Journal of Innovative Image Processing (ISSN: 2582-4252)
www.irojournals.com/iroiip/

Improving Narrative Coherence in Dense Video Captioning through Transformer and Large Language Models

Dvijesh Bhatt[1], Priyank Thakkar[2]

Department of Computer Science and Engineering, Institute of Technology, Nirma University, Ahmedabad, India.

E-mail: [1]dvijeshbhatt@gmail.com, [1]17ptphde174@nirmauni.ac.in, [2]priyank.thakkar@nirmauni.ac.in

Abstract

Dense video captioning aims to identify events within a video and generate natural language descriptions for each event. Most existing approaches adhere to a two-stage framework consisting of an event proposal module and a caption generation module. Previous methodologies have predominantly employed convolutional neural networks and sequential models to describe individual events in isolation. However, these methods limit the influence of neighboring events when generating captions for a specific segment, often resulting in descriptions that lack coherence with the broader storyline of the video. To address this limitation, we propose a captioning module that leverages both Transformer architecture and a Large Language Model (LLM). A convolutional and LSTM-based proposal module is used to detect and localize events within the video. An encoder-decoder-based Transformer model generates an initial caption for each proposed event. Additionally, we introduce a Large Language Model (LLM) that takes the set of individually generated event captions as input and produces a coherent, multi-sentence summary. This summary captures cross-event dependencies and provides a contextually unified and narratively rich description of the entire video. Extensive experiments on the ActivityNet dataset demonstrate that the proposed model, Transformer-LLM based Dense Video Captioning (TL-DVC), achieves a 9.22% improvement over state-of-the-art models, increasing the Meteor score from 11.28 to 12.32.

DOI: https://doi.org/10.36548/jiip.2025.2.005
Received: 15.04.2025, received in revised form: 13.05.2025, accepted: 24.05.2025, published: 03.06.2025

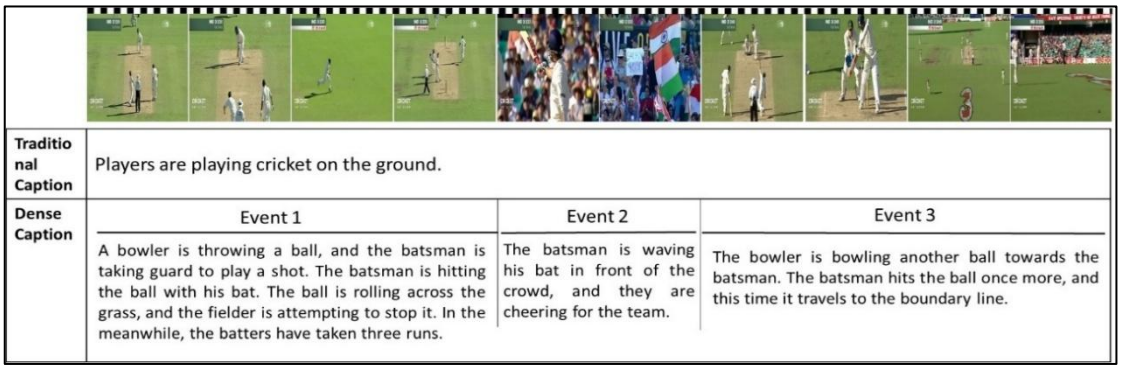
Keywords: Convolution 3D, Dense Video Caption, LSTM, Transformer, Encoder-Decoder, LLM.

1. Introduction

In today's world of social media, videos have emerged as a valuable medium for sharing information. However, extracting valuable insights from raw video continues to be a labor-intensive and time-consuming process. This challenge becomes even more significant in applications such as surveillance log generation and query-based video retrieval systems. Video summarization aims to address these issues. There are two primary categories of video summaries: visual summaries and textual summaries. Visual summaries select keyframes or clips, whereas textual summaries are designed to produce coherent natural-language descriptions of the visual content, a process referred to as video captioning. Traditional video captioning models generally generate one or more sentences to explain the content of a video. Nevertheless, overlapping events in real-world videos present a challenge for conventional models to identify and describe these events accurately.

Dense video captioning (DVC) addresses these limitations of conventional models. DVC is capable of precisely identifying and locating events. It distinguishes between simultaneous and consecutive events, making it useful in complex video scenarios. A captioning module receives the identified events and generates detailed descriptions of each one.

Figure 1 illustrates the difference between traditional and dense video captioning. As illustrated, traditional captioning models provide general video descriptions, whereas DVC identifies and locates all events in a video and generates detailed captions for each event. By automating event localization and description, DVC becomes useful for various applications, including video summarization [1,2,3], video retrieval [4], video segment localization based on queries [5,6], visual assistance for the visually impaired [7], intelligent visual-based FAQ chatbots [8] and instructional video generation [9].



Traditional Caption	Players are playing cricket on the ground.		
Dense Caption	Event 1	Event 2	Event 3
	A bowler is throwing a ball, and the batsman is taking guard to play a shot. The batsman is hitting the ball with his bat. The ball is rolling across the grass, and the fielder is attempting to stop it. In the meanwhile, the batters have taken three runs.	The batsman is waving his bat in front of the crowd, and they are cheering for the team.	The bowler is bowling another ball towards the batsman. The batsman hits the ball once more, and this time it travels to the boundary line.

Figure 1. This Image Illustrates the Differences Between Traditional and Dense Video Captioning

(Source of Video: https://www.youtube.com/watch?v=uOA25BRgSic&t=234s")

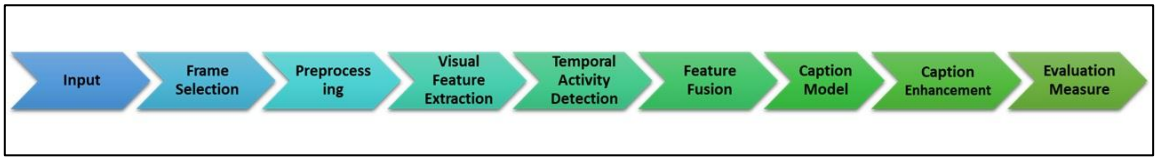


Figure 2. The Block Diagram of Dense Video Captioning Model

The entire DVC process is illustrated in Figure 2. The overall Dense Video Captioning (DVC) workflow begins with selecting frames from a video, either at regular intervals or through learning-based methods like PickNet [10] or reinforcement learning [11]. After frame selection, the frames are pre-processed, which involves resizing and augmentation. Additionally, the collection of videos is divided into training, validation, and test sets. Models such as C3D, Temporal CNN, or visual transformers are subsequently used to extract spatio-temporal features. These features feed into a temporal event (activity) proposal module, which divides the video into distinct event proposals. In the feature fusion stage, additional modalities like Mel Frequency Cepstral Coefficients (MFCC) from audio, C3D features from the original video, keyword cues, etc, can be merged with event proposal features. The enriched features are then fed into the captioning module, typically based on sequential models such as long-short-term memory (LSTM), gated recurrent units (GRU), or transformer models, to generate captions for the proposed event. Techniques such as reinforcement learning, semantic attention, generative adversarial networks (GANs), and large language models (LLMs) can be incorporated into the caption enhancement stage to improve the captions. Lastly, standard evaluation metrics, including BLEU, METEOR, ROUGE, SPICE, CIDEr, SODA, and WMD, are employed to compare the generated captions with human-written references, thereby guaranteeing that they are contextually accurate and meaningful during the evaluation measure stage.

The proposal module and captioning module form the core components of the DVC. Thus, the performance of the DVC model depends on the performance of both modules. One widely used technique for generating activity proposals is the sliding window method, initially introduced by Victor Escorcia et al. in their temporal action detection model [12]. This method entails the movement of a fixed-size window across the video timeline, with each segment categorized as either foreground (activity) or background. However, its performance is constrained by the fixed window size and the need to process the video multiple times. Shyamal Buch et al. introduced the Single Stream Temporal Action Proposals (SST) method to address the limitations of the sliding-window approach [13] This method scans the video in a single pass and generates potential event proposals at each time step t with k unique offsets [13]. Building on these proposals, the captioning module employs sequential models to generate coherent, context-aware descriptions for each event.

The C3D, TCN (Temporal Convolution Network), and visual transformer models (such as ViT) extract spatiotemporal features from videos. Many studies have experimental proof that suggests C3D and transformer-based solutions provide better accuracy with state-of-the-art datasets in the use-case of dense video captioning [14, 15, 16, 17, 18, 19, 20]. By utilizing information from past and future events, C3D with LSTM and transformer-based approaches improve event comprehension by offering a more precise interpretation of the current event. Transformer-based visual models are more accurate; however, their high computational requirements make them challenging to implement on handheld devices. As a result, C3D was chosen to extract the spatiotemporal features because of its ability to balance computational efficiency with temporal feature extraction capability. These visual features are forwarded into the captioning module to generate the description of each event. Researchers use RNNs or transformer-based sequence models for the captioning module to generate an event's description from visual features. Transformer-based architectures are well-suited for capturing long-range dependencies and have demonstrated superior performance over traditional RNN-based models in producing accurate and coherent captions. In conventional DVC frameworks, event proposals are generated independently, and their features are used to generate captions in isolation. As a result, the captioning module often lacks awareness of the broader video context, leading to descriptions that fail to reflect the overall narrative.

These challenges motivate us to design a practical solution capable of generating event descriptions that are accurate and contextually coherent while ensuring that the final video

captioning maintains the overall narrative of the video. The proposed solution contributes to the following key areas: (i) the captioning module utilizes an encoder-decoder Transformer to generate captions based on visual features, and (ii) a large language model (LLM) is employed to generate the final video-level caption that cohesively describes the entire video while preserving its narrative flow. This highlights that the primary objective of our research is to enhance the captioning module and the caption refinement component within the dense video captioning pipeline, as illustrated in Figure 2.

The remainder of this paper is structured as follows. Section 2 presents a comprehensive review of existing dense video captioning methodologies. Section 3 details the architecture and components of the proposed model. Section 4 discusses the experimental results, offering a comparative evaluation of the proposed model against leading state-of-the-art approaches. Section 5 constitutes the concluding segment of the analysis and delineates pertinent domains for prospective research.

2. Review of Existing Approaches in Dense Video Captioning (DVC)

The DVC is a procedure that identifies and locates temporal activity proposals within video clips, subsequently articulating each proposal in one or more sentences. This section will present a thorough review of the literature published on dense video captioning.

2.1 CNN-LSTM based Approach for DVC

In 2017, Ranjay Krishna et al. proposed the first dense video captioning model [21]. The temporal activity proposal module DAPs is fed C3D features extracted from video by the proposed model. The DAPs produce K proposal outputs at each time step, each containing activity. The captioning module receives each proposed event. Using an attention-based LSTM, the caption module extracts the event's concept by observing past and future event proposals to generate captions. The computational cost is increased by the use of DAPs, which employ the sliding window concept, which is the sole flaw in the proposed model.

Jingwen Wang et al. utilized two SST models to extract temporal event proposals in both forward and backward directions, thereby overcoming the constraint of DAPs [14]. The captioning module implements a two-layer LSTM model with context gates. Yehao Li et al. proposed a new DVC model in which the proposal module employs regression and

classification models [18]. The captioning module utilizes reinforcement learning as an optimization technique to generate an appropriate caption by receiving all proposals. Consequently, numerous techniques were devised for DVC, which include the utilization of R-C3D [22], the combination of SST and GRU [15], graph-based partitioning and summarization (GPaS) [23] and the event sequence generation network [24]. Maitreya Suin et al. proposed an effective framework for dense video captioning, as only a limited number of frames are necessary to depict an event accurately [11].

2.2 Transformer based Approach for DVC

The researchers have employed transformer models to enhance caption generation in the era of transformers. Luowei Zhou et al. proposed the first encoder-decoder transformer model explicitly designed for DVC [25]. Rather than employing the C3D model, they selected the visual transformer to extract visual features. The model uses two distinct decoders: the proposal decoder and the captioning decoder. The temporal event proposals are generated by the proposal decoder, which decodes the visual feature of each frame. On the other hand, the captioning decoder employs a masking network to convert the suggested events into distinguishable masks, which are then used to generate captions. Iashin et al. introduced a multi-modal transformer-based DVC model that employs I3D features to represent visual information, VGGish to extract audio features [26], GloVe embeddings for linguistic representation [27, 28] and a speech model [29]. Later, researchers used parallel processing to detect and describe the event for faster output [30, 31].

The ViSE (Visual-Semantic Embedding) framework, which maps visual features extracted via 2D-CNNs and encoded captions using weighted n-grams into a common semantic space, was subsequently proposed by Nayyer Aafaq et al. This method integrates linguistic expertise into the caption generation process [17]. Deng et al. describe the events and subsequently identify them in the video, which is the opposite of the process previously discussed [33]. Wanhyu Choi et al. identified two subtasks, boundary detection, and event count estimation, in order to improve the proposal module [35]. Antoine Yang et al. enhanced the localization of events and the generation of captions by incorporating time tokens and captions into the training process [36]. Yiwei Wei et al. proposed a multi-perspective perception (MPP) model that uses a hierarchical temporal-spatial summary and a multi-perspective attention layer to generate a dense video caption [37]. Moreover, Xingyi Zhou et

al. introduced Streaming DVC, a method designed to accommodate longer videos by incorporating a memory module based on a clustering technique. This method facilitates the efficient processing of extended video content [38]. Minkuk et al. employed cross-attention and the external knowledge memory block to produce meaningful captions [39]. In order to enhance the caption and optimize the boundaries of the proposed event, Hao Wu et al. employed a large language model [40].

Table 1. Literature Survey on Dense Video Captioning Deep Learning Models

Dense Video Captioning Model	Year	Multi Model	Visual Model	Localization Model	Captioning Model	Audio Features	Loss Function		
							Detection Loss	Localization Loss	Captioning Loss
Research Literature's Result									
DenseCap [21]	2017	-	Convolution 3D	DAPs	LSTM	-	Weighted Binary Cross Entropy	Recall with tIoU	Cross Entropy
Jointly Dense [18]	2018	-	Convolution 3D	TEP	LSTM	-	Soft max	Smooth L1	Reinforcement Learning
EndtoEnd [25]	2018	-	CNN with Self Attention	ProcNet without LSTM Gate	Transformer	-	Binary Cross Entropy	Smooth L1 with tIoU	Weighted Cross Entropy
Bidirectional [14]	2018	-	Convolution 3D	Bidirectional SST	LSTM with Context Gate	-	Weighted Binary Cross Entropy	tIoU	Sum of Negative Likelihood
JEDDi-Net [22]	2019	-	Convolution 3D	R-C3D	LSTM	-	Binary Cross Entropy	tIoU	Log Likelihood

Streamline [15]	2019	-	Convolution 3D	SST Pointer Net	LSTM with Context Gate	-	Weighted Binary Cross Entropy	tIoU	Reinforcement Learning
Watch Listen Tell [32]	2019	✓	Convolution 3D	-	GRU	MFCC, CQT, SoundNet	Binary Cross Entropy	L2	Cross Entropy
EfficientNet [11]	2020	-	CNN with Self Attention	ProcNet without LSTM Gate	Transformer	-	Binary Cross Entropy	Smooth L1 with tIoU	Weighted Cross Entropy
GPaS [23]	2020	-	Convolution Graph Network	GCN-LSTM	LSTM	-	-	-	Cross Entropy
Bidirectional Transformer [27]	2020	✓	Inflated 3D	Bimodal Multi Headed Attention	Bimodal Transformer	VGGish	Binary Cross Entropy	MSE	KL-divergence
AMT [28]	2021	✓	TCN Self Attention	Anchor-free Local Attention	Single Shot Masking	-	Cross Entropy	Regression Loss	Weighted Cross Entropy
SGR [33]	2021	✓	CNN back-bone with Transformer	Dual Path Cross Attention		-	-	Logistic Regression Loss	Cross Entropy
PDVC [30]	2021	✓	CNN with Transformer	Transformer with Localization Head	LSTM with Soft Attention	-	Binary Cross Entropy	Generalize IOU	Cross Entropy

PPVC [31]	2022	✓	Convolution 3D with Self Attention	Cross Attention	Multi-Stack Cross Attention	-	-	Regression Loss	Log Likelihood
VSJM-Net [34]	2022	✓	2D-CNN ViSE	Multi-headed Attention	FFN	-	Binary Cross Entropy	-	Cross Entropy
SBS [35]	2023	-	C3D + Transformer	Multi-headed Attention	LSTM	-	Binary Cross Entropy	Negative log likelihood	Cross Entropy
Vid2seq [36]	2023	-	Transformer	Multi-headed Attention	Transformer	-	-	-	Cross Entropy
MPP-net [37]	2023	-	CNN + Transformer	Self Attention	LSTM + Transformer	-	Cross Entropy	Generalize tIoU	Cross Entropy
Streaming DVC [38]	2024	-	Transformer	Self Attention	Transformer	-	-	-	-
CM-DVC [39]	2024	-	Transformer	Self Attention	Transformer	-	Cross Entropy	-	Cross Entropy
DIBS [40]	2024	-	Transformer	LLM	LLM	-	-	-	Cross Entropy

2.3 Weakly Supervised Model for DVC

All of the aforementioned DVC models have implemented the ActivityNet dataset, a benchmark that has been annotated for dense video captioning. However, early research in DVC also examined unsupervised and weakly supervised models. Captions, as well as the beginning and end times of events, are provided by ActivityNet and other supervised learning datasets for training. In contrast, weakly supervised learning methods do not require temporal segment annotations, which enables dense captioning on any video dataset without detailed labelling, These models are capable of identifying spatial regions of interest in videos and

detecting events, which leads to the generation of captions [41,20]. Audio, visual, and linguistic modalities were integrated into the initial attempt at multi-modal DVC using weakly supervised learning [32]. This method yielded more comprehensive and accurate caption generation from speech data by employing techniques such as Mel-Frequency Cepstral Coefficients (MFCC), Constant-Q Transform (CQT), and SoundNet [42] to extract audio features. Valter Estevam et al. presented an unsupervised model that acquires visual features without human annotations, thereby demonstrating that multi-modal data can significantly enhance model performance [43].

Table 1 offers a comprehensive review of the diverse dense video captioning models that have been proposed over the years. It compares models based on vital attributes such as the visual feature extraction model, event localization model, captioning model, and loss functions. The transition from LSTM-based architectures to Transformer-based models is illustrated in the table, demonstrating the evolution of the techniques used for dense video captioning. A comprehensive comparison is provided by the table, enabling researchers to understand the trends and advancements in the dense video captioning field.

As previously mentioned, the proposal and captioning modules are both equally important in the production of precise captions for dense video captioning. Various models generate temporal event proposals using methods such as the sliding window approach [21], Single Stream Temporal (SST) models [14, 15], regression techniques [18, 22], or transformer-based architectures [25, 17, 14, 27]. To accurately contextualize the current event, it is essential to have contextual information from both past and future events. In order to resolve this issue, certain models integrate context from adjacent events through the use of attention mechanisms [21] or Bi-directional SST [14, 15]. Despite these advancements, numerous models are proposed that prioritize the production of linguistically, contextually, and semantically accurate captions over the management of longer temporal dependencies. For example, the model in [14] integrates past and future proposals to generate contextually enhanced visual features for the current event. A standard LSTM with a gating mechanism is then employed to combine these features in order to produce semantically accurate sentences. However, it overlooks the importance of integrating contextual and linguistic knowledge at the caption level. In contrast, the model in [18] focuses on creating linguistically precise captions, but it neglects to account for previous and future event proposals, resulting in a lack of contextual information. This

serves as a strong motivation for us to propose a novel captioning module that generates accurate captions for individual events while preserving the overall narrative of the video.

The proposed model leverages both past and future event proposals to determine the visual features of the current event more accurately. These extracted visual features are then fed into the encoder of a Transformer-based model, where the decoder generates a caption for each event. Furthermore, to ensure that the overall narrative of the video is preserved throughout the captioning process, a Large Language Model (LLM) is employed to generate a video-level summary based on the event-level captions produced by the transformer.

3. Transformer and LLM based Captioning Module for DVC

This section introduces the TL-DVC dense video captioning model, which comprises two primary components: the proposal module and the captioning module. Contextual information from both past and future events is necessary to understand the current event in a video. The proposal module enhances the model's understanding of the current event by extracting features from surrounding events. For this purpose, the temporal activity proposal module from [14] was employed, together with an innovative captioning module in the proposed model. The block diagram of the proposed model (TL-DVC) for dense video captioning is depicted in Figure 3. The proposal module utilizes the Convolutional 3D (C3D) model to extract spatiotemporal features from the video. These extracted features are subsequently processed by forward and backward directional LSTMs inspired by SST [13] in order to incorporate sequential dependencies. This aids in the accurate identification and detection of the start and end timestamps of the proposed event. These LSTM features are classified as either background activity or event proposals. The event proposals that include the original C3D features of the event are forwarded for caption generation. The captioning module implements an encoder-decoder-based Transformer architecture to produce captions for each detected event. In addition, a Large Language Model (LLM) is employed to synthesize a video-level caption that integrates the individual event captions into a coherent summary, thereby ensuring narrative consistency throughout the video.

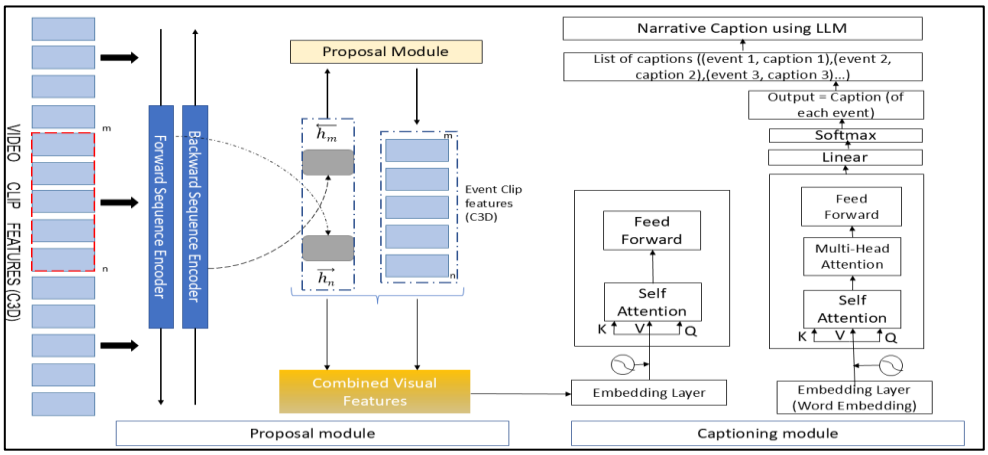


Figure 3. The Diagram Illustrates a General Architecture of the Proposed Model TL-DVC

3.1 Event Proposal Module

The event proposal module is designed to classify temporal segments within a video as either an event or a background activity. A video is a sequence of N frames, each of which is represented by the symbol $V = \{f_1, f_2, f_3, \ldots, f_N\}$. Utilizing a temporal window of 16 frames ($\delta = 16$), the Convolution 3D (C3D) model [44] extracts spatio-temporal features from the video. Time steps, denoted by $T = N/\delta$, are the distinct time segments that this method divides the video into. At each time step, the C3D visual feature are denoted as $V = \{v_1, v_2, v_3, \ldots., v_N\}$, where v_1 is the features vector of 16 frames from the video. The principal component analysis (PCA) algorithm is employed to reduce the C3D features to 500 dimensions.

To sequentially encode the extracted visual features (C3D features) in both directions, they are fed into a forward and backward-directional LSTM model. By utilizing contextual information from both preceding and subsequent events, this dual-directional encoding enables an in-depth understanding of the current event. The visual features are processed, and the forward and backward LSTM encoders accumulate information. The temporal dynamics of visual information across all time steps t are captured by the hidden state of the forward LSTM encoder, denoted as $\overrightarrow{h_t} \in \overrightarrow{h_{i=1}^{T}}$.

At each time step t, the concealed representation is partitioned into K event proposals according to pre-established K proposal anchors. The anchors are identified by applying K-means clustering to the ground-truth event durations included in the ActivityNet dataset's

training set. Each cluster center that was acquired serves as a temporal anchor and represents a prototypical event duration. To balance model precision and computational cost, we extract all annotated event durations from the training data and group them into K clusters using the K-means algorithm. K = 128 is selected based on the elbow method. The value of K can influence the performance of the proposal module.

Algorithm 1 Transformer and LLM based dense video captioning model

Input: X = $[(x_1, x_2, ..., x_N])$ ▷ N = Total Frames
Input: V = $[(v_1, v_2, ..., v_T])$ ▷ T = N/ δ
Input: E = Word embedding vector
Input: δ = 16
Output: R = $[((P_1, S_1), ..., (P_n, S_n))]$
v_i : C3D features of video V for δ frames ▷ i = 1,2,...T
P_i : n number of events (P) from video ▷ i = 1,2,...n
S_i : n number of captions (c) from video ▷ i = 1,2,...n
S_p : List of captions for (p) event in video
P_n^t : Generated n proposal at t time stamp
C_p^t : confidence score of n proposal at t time stamp
$\overleftarrow{h_m}$: hidden vector of selected event at m time stamp, m is start time
$\overrightarrow{h_n}$: hidden vector of selected event at n time stamp, n is end time
$F_{(p_i)}$: C3D features + hidden vectors of event
$\vec{v}^t$: C3D features of selected event
$WV_{(f_i)}$: word Vector at timestamp t
C_v: video level caption

function DENSEVIDEOCAPTIONING(X,δ,E)
  $V \leftarrow C3D\_Feature(X, \delta)$
  $F_{(p_i)} \leftarrow ProposalModule(V)$
  $S_p \leftarrow CaptionModule(F_{(p_i)}, E, FS_{(p_i)})$
  $C_v \leftarrow LLM(S_p)$
end function

function PROPOSALMODULE(V)
  $\overrightarrow{P}_n^t, \overrightarrow{C}_p^t, \overrightarrow{h}_n \leftarrow Forward\_SST(V)$
  $\overleftarrow{P}_n^t, \overleftarrow{C}_p^t, \overleftarrow{h}_m \leftarrow Backward\_SST(V)$
  $C_p \leftarrow \overrightarrow{C}_p^t \times \overleftarrow{C}_p^t$ ▷ Selection of Proposal
  $F_{(p_i)} \leftarrow [\vec{v}^t, \overrightarrow{h_n}, \overleftarrow{h_m}]$
return $F_{(p_i)}$
end function

function CAPTIONMODULE($F_{(p_i)}, E, FS_{(p_i)}$)
  $WV_{(f_i)} \leftarrow Transformer(F_{(p_i)}, E$
  $S_p \leftarrow Concat(Selected\_word)$ ▷ Caption for each event
return S_p
end function

Figure 4. Algorithm of TL-DVC

Every event proposal is denoted as $\vec{P}^t = \vec{p}_i^t$, where i = 1,2,...,K. Each proposal commences at t-l and concludes at t, with $l_{i\,i=1}^{K}$ representing the lengths of K proposals. All proposals have a common conclusion time t in a forward trajectory. Independent binary classifiers subsequently examine the concealed states of each proposal to produce confidence scores. Each classifier ascertains whether the encoded features correspond to an event or background, with the confidence score reflecting the classifier's certainty regarding the classification. We possess K proposals, each evaluated by independent classifiers, yielding a K confidence score. The confidence scores are derived from a fully connected neural network.

$$\overrightarrow{C_p}^t = \sigma\left(\overrightarrow{W_c}\,\overrightarrow{h_t} + b_c\right) \tag{1}$$

Equation 1 illustrates the application of the sigmoid activation function to determine the confidence score of the proposal. σ denotes the sigmoid nonlinearity. The weight matrix $\overrightarrow{W_c}$ is utilized to calculate the confidence score via the sigmoid function, whereas b_c represents the associated bias term. $\overrightarrow{C_p}^t$ denotes the confidence score vector for all proposals at time step t. The weight matrix ($\overrightarrow{W_c}$) and bias (b_c) are uniformly applied across all time steps. The resultant confidence score $\overrightarrow{C_p}^t = \overrightarrow{c_l}^t$, where $i = 1,2,\ldots,K$, denotes the probability of the K event proposals. In order to eliminate irrelevant event-caption pairs, we will later combine the confidence score of the proposal with the confidence score of the caption.

As with the forward-directional LSTM, the backward-directional LSTM encodes visual information across all time steps, commencing from the final frame T_n and proceeding in reverse order. It generates K proposals at each time step by reversing the video sequence, which is represented as $\overleftarrow{P}^t = \overleftarrow{p_l}^t{}_{i=1,2,\ldots K}$. Additionally, it generates K corresponding confidence scores, $\overleftarrow{C_p}^t = \overleftarrow{c_l}^t{}_{i=1,2,\ldots,K}$.

Several proposals, denoted as NP, are collectively generated by the forward and backward LSTMs. The remaining proposal scores from both directions are combined using a multiplication method, while proposals with low confidence scores (<50\%) are eliminated to improve these results. Finally, the equation that represents the cumulative confidence score C_p is as follows.

$$C_p = \{\overrightarrow{c_l} \times \overleftarrow{c_l}\}_{i=1}^{NP} \tag{2}$$

The total number of proposals generated by both the forward and backward LSTMs is denoted by NP in the above equation. Proposals that meet a cumulative confidence score threshold (0.8 tIoU) are forwarded to the captioning module for further processing. This guarantees that the captioning module is supplied with only the most relevant events for a comprehensive description.

In the context vectors, the hidden states of the proposal model are represented by $\overrightarrow{h_m}$ the forward-directional LSTM and $\overleftarrow{h_n}$ the backward-directional LSTM. The start and end time steps for the detected event proposal p_i are denoted by m and n. The context vectors are fused with the original C3D features of the event proposal p_i, rather than being directly passed to the captioning module. The following equation defines the visual feature vector F_t in the forward direction for the p_i, proposal:

$$F_t(p_i) = f\left(\overrightarrow{h_n}, \overleftarrow{h_m}, \widehat{V} = \{v_i\}_{i=m}^{n}, H_{t-1}\right) \quad (3)$$

The C3D features of the detected event are denoted by $\widehat{V}$ in equation 3. The C3D feature of an event that commences at the m time stamp and concludes at the n time stamp is denoted by $(v_i)_{i=m}^{n}$. $\overrightarrow{h_n}$ denotes the context vector in a forward direction, while $\overleftarrow{h_m}$ denotes the context vector in a backward direction. The captioning module is fed all of these features, as well as the previous hidden state of the LSTM (H_{t-1}). Temporal dynamic attention is employed to identify critical visual features (C3D) through the use of context vectors, thereby facilitating the identification of significant frames. The model's capacity to comprehend the event's context is enhanced by integrating context vectors with C3D features. The following equation is used to determine the relevance score z_i^t at each time step t:

$$z_i^t = W_a^T \cdot \tanh\left(W_v v_{i+m-1} + W_h\left[\overrightarrow{h_n}, \overleftarrow{h_m}\right] + W_H H_{t-1} + b\right) \quad (4)$$

A C3D feature of a specific time step is represented by v_{i+m-1} in equation~4. The value of i commences at zero. $\left[\overrightarrow{h_n}, \overleftarrow{h_m}\right]$ represents a vector concatenation, and H_{t-1} is a hidden representation of the previous timestamp. Weight matrices are denoted as W_a, W_v, W_h and W_H. The weight (α_i^t) of visual features (v_{i+m-1}) is determined using softmax normalization:

$$\alpha_i^t = \exp(z_i^t) \,/\, \sum_{k=1}^{n-m+1} \exp(z_k^t) \quad (5)$$

where n-m+1 represents the length of the event proposal. The weighted sum is used to calculate the attended visual feature ($\tilde{v}^t$)

$$\widetilde{v^t} = \sum_{i=1}^{n-m+1} \alpha_i^t \cdot v_{i+m-1} \quad (6)$$

The most appropriate original visual features that are in alignment with the context vectors ($\overrightarrow{h_n}, \overleftarrow{h_m}$) are determined by the relevance score (z_i^t). The event is comprehensively understood by aggregating visual features and context vectors, which are represented as $F(p_i)$. Consequently, the captioning module's encoder of the transformer receives the following final input:

$$F(p_i) = \left[\widetilde{v^t}, \overrightarrow{h_n}, \overleftarrow{h_m}\right] \quad (7)$$

The final feature vector in the forward direction is a combination of the context vectors ($\overrightarrow{h_n}, \overleftarrow{h_m}$) and the attended visual feature of the original video frame (C3D feature) in the aforementioned equation.

3.2 Caption Module with Encoder-Decoder based Transformer and LLM

The encoder-decoder-based Transformer model in the captioning module is designed to capture both temporal and contextual relationships between the visual feature vector $\widetilde{v^t}$ and the hidden representations $[\overrightarrow{h_n}, \overleftarrow{h_m}]$ obtained from the forward and backward proposal modules. The encoder's self-attention mechanism receives the embeddings of $\widetilde{v^t}$ and $[\overrightarrow{h_n}, \overleftarrow{h_m}]$and integrates positional encodings to generate a combined representation MV. This representation serves as the query (Q), key (K), and value (V) inputs to the self-attention module. The resulting output is passed through a feed-forward layer to produce the final encoded visual representation MV'. In the decoder, embedded caption tokens, along with their positional encodings (denoted as E), undergo self-attention to model dependencies among the generated words. These features are then passed through a cross-attention module, where the decoder attends to the encoder outputs (MV'), enabling interaction between visual and linguistic modalities. The output of this cross-attention mechanism is processed through feed-forward layers, followed by a fully connected layer and a softmax activation to produce the probability distribution over the vocabulary, thereby generating the final caption for each event.

The Transformer-based encoder-decoder model generates event-level captions that are received by the captioning evaluation module. Nevertheless, as previously mentioned, the majority of dense video captioning (DVC) pipelines treat each event independently, which frequently leads to captions that are inconsistent with the overall video narrative. Although these captions may accurately convey the contextual significance of individual events, they are unable to depict the interrelationships between events, particularly when they occur concurrently or in parallel. In order to overcome this constraint, we employ a large language model (LLM) to produce a unified video-level caption. This is accomplished by incorporating all event-level captions into the LLM, in addition to a meticulously crafted prompt that directs the model to generate a narrative-driven and coherent description of the entire video. The prompt for the same is: “You are given a list of individual event-level captions, each accompanied by its corresponding start and end timestamp. These events describe different segments of a video and may occur sequentially, simultaneously, or with partial overlap. Your task is to merge these captions into a single coherent paragraph that reads as a natural narrative. Use at least 80% of the original words from the input captions. Do not invent additional details beyond what is necessary for fluency. Use the timestamps to infer and reflect the temporal relationships between events: If events are sequential, present them in order. If events overlap, describe them as occurring simultaneously or in parallel. Maintain logical flow, avoid presenting the captions as a disjointed list, and ensure the final paragraph reads like a cohesive summary of the video.” This approach leads to a contextually rich, narrative-driven, video-level caption. Suppose the video generates three individual event-level captions: (1) “A person is cooking food in the kitchen” (Start: 00:00, End: 00:20) (2) “A child is playing with toys on the living room floor” (Start: 00:05, End: 00:25) and (3) “Someone is watching TV while seated on the couch” (Start: 00:15, End: 00:30). The video-level description generated by the large language model (LLM) is: “A person is cooking food in the kitchen while a child is playing with toys on the living room floor. At the same time, someone is watching TV while seated on the couch". In another example, (1) “A person enters the room and turns on the lights" (Start: 00:00, End: 00:10). (2) “The person sits down and opens the laptop” (Start: 00:11, End: 00:20) (3) “He is using laptop and looking at screen” (Start: 00:21, End: 00:30). The video-level description generated by the large language model (LLM) is: “A person enters the room and turns on the lights. Then, the person sits down and opens the laptop. He is using the laptop and looking at the screen.” This illustrates how the LLM effectively integrates parallel events into a coherent and semantically connected summary of the overall video content.

Previously, we discussed a confidence score to select events from video. A similar approach is implemented in the caption selection module, where confidence scores assist in the identification of the most appropriate caption for each event. The caption confidence score (C_c) has been calculated using an identical score. The final confidence score ($C = C_p + C_c$) was determined by combining the confidence scores of the proposal and captioning results. Subsequently, select the top n event and caption pairs during the inference phase.

3.3 Loss Function

The DVC process is divided into two primary modules: caption generation and proposal generation. The weighted multi-class categorical cross-entropy loss function, denoted as L_p, is employed in the proposal generation module. Utilizing temporal intersection over union (tIoU), this loss function is computed by comparing the generated proposal intervals to the ground truth values. The lengths of all ground truth proposals are collected and grouped into 128 clusters for loss calculation, which corresponds to the value K that was previously discussed. A ground truth label is assigned to each training sample, which is represented as $(y_t)_{t=1}^{T}$. A K-dimensional binary vector is used to represent each ground truth label y_t. If the corresponding proposal interval has a tIoU with the ground truth that is less than 0.5, the value of y_t is set to 0. Otherwise, it is set to 1. The formula is employed to determine the loss L_p at time t for video V with ground truth (y):

$$L_p(c, t, V, y) = -\sum_{j=1}^{K} w_0^i \, y_t^j \log c_t^j + w_1^j \left(1 - y_t^j\right) \log\left(1 - c_t^j\right) \quad (8)$$

The weights w_0 and w_1 in the aforementioned equation are assigned based on the frequencies of positive and negative samples, respectively. The prediction score associated with the j-th proposal at time step t is denoted by the variable c_t^j. The ground truth of the j-th proposal at timestamp t is denoted by ${y_t}^t$. Gradients are back-propagated through both directions to support concurrent training, and losses are computed for both forward and backward directions.

The captioning module receives proposals that satisfy the tIoU threshold (> 0.8). The negative log-likelihood of the correct words in a sentence containing M words is the sum of the captioning loss in the forward direction (L_{c_f}) and backward direction (L_{c_b}). This is denoted as the negative log-likelihood of the predicted words and is expressed as follows:

$$L_{c_f} = -\sum_{i=1}^{M} \log\left(p(w_i)\right) \tag{9}$$

$$L_{c_f} = -\sum_{i=1}^{M} \log\left(p(w_i)\right) \tag{10}$$

where w_i denotes the i-th word in the ground truth sentences from both directions. By combining the two losses, the total loss L is determined:

$$\mathrm{L} = \lambda\left(\mathrm{L_p} + \mathrm{L_c}\right) \tag{11}$$

In this equation, λ is a default value of 0.5 that balances the contributions of the proposal and captioning modules. L_p represents the proposal loss, while L_c represents the captioning loss.

4. Result and Implementation

A diverse array of concurrent events performed by various subjects is frequently depicted in real-world videos. To implement the DVC model, datasets must include captions and detailed event timeline annotations. Ranjay Krishna et al. introduced the ActivityNet caption dataset, a state-of-the-art dataset for dense video captioning, to address the challenge [21]. The dataset consists of 20,000 videos, including both trimmed and untrimmed formats. It is divided into three subsets: training, validation, and test. The distinct start and end times assigned to each annotation enable the precise localization of events within the video. A total of 100,000 annotations are generated from 180-second videos, with an average of 3.65 temporally grounded sentences. The average length of a sentence is 13.46 words. This dataset has been extensively employed in DVC research due to its detailed annotations and comprehensive nature. Furthermore, it is observed that 10% of the temporal descriptions overlapped, indicating that the events occurred simultaneously.

The proposed TL-DVC model integrates a C3D network with a two-layer LSTM-based proposal module and an encoder-decoder Transformer architecture for caption generation. The C3D model extracts spatiotemporal features, which are subsequently compressed to 500 dimensions using Principal Component Analysis (PCA) to reduce redundancy and enhance computational efficiency. These reduced-dimensional features are fed into a two-layer LSTM network, where each layer comprises 512 hidden units and a dropout rate of 0.3 for generating event proposals. For the captioning module, we employ a T5-small Transformer model. The

encoder consists of six layers, each with eight self-attention heads, and each attention head utilizes a 64-dimensional projection for the query, key, and value matrices. The feed-forward network dimension is set to 2048. The decoder mirrors the architecture of the encoder, comprising six layers with eight attention heads per layer. To embed input tokens, we utilize the T5Tokenizer, which generates 512-dimensional token embeddings. For paragraph-level caption generation based on a large language model, we employed Azure OpenAI's GPT-4o model with the temperature parameter configured to 0.

Prior to the end-to-end training of the complete model, the proposal module was initially trained for five epochs to improve the performance of weight initialization. Beginning with a learning rate of 0.001 and a momentum of 0.9, the Adam optimization algorithm was implemented and subsequently adjusted downward during the training process. During training, the weights and biases of the LSTM nodes were initialized using a normal distribution and the He normal initialization method. This initialization strategy was employed to ensure stable convergence and effective learning. The spatial-temporal features were derived using transfer learning, and the C3D feature extraction model was maintained in its original form. The F1 score was used to determine the optimal tIoU value for the dataset after evaluating the generation of event proposals with multiple tIoU thresholds (0.5, 0.7, 0.8, 0.9). This experiment helps to decide the tIoU value, as a lower threshold value forwards all the proposals to the captioning module, while a higher threshold value may eliminate a few important events. After experimentation, we concluded that a tIoU threshold of 0.7 provides the best trade-off, achieving an F1@1000 score of 0.65. For training the captioning module, we utilized a pre-trained T5 model and fine-tuned it for the dense video captioning task.

Table 2. Evaluation of DVC Models on ActivityNet Caption Dataset. Higher Scores Indicate Better Performance. *Vid2seq Does Not Provide the METEOR and CIDEr Scores, So they have been Taken from [38].

Model	**Evaluation Measures**						
	BLEU @ 1	**BLEU @ 2**	**BLEU @ 3**	**BLEU @ 4**	**METEOR**	**ROUGH**	**CIDEr**
DenseCap[21]	17.95	7.69	3.89	2.2	4.05	-	17.29
Bi-SST[14]	19.37	8.84	4.41	2.3	9.6	19.29	12.68
JEDDi-Net[22]	19.97	9.1	4.06	1.63	8.58	19.63	19.88

Masked-Transformer[25]	-	-	4.76	2.23	9.56	-	-
DVC[18]	12.22	5.72	2.27	0.74	6.93	-	13.21
WS-DEC[45]	12.41	5.5	2.62	1.27	6.3	12.55	18.77
Multimodel-WSDEC [46]	10	4.2	1.92	0.94	5.03	10.39	14.27
SDVC[15]	17.92	7.99	2.94	0.93	8.82	-	30.68
RUC AI M3 [47]	16.59	9.65	5.32	2.91	11.28	-	14.03
GPaS [23]	19.78	9.96	5.06	2.34	10.75	-	14.84
EfficientNet [11]	-	-	2.54	1.1	5.7	-	-
AMT [28]	11.75	5.61	2.42	1.2	5.82	-	10.87
MDVC [43]	-	-	4.57	2.5	8.65	13.62	13.09
PDVC [30]	-	-	-	1.96	8.08	-	28.59
TL-NMS [48]	-	-	-	1.29	9.63	-	14.71
PPVC [31]	14.93	7.4	3.58	1.68	7.91	-	23.02
ViSE + Transformer [17]	22.18	10.92	5.58	2.72	10.78	21.98	23.89
SBS [35]	-	-	-	1.08	9.05	-	27.92
MPP-net [37]	-	-	-	2.04	7.61	-	29.76
Vid2seq* [36]	-	-	-	-	10	-	37.8
Streaming DVC [38]	-	-	-	-	9	-	41.2
CM-DVC [39]	-	-	-	2.88	9.43	-	40.24
DIBS [40]	-	-	-	-	8.93	-	31.89
Our model							
TL-DVC	23.01	11.27	5.78	2.84	12.32	22.64	25.81

The dense video captioning model is being assessed using conventional evaluation metrics, including BLEU [49], METEOR [50], CIDEr [51], and ROUGE [52]. The SODA metric [53] has been introduced recently for the evaluation of dense video captioning. Nevertheless, comparisons with historical models are not feasible due to the limited adoption of the metric by the research community. ROUGE is a recall-based evaluation measure, whereas BLEU is precision-based. METEOR's capacity to evaluate synonymous matching results in a higher level of consistency than BLEU, particularly in datasets with single reference sentences. The consensus between reference sentences and generated sentences is assessed by CIDEr, which employs a weighted tf-idf approach. This evaluation is indicative of the

alignment of textual summaries with visual content. This assessment is conducted before the final video-level caption is generated using the LLM. Video-level captioning is crucial for converting individual event captions into a cohesive narrative that maintains the overall flow of the video.

A comparative analysis of the proposed TL-DVC model against several state-of-the-art dense video captioning methods on the ActivityNet Captions dataset is presented in Table 2. The ViSE+Transformer model [17] previously reported the highest METEOR score at 10.78. Our model outperforms the ViSE model by 14.29%, as proven by its METEOR score of 12.32. This suggests that the proposed model, which integrates both past and future event proposals to enhance the features of the current timestamp, offers a more comprehensive understanding of event semantics. TL-DVC reported a 9.22% improvement in compared to the competitive RUC-AI M3 model, which reported a METEOR score of 11.28. It is important to note that our TL-DVC model and ViSE both achieve superior METEOR scores without relying on vision transformer features. Conversely, end-to-end transformer-based models, including Vid2Seq [36], DIBS [40], and StreamingDVC [38], exhibit higher CIDEr scores. This trend suggests that transformer-based architectures benefit significantly from richer visual features in generating accurate and contextually coherent captions. In addition, we have employed the BERTScore to assess the coherence of the final LLM-generated video description in relation to the original proposal-level captions. The generated paragraph is only accepted if the BERTScore exceeds 85%, which indicates strong semantic alignment and coherence. Furthermore, we calculate the BLEU score to evaluate n-gram-level word overlap, which aids in the identification and reduction of hallucinated content in the LLM.

The results indicate that the TL-DVC model outperforms current methods; however, there are still obstacles to enhancing CIDEr scores. This suggests that future research should focus on improving the alignment between visual and textual features to generate more descriptive and effective captions. Furthermore, integrating video-level visual features into the final video-level summary generation process could contribute to preserving narrative consistency across all event captions.

5. Conclusion

This paper introduces a novel methodology for dense video captioning, termed Transformer and LLM-based Dense Video Captioning (TL-DVC). The proposed model

employs C3D and SST-based event proposal techniques to extract visual features at the current timestamp, integrating information from both past and future events to enhance contextual understanding. These extracted features, in conjunction with the original C3D features, are input into an encoder-decoder-based Transformer to generate captions for each event. The Transformer encoder processes the visual features, while the decoder generates captions based on the encoded visual representations. To ensure narrative coherence throughout the video, a Large Language Model (LLM) is utilized to synthesize and produce a final video-level caption from the individually generated event captions. The TL-DVC model demonstrates superior performance compared to current state-of-the-art approaches on the ActivityNet Captions dataset, achieving a 9.22% increase in METEOR score, improving from 11.28 to 12.32.

References

[1] Kim, Jinkyu, Anna Rohrbach, Trevor Darrell, John Canny, and Zeynep Akata. "Textual explanations for self-driving vehicles." In Proceedings of the European conference on computer vision (ECCV), 2018, 563-578.

[2] Potapov, Danila, Matthijs Douze, Zaid Harchaoui, and Cordelia Schmid. "Category-specific video summarization." In Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part VI 13, Springer International Publishing, 2014, 540-555.

[3] Dinh, Quang Minh, Minh Khoi Ho, Anh Quan Dang, and Hung Phong Tran. "Trafficvlm: A controllable visual language model for traffic video captioning." In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2024, 7134-7143.

[4] Yang, Haojin, and Christoph Meinel. "Content based lecture video retrieval using speech and video text information." IEEE transactions on learning technologies 7, no. 2 (2014): 142-154.

[5] Anne Hendricks, Lisa, Oliver Wang, Eli Shechtman, Josef Sivic, Trevor Darrell, and Bryan Russell. "Localizing moments in video with natural language." In Proceedings of the IEEE international conference on computer vision, 2017, 5803-5812.

[6] Aggarwal, Akshay, Aniruddha Chauhan, Deepika Kumar, Mamta Mittal, Sudipta Roy, and Tai-hoon Kim. "Video caption based searching using end-to-end dense captioning and sentence embeddings." Symmetry 12, no. 6 (2020): 992.

[7] Wu, Shaomei, Jeffrey Wieland, Omid Farivar, and Julie Schiller. "Automatic alt-text: Computer-generated image descriptions for blind users on a social network service." In proceedings of the 2017 ACM conference on computer supported cooperative work and social computing, 2017, 1180-1192.

[8] V. U. K. V. Y. S. D. R. B., Sai Jyothi and R. S. G., "Intelligent faq chatbot: A user-centric approach using large language models," Journal of Artificial Intelligence and Capsule Networks, vol. 7, no. 1, 2025, https://doi.org/10.36548/jaicn.2025.1.006, 78–93.

[9] Shi, Botian, Lei Ji, Yaobo Liang, Nan Duan, Peng Chen, Zhendong Niu, and Ming Zhou. "Dense procedure captioning in narrated instructional videos." In Proceedings of the 57th annual meeting of the association for computational linguistics, 2019, 6382-6391.

[10] Chen, Yangyu, Shuhui Wang, Weigang Zhang, and Qingming Huang. "Less is more: Picking informative frames for video captioning." In Proceedings of the European conference on computer vision (ECCV), 2018, 358-373.

[11] Suin, Maitreya, and A. N. Rajagopalan. "An efficient framework for dense video captioning." In Proceedings of the AAAI Conference on Artificial Intelligence, vol. 34, no. 07, 2020, 12039-12046.

[12] Escorcia, Victor, Fabian Caba Heilbron, Juan Carlos Niebles, and Bernard Ghanem. "Daps: Deep action proposals for action understanding." In Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part III 14, Springer International Publishing, 2016, 768-784.

[13] Buch, Shyamal, Victor Escorcia, Chuanqi Shen, Bernard Ghanem, and Juan Carlos Niebles. "Sst: Single-stream temporal action proposals." In Proceedings of the IEEE conference on Computer Vision and Pattern Recognition, 2017, 2911-2920.

[14] Wang, Jingwen, Wenhao Jiang, Lin Ma, Wei Liu, and Yong Xu. "Bidirectional attentive fusion with context gating for dense video captioning." In Proceedings of the IEEE conference on computer vision and pattern recognition, 2018, 7190-7198.

[15] Mun, Jonghwan, Linjie Yang, Zhou Ren, Ning Xu, and Bohyung Han. "Streamlined dense video captioning." In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2019, 6588-6597.

[16] Krishna, Ranjay, Kenji Hata, Frederic Ren, Li Fei-Fei, and Juan Carlos Niebles. "Dense-captioning events in videos." In Proceedings of the IEEE international conference on computer vision, 2017, 706-715.

[17] Aafaq, Nayyer, Ajmal Mian, Naveed Akhtar, Wei Liu, and Mubarak Shah. "Dense video captioning with early linguistic information fusion." IEEE Transactions on Multimedia 25 (2022): 2309-2322.

[18] Li, Yehao, Ting Yao, Yingwei Pan, Hongyang Chao, and Tao Mei. "Jointly localizing and describing events for dense video captioning." In Proceedings of the IEEE conference on computer vision and pattern recognition, 2018, 7492-7500.

[19] GHuang, Gabriel, Bo Pang, Zhenhai Zhu, Clara Rivera, and Radu Soricut. "Multimodal pretraining for dense video captioning." arXiv preprint arXiv:2011.11760 (2020).

[20] Shen, Zhiqiang, Jianguo Li, Zhou Su, Minjun Li, Yurong Chen, Yu-Gang Jiang, and Xiangyang Xue. "Weakly supervised dense video captioning." In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, 1916-1924.

[21] Krishna, Ranjay, Kenji Hata, Frederic Ren, Li Fei-Fei, and Juan Carlos Niebles. "Dense-captioning events in videos." In Proceedings of the IEEE international conference on computer vision, 2017, 706-715.

[22] Xu, Huijuan, Boyang Li, Vasili Ramanishka, Leonid Sigal, and Kate Saenko. "Joint event detection and description in continuous video streams." In 2019 IEEE winter conference on applications of computer vision (WACV), IEEE, 2019, 396-405.

[23] Zhang, Zhiwang, Dong Xu, Wanli Ouyang, and Luping Zhou. "Dense video captioning using graph-based sentence summarization." IEEE Transactions on Multimedia 23 (2020): 1799-1810.

[24] Qi, Charles R., Hao Su, Kaichun Mo, and Leonidas J. Guibas. "Pointnet: Deep learning on point sets for 3d classification and segmentation." In Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, 652-660.

[25] Zhou, Luowei, Yingbo Zhou, Jason J. Corso, Richard Socher, and Caiming Xiong. "End-to-end dense video captioning with masked transformer." In Proceedings of the IEEE conference on computer vision and pattern recognition, 2018, 8739-8748.

[26] Hershey, Shawn, Sourish Chaudhuri, Daniel PW Ellis, Jort F. Gemmeke, Aren Jansen, R. Channing Moore, Manoj Plakal et al. "CNN architectures for large-scale audio classification." In 2017 ieee international conference on acoustics, speech and signal processing (icassp), IEEE, 2017, 131-135.

[27] Iashin, Vladimir, and Esa Rahtu. "A better use of audio-visual cues: Dense video captioning with bi-modal transformer." arXiv preprint arXiv:2005.08271 (2020).

[28] Yu, Zhou, and Nanjia Han. "Accelerated masked transformer for dense video captioning." Neurocomputing 445 (2021): 72-80.

[29] Iashin, Vladimir, and Esa Rahtu. "Multi-modal dense video captioning." In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops, 2020, 958-959.

[30] Wang, Teng, Ruimao Zhang, Zhichao Lu, Feng Zheng, Ran Cheng, and Ping Luo. "End-to-end dense video captioning with parallel decoding." In Proceedings of the IEEE/CVF international conference on computer vision, 2021, 6847-6857.

[31] Choi, Wangyu, Jiasi Chen, and Jongwon Yoon. "Parallel pathway dense video captioning with deformable transformer." IEEE Access 10 (2022): 129899-129910.

[32] Rahman, Tanzila, Bicheng Xu, and Leonid Sigal. "Watch, listen and tell: Multi-modal weakly supervised dense event captioning." In Proceedings of the IEEE/CVF international conference on computer vision, 2019, 8908-8917.

[33] Deng, Chaorui, Shizhe Chen, Da Chen, Yuan He, and Qi Wu. "Sketch, ground, and refine: Top-down dense video captioning." In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, 234-243.

[34] Aafaq, Nayyer, Ajmal Mian, Naveed Akhtar, Wei Liu, and Mubarak Shah. "Dense video captioning with early linguistic information fusion." IEEE Transactions on Multimedia 25 (2022): 2309-2322.

[35] Choi, Wangyu, Jiasi Chen, and Jongwon Yoon. "Step by step: A gradual approach for dense video captioning." IEEE Access 11 (2023): 51949-51959.

[36] Yang, Antoine, Arsha Nagrani, Paul Hongsuck Seo, Antoine Miech, Jordi Pont-Tuset, Ivan Laptev, Josef Sivic, and Cordelia Schmid. "Vid2seq: Large-scale pretraining of a visual language model for dense video captioning." In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023, 10714-10726.

[37] Wei, Yiwei, Shaozu Yuan, Meng Chen, Xin Shen, Longbiao Wang, Lei Shen, and Zhiling Yan. "MPP-net: multi-perspective perception network for dense video captioning." Neurocomputing 552 (2023): 126523.

[38] Zhou, Xingyi, Anurag Arnab, Shyamal Buch, Shen Yan, Austin Myers, Xuehan Xiong, Arsha Nagrani, and Cordelia Schmid. "Streaming dense video captioning." In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2024, 18243-18252.

[39] Kim, Minkuk, Hyeon Bae Kim, Jinyoung Moon, Jinwoo Choi, and Seong Tae Kim. "Do you remember? dense video captioning with cross-modal memory retrieval." In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2024, 13894-13904.

[40] Wu, Hao, Huabin Liu, Yu Qiao, and Xiao Sun. "DIBS: Enhancing dense video captioning with unlabeled videos via pseudo boundary enrichment and online refinement." In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2024, 18699-18708.

[41] Duan, Xuguang, Wenbing Huang, Chuang Gan, Jingdong Wang, Wenwu Zhu, and Junzhou Huang. "Weakly supervised dense event captioning in videos." Advances in Neural Information Processing Systems 31 (2018).

[42] Aytar, Yusuf, Carl Vondrick, and Antonio Torralba. "Soundnet: Learning sound representations from unlabeled video." Advances in neural information processing systems 29 (2016).

[43] Estevam, Valter, Rayson Laroca, Helio Pedrini, and David Menotti. "Dense video captioning using unsupervised semantic information." arXiv preprint arXiv:2112.08455 (2021).

[44] Ji, Shuiwang, Wei Xu, Ming Yang, and Kai Yu. "3D convolutional neural networks for human action recognition." IEEE transactions on pattern analysis and machine intelligence 35, no. 1 (2012): 221-231.

[45] Duan, Xuguang, Wenbing Huang, Chuang Gan, Jingdong Wang, Wenwu Zhu, and Junzhou Huang. "Weakly supervised dense event captioning in videos." Advances in Neural Information Processing Systems 31 (2018).

[46] Rahman, Tanzila, Bicheng Xu, and Leonid Sigal. "Watch, listen and tell: Multi-modal weakly supervised dense event captioning." In Proceedings of the IEEE/CVF international conference on computer vision, 2019, 8908-8917.

[47] Song, Yuqing, Shizhe Chen, Yida Zhao, and Qin Jin. "Team ruc_aim3 technical report at activitynet 2020 task 2: Exploring sequential events detection for dense video captioning." arXiv preprint arXiv:2006.07896 (2020).

[48] Wang, Teng, Huicheng Zheng, Mingjing Yu, Qian Tian, and Haifeng Hu. "Event-centric hierarchical representation for dense video captioning." IEEE Transactions on Circuits and Systems for Video Technology 31, no. 5 (2020): 1890-1900.

[49] Papineni, Kishore, Salim Roukos, Todd Ward, and Wei-Jing Zhu. "Bleu: a method for automatic evaluation of machine translation." In Proceedings of the 40th annual meeting of the Association for Computational Linguistics, 2002, 311-318.

[50] Banerjee, Satanjeev, and Alon Lavie. "METEOR: An automatic metric for MT evaluation with improved correlation with human judgments." In Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization, 2005, 65-72.

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

[52] Lin, Chin-Yew. "Rouge: A package for automatic evaluation of summaries." In Text summarization branches out, 2004, 74-81.

[53] Fujita, Soichiro, Tsutomu Hirao, Hidetaka Kamigaito, Manabu Okumura, and Masaaki Nagata. "Soda: Story oriented dense video captioning evaluation framework." In Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part VI 16, Springer International Publishing, 2020, 517-531.