al., "FLAVA: A foundational language and vision alignment model," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 15617–15629.

[61] C. Sun, A. Myers, C. Vondrick, K. Murphy, and C. Schmid, "VideoBERT: A joint model for video and language representation learning," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 7464–7473.

[62] H. Luo et al., "UniVL: A unified video and language pre-training model for multimodal understanding and generation," 2020, *arXiv:2002.06353*.

[63] J. Lei, T. L. Berg, and M. Bansal, "Detecting moments and highlights in videos via natural language queries," in *Proc. Adv. Neural Inf. Process. Syst.*, 2021, pp. 11846–11858.

[64] M. Sun, A. Farhadi, and S. Seitz, "Ranking domain-specific highlights by analyzing edited videos," in *Proc. 13th Eur. Conf. Comput. Vis.*, 2014, pp. 787–802.

[65] M. Otani, Y. Nakashima, E. Rahtu, and J. Heikkila, "Rethinking the evaluation of video summaries," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 7588–7596.

[66] Q. Ruan, M. Ostendorff, and G. Rehm, "HiStruct: Improving extractive text summarization with hierarchical structure information," in *Proc. Findings Assoc. Comput. Linguistics*, 2022, pp. 1292–1308.

[67] Y.-C. Chen et al., "UNITER: Universal image-text representation learning," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 104–120.

[68] V. Gabeur, C. Sun, K. Alahari, and C. Schmid, "Multi-modal transformer for video retrieval," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 214–229.

[69] M. Patrick et al., "Support-set bottlenecks for video-text representation learning," in *Proc. Int. Conf. Learn. Representations*, 2021.

[70] B. Zhang, H. Hu, and F. Sha, "Cross-modal and hierarchical modeling of video and text," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 385–401.

[71] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," in *Proc. Int. Conf. Learn. Representations*, 2019.

[72] I. Loshchilov and F. Hutter, "SGDR: Stochastic gradient descent with warm restarts," in *Proc. Int. Conf. Learn. Representations*, 2017.

[73] M. G. Kendall, "The treatment of ties in ranking problems," *Biometrika*, vol. 33, pp. 239–251, 1945.

[74] D. Zwillinger and S. Kokoska, *CRC Standard Probability and Statistics Tables and Formulae*. Boca Raton, FL, USA: CRC Press, 1999.

[75] H. Xu et al., "mPLUG-2: A modularized multi-modal foundation model across text, image and video," 2023, *arXiv:2302.00402*.

[76] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "BLEU: A method for automatic evaluation of machine translation," in *Proc. 40th Annu. Meet. Assoc. Comput. Linguist.*, 2002, pp. 311–318.

[77] S. Banerjee and A. Lavie, "Meteor: An automatic metric for MT evaluation with improved correlation with human judgments," in *Proc. Assoc. Comput. Linguistics Workshop*, 2005, pp. 65–72.

[78] C.-Y. Lin, "ROUGE: A package for automatic evaluation of summaries," in *Proc. Text Summarization Branches Out*, 2004, pp. 74–81.

[79] R. Vedantam, C. Lawrence Zitnick, and D. Parikh, "CIDEr: Consensus-based image description evaluation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 4566–4575.

[80] U. Singer et al., "Make-a-video: Text-to-video generation without text-video data," in *Proc. Int. Conf. Learn. Representations*, 2023.

[81] T. Zhang, V. Kishore, F. Wu, K. Q. Weinberger, and Y. Artzi, "BERTScore: Evaluating text generation with BERT," in *Proc. Int. Conf. Learn. Representations*, 2020.

[82] N. Reimers and I. Gurevych, "Sentence-BERT: Sentence embeddings using siamese BERT-networks," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2019, pp. 3982–3992.

[83] Y. Li, T. Yao, Y. Pan, H. Chao, and T. Mei, "Jointly localizing and describing events for dense video captioning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 7492–7500.

[84] S. Arora, R. Ge, B. Neyshabur, and Y. Zhang, "Stronger generalization bounds for deep nets via a compression approach," in *Proc. Int. Conf. Mach. Learn.*, 2018, pp. 254–263.

[85] H. Hua, X. Li, D. Dou, C. Xu, and J. Luo, “Noise stability regularization for improving BERT fine-tuning,” in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics*, 2021, pp. 3229–3241.

[86] H. Hua, X. Li, D. Dou, C.-Z. Xu, and J. Luo, “Fine-tuning pre-trained language models with noise stability regularization,” 2022, *arXiv:2206.05658*.

[87] K. Wang, H. Hua, and X. Wan, “Controllable unsupervised text attribute transfer via editing entangled latent representation,” in *Proc. Adv. Neural Inf. Process. Syst.*, 2019, pp. 11034–11044.

[88] OpenAI, “GPT-4 technical report,” 2023, *arXiv:2303.08774*.

[89] H. Touvron et al., “Llama 2: Open foundation and fine-tuned chat models,” 2023, *arXiv:2307.09288*.

Jingyang Lin (Graduate Student Member, IEEE) received the B.E. and M.Sc. degrees from Sun Yat-sen University, Guangzhou, China, in 2019 and 2022, respectively. He is currently working toward the Ph.D. degree in computer science with the University of Rochester, Rochester, NY, USA. His research interests include vision-and-language, video analysis, multi-task learning, and self-supervised learning.

Hang Hua received the bachelor’s degree from the South China University of Technology, Guangzhou, China, and the master’s degree from Peking University, Beijing, China. He is currently working toward the Ph.D. degree in computer science with the University of Rochester, Rochester, NY, USA. He is a Member of the VIStA group advised by Professor Jiebo Luo. His research interests include vision and language, machine learning, and social media analysis.

Ming Chen received the Ph.D. degree in mechanical and aerospace engineering from the University of California Davis, Davis, CA, USA, in 2018. She is currently a Senior Staff Engineer with OPPO US Research Center, Palo Alto, CA, USA . After that she joined OPPO US Research Center, Palo Alto, USA. Her research interests include video understanding, video summarization/highlight detection, and visual semantic representation learning.

Yikang Li received the Ph.D. degree in electrical engineering from Arizona State University, Tempe, AZ, USA, in 2020. During his Ph.D., he investigated different video semantic representations via deep learning techniques. He is currently a Senior Research Engineer with the OPPO US Research Center, Palo Alto, CA, USA. His research interests include video understanding, multi-modal foundational model learning, and visual semantic representation learning. .

Jenhao Hsiao received the Ph.D. degree in electrical engineering and computer science from National Taiwan University, Taipei, Taiwan. He is currently a principle research Scientist with OPPO US Research Center, Palo Alto, CA, USA. Before joining OPPO, he held research positions with Yahoo! Research and IBM Research. His main research interests include machine learning, deep learning, computer vision, vision-language model (VLM), and other related problems in AI.

Chiuman Ho received the Ph.D. degree in theoretical physics from the University of Pittsburgh, Pittsburgh, PA, USA, and the postdoc degree with UC Berkeley, Berkeley, CA, USA . He is currently the Senior Director of AI with OPPO US Research Center, Palo Alto, CA, USA. Before leaving the academia, he was a Research Assistant Professor with Michigan State University, East Lansing, MI, USA. Prior to joining OPPO, he was a Senior Staff Data Scientist with Huawei US R&D. research interests include AI research and its commercialization. He also helps to develop AI strategies for OPPO, and leads a team to apply AI to improve user experience.

Jiebo Luo (Fellow, IEEE) is currently the Albert Arendt Hopeman Professor of engineering and Professor of computer science with the University of Rochester, Rochester, NY, USA, which he joined in 2011 after a prolific career of fifteen years with Kodak Research Laboratories. He has authored nearly 600 technical papers and holds more than 90 U.S. patents. His research interests include computer vision, NLP, machine learning, data mining, computational social science, and digital health. He has been involved in numerous technical conferences, including serving as program co-chair of ACM Multimedia 2010, IEEE CVPR 2012, ACM ICMR 2016, and IEEE ICIP 2017, as well as general co-chair of ACM Multimedia 2018 and IEEE ICME 2024. He was on the Editorial boards of IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, IEEE TRANSACTIONS ON MULTIMEDIA, IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY, IEEE TRANSACTIONS ON BIG DATA, *ACM Transactions on Intelligent Systems and Technology*, *Pattern Recognition*, *Knowledge and Information Systems*, *Machine Vision and Applications*, and *Intelligent Medicine*. He was the Editor-in-Chief of IEEE TRANSACTIONS ON MULTIMEDIA during 2020–2022. Professor Luo is also a Fellow of NAI, ACM, AAAI, SPIE, and IAPR.