Multimedia Systems (2023) 29:2469–2482
https://doi.org/10.1007/s00530-023-01130-w

REGULAR PAPER

Multimodal-enhanced hierarchical attention network for video captioning

Maosheng Zhong[1] · Youde Chen[1] · Hao Zhang[1] · Hao Xiong[1] · Zhixiang Wang[1]

Received: 18 March 2023 / Accepted: 26 June 2023 / Published online: 15 July 2023

Abstract
In video captioning, many pioneering approaches have been developed to generate higher-quality captions by exploring and adding new video feature modalities. However, as the number of modalities increases, the negative interaction between them gradually reduces the gain of caption generation. To address this problem, we propose a three-layer hierarchical attention network based on a bidirectional decoding transformer that enhances multimodal features. In the first layer, we execute different encoders according to the characteristics of each modality to enhance the vector representation of each modality. Then, in the second layer, we select keyframes from all sampled frames of the modality by calculating the attention value between the generated words and each frame of the modality. Finally, in the third layer, we allocate weights to different modalities to reduce redundancy between them before generating the current word. Additionally, we use a bidirectional decoder to consider the context of the ground-truth caption when generating captions. Experiments on two mainstream benchmark datasets, MSVD and MSR-VTT, demonstrate the effectiveness of our proposed model. The model achieves state-of-the-art performance in significant metrics, and the generated sentences are more in line with human language habits. Overall, our three-layer hierarchical attention network based on a bidirectional decoding transformer effectively enhances multimodal features and generates high-quality video captions. Codes are available on https://github.com/nickchen121/MHAN.

Keywords Video captioning · Bidirectional decoding transformer · Multimodal enhancement · Hierarchical attention network

1 Introduction

The task of video captioning involves understanding the scenes in a video and describing them with plausible sentences. This task has numerous applications, including video retrieval, video recommendation, disabled support, and scene understanding. With the rapid development of deep learning, neural caption methods based on encoder-decoder architectures have become increasingly popular for video captioning [1–4]. In particular, transformer-based models have advanced the state-of-the-art [5–9].

To exploit the temporal structure of the video, the utilization of video feature modalities has attracted significant attention. While many methods [10–14] have been proposed for solving the video captioning task, most employ a simple concatenation method to fuse modalities, which may lead to redundancy as the number of modalities increases. Therefore, novel multimodal fusion methods for video captioning have been proposed [15–19], demonstrating that reducing redundancy between modalities through a reasonable fusion approach can improve a model's performance more effectively.

Inspired by these methods, we first extract four video feature modalities, including image, motion, object, and relationship between objects separately through

Communicated by P. Pala.

✉ Youde Chen
nickchen121@163.com

Maosheng Zhong
zhongmaosheng@sina.com

Hao Zhang
904666694@qq.com

Hao Xiong
1506035201@qq.com

Zhixiang Wang
229894871@qq.com

[1] Jiangxi Normal University, 99 Ziyang Avenue, Nanchang 330022, China

Inception-ResNet-V2, I3D, Mask R-CNN and TransE. Then we enhance the vector representation of modalities and reduce redundancy between modalities through a three-layer hierarchical attention network. The network is divided into the following three layers:

- Self-encoding of modalities (Transformer Encoder): This layer obtains the association between frames to enhance the vector representation of each modality.
- Keyframes selected attention: This layer selects keyframes through attention calculation between the currently generated words and the frames of each modality.
- Multimodal fusion attention: This layer assigns weights to different modalities to reduce redundancy between modalities before generating the current word.

Self-Encoding of Modalities. Given a video frame sequence $V = V_1, \cdots, V_L$ with a sampling frame length of L, the i-th feature modality of the video is expressed as $F_i = F_{i1}, \cdots, F_{iL}$. After obtaining the video's image, motion, and object modalities, we obtain the association between their frames through self-attention calculation to fully capture the semantic information of the modalities. Among them, we capture the association between smaller video clips than frames by using extended multi-head attention to obtain fine-grained image and motion information. Since the relationship modality is composed of a series of entities that are not internally associated, we do not use self-attention calculation for it.

Keyframes selected attention. In video captioning, each word corresponds to an object or scene in the video. However, unimportant objects or consecutive similar frames may repeatedly appear in multiple frames, making the generated captions inaccurate. As shown in Fig. 1, an unimportant object such as a "dog" can be mistakenly selected as the subject of the description. Therefore, it is essential to accurately identify valuable objects in the video and remove redundant frames to make the information transmitted from the encoder to the decoder more accurate. To achieve this, we not only obtain objects in the video using Mask R-CNN, but also obtain the relationship between objects using TransE. This allows us to indirectly reduce the weight of some useless objects through the mining of relationships. Additionally, we calculate the attention value between the generated words and each frame in a modality to automatically select keyframes and reduce redundant frames. In Fig. 1, after we select keyframes, the subject of the video description becomes a woman, which is consistent with the ground truth.

Multimodal fusion attention. With the increase in the number of feature modalities, because different modalities may carry task-relevant information at different times, fusing them by a concatenation method of element-wise addition may limit the model's ability to dynamically determine the relevance of each type of feature to different parts of the

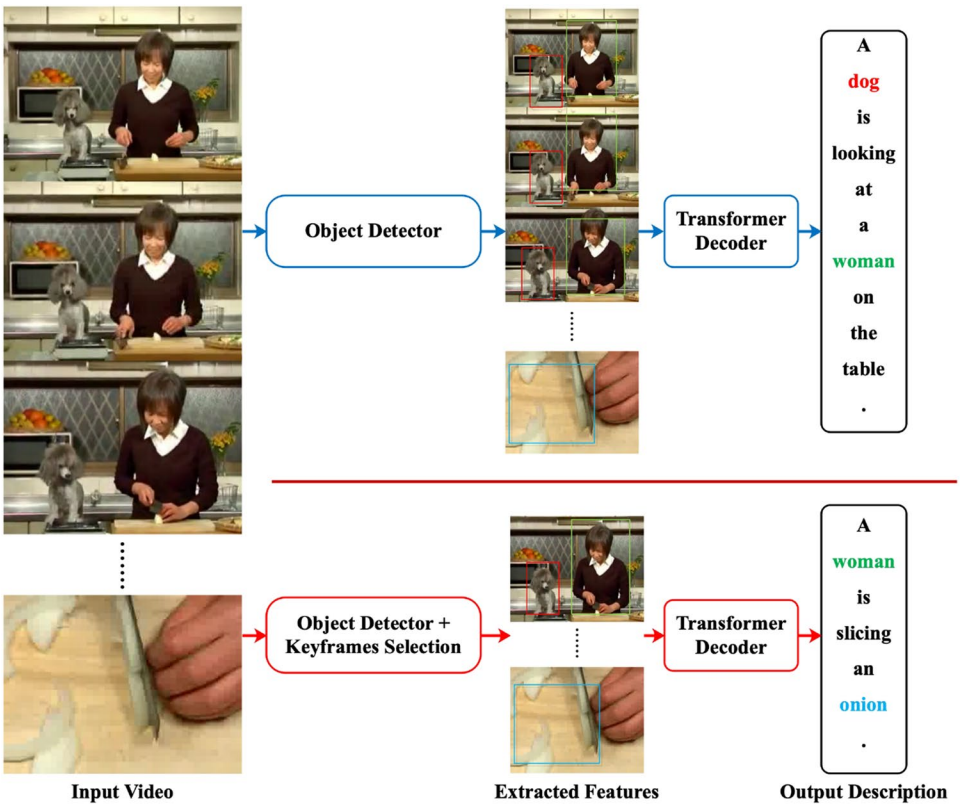

Fig. 1 An example of keyframe selection: suppose the ground truth for a given video is that a woman is cutting an onion. If we rely solely on an object detector, it may mistakenly identify unimportant objects such as a "dog" in redundant frames as the subject of the description. Keyframe selection can reduce such redundancy and enable us to generate more accurate sentences

description [9]. At the same time, different input modalities are crucial for selecting the current word. For example, the caption "a girl is getting into a car" refers to objects and their relations. On the contrary, "a girl is jumping on a car" may rely on motion features to determine the behaviour of "jumping". To solve this problem, before the decoder generates the current word, we calculate the attention value between the generated words and different modalities for assigning weights to these modalities.

The decoder of the Transformer is an autoregressive decoder that only considers the generated words when generating the current word. To enhance the captions' generation ability, we adopt a bidirectional decoder consisting of a backward decoder and a forward decoder. The backward decoder generates captions in reverse order, and the forward decoder adds a cross multi-head attention to integrate the context of the reverse caption. This way, the forward decoder can take into account the context of the ground-truth caption every time it predicts the next word. Additionally, we generate pseudo reverse captions to mitigate the information leakage of the bidirectional decoder.

The main contributions of this paper can be summarized in four aspects.

- We use different encoders according to the characteristics of different modalities. For instance, we utilize the extended multi-head attention to capture the association between time clips smaller than video frames for obtaining fine-grained image and motion features.
- We address the issue of unimportant objects and several consecutive similar frames that may repeatedly appear in multiple frames and make the generation of captions inaccurate. We calculate the attention value between the generated words and each frame in modality to select the keyframes.
- We propose a method for multimodal fusion, called multimodal fusion attention, which assigns weights to different modalities to reduce the damage caused by the interaction between modalities to a caption generation.
- We evaluate the Multimodal-enhanced Hierarchical Attention Network for Video Captioning (MHAN) on two datasets, MSVD and MSR-VTT. In particular, our MHAN achieves state-of-the-art performance and outperforms the runner-up methods by a large margin in CIDEr-D, which is specially designed for captioning tasks.

The rest of this paper is organized as follows. We first discuss related work on video captioning in Sect. 2. Secondly, we detail the Multimodal-enhanced Hierarchical Attention Network for Video Captioning (MHAN) in Sect. 3. Then, we present the experimental results and analyses in Sects. 4 and 5, respectively. Finally, we conclude the paper in Sect. 6.

2 Related work

This section describes the related work from two perspectives: (1) Video Captioning and (2) Multimodal Extraction and Fusion.

2.1 Video captioning

Captioning a short video in natural language has been challenging for machines, but with the rapid development of deep learning, several methods have emerged to address this problem. Donahue et al. were the first to adopt a deep neural network to solve the video captioning problem [1]. Later, many video captioning methods based on encoder-decoder architecture rose to prominence [2–4]. These methods encode the video using a Convolutional Neural Network (CNN) [20] and employ a Long Short-Term Memory (LSTM) [21] to generate video captions. One of the first works to utilize an encoder-decoder framework is Venugopalan et al., where captions are generated by LSTM and visual features extracted by CNN [2]. With the emergence of the attention mechanism and Transformer [22], Singh et al. employ a hybrid attention mechanism by extending the soft temporal attention mechanism with a semantic attention to make the system able to decide when to focus on visual context vector and semantic input. [5]. Wang et al. proposed a bidirectional decoding Transformer to generate captions using the context [7]. The CLIP-DCD [8] proposed by Yang et al. uses CLIP [23] to encode video content and then uses a Transformer decoder to generate subtitles.

2.2 Multimodal extraction and fusion

The utilization of multimodal features in video captioning has attracted great attention due to the need to exploit the temporal structure of the video. For instance, Aafaq et al. leverage the state-of-the-art 2-D CNN and 3-D CNN (C3D [10]) pre-trained on a large dataset to extract visual spatio-temporal features [11]. With the success of object detection in Computer Vision, the bottom-up attention algorithm applies object detection to extract regional features, significantly improving the video captioning performance [12]. Zhang et al. respectively utilize object detection and knowledge graph to extract objects and relationships between the objects in the video and then fuse them with spatio-temporal features of the video to refine the fine-grained actions between the objects [6]. Jin proposed a deep multimodal embedding network that embeds four different modality features, namely frame, audio, motion, and category, into different LSTM layers according to their characteristics [13]. To discover and integrate the rich and primeval external knowledge (i.e., frame-based image caption), Liu

et al. proposed a Hierarchical & Multimodal Video Caption (HMVC) model to jointly learn the dynamics within both visual and textual modalities for video caption task, which infers an arbitrary length sentence according to the input video with arbitrary number of frames [14].

However, the above methods mainly supplement new feature modalities during encoding, and most use a simple concatenation method to fuse these modalities. When the number of modalities increases, fusing them by simple concatenation may limit the model's ability to dynamically determine the relevance of each type of feature to different parts of the description and weaken the vector representation of the video.

Therefore, several novel video captioning methods [9, 15–19] based on multimodal fusion have been proposed. Qin et al. propose a multi-modal fusion encoder that fuses features from visual, aural, speech and meta modalities to represent video contents [15]. Hori et al. introduced a multimodal attention model that can selectively utilize features from different modalities for each word in the output description [9]. Based on the mainstream pre-fusion and post-fusion methods, Yubo Jiang proposed a two-stage method to fuse features and improve the accuracy of natural language description [16]. Li et al. proposed an adaptive spatial location module for the video captioning task, which dynamically predicts the importance of each video frame in generating the description sentence [17]. Huang et al. proposed the XlanV model, which decides whether to attend to static or dynamic features by fusing these multi-modality features adaptively [18]. Yan et al. constructed a multimodal features fusion network to learn the relationship between different feature modalities, which is used to fuse different feature modalities [19].

2.3 Summary

Inspired by the aforementioned methods, we extract four feature modalities from videos: image, motion, object, and the relationship between objects. However, when we fuse these modalities using a simple concatenation method with element-wise addition, we do not observe a noticeable improvement in model performance. To address this issue and further improve accuracy, we propose a three-layer hierarchical attention network based on a bidirectional decoding transformer. This network enhances the vector representation of the modalities and reduces redundancy between them.

3 Approach

In this section, we first briefly introduce the basic modules used in our approach, and then describe our approach in detail.

3.1 Basic module

These basic modules are based on a transformer architecture and multimodal feature extraction.

3.1.1 Transformer architecture

The transformer architecture consists of an encoder and a decoder, each composed of a stack of identical layers. Each layer comprises sublayers constructed by Multi-Head Attention (MHA) and Feed-Forward Network (FFN). Additionally, there is a residual connection around the two sub-layers, followed by layer normalization.

The MHA sublayer allows the model to jointly attend to information from different representation subspaces at different positions and comprises h identical heads. Each head_i uses $Q_i = QW_i^Q, K_i = KW_i^K$, and $V_i = VW_i^V$ as input. The MHA is calculated as follows:

$$\begin{aligned} &\text{MHA}(Q,K,V) = \text{Concat}(\text{head}_1,\dots,\text{head}_h)W^O, \\ &\text{where head}_i = \text{Attention}(QW_i^Q, KW_i^K, VW_i^V), \end{aligned} \tag{1}$$

where, W_i^Q, W_i^K, and W_i^V are learned parameters, and $W_i^Q, W_i^K, W_i^V \in \mathbb{R}^{N\times d_k}$, where W^O is a learned projection matrix.

The attention in Eq. (1) is known as "Scaled Dot-Product Attention." It maps a query and a set of key-value pairs to an output:

$$\text{Attention}(Q_i,K_i,V_i) = \text{softmax}\left(\frac{Q_iK_i^T}{\sqrt{d_k}}\right)V_i, \tag{2}$$

where $E_i = \frac{Q_iK_i^T}{\sqrt{d_k}}$ represents the $N\times N$ attention score matrix E_i, whose element e_i^{mn} is the attention score between the m-th and n-th token in the i-th subspace.

The FFN sublayer comprises two linear transformations with a ReLU activation in between:

$$FFN(X) = \max(0, XW_1 + b_1)W_2 + b_2, \tag{3}$$

where, $W_1 \in \mathbb{R}^{d\times d_f}, W_2 \in \mathbb{R}^{d_f\times d}, b_1 \in \mathbb{R}^{d_f}$, and $b_2 \in \mathbb{R}^d$ are learned parameters, and $d = d_k \times h$.

3.1.2 Feature modalities extraction

Multimodal extraction using image and motion modalities has become prevalent in video captioning. The image modality is typically used to express the colors and shapes in the image, while the motion modality is essential for capturing the spatio-temporal motion dynamics in the video. In this study, given a sequence of video frames $V = V_1,\cdots,V_L$ with length L, we employed the Inception-ResNet-V2 [24]

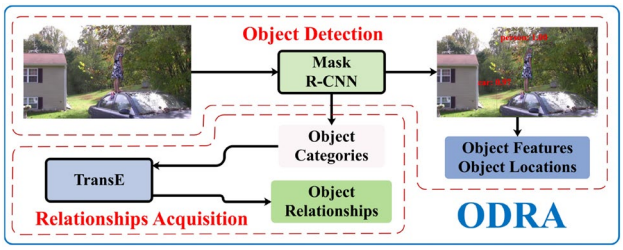

Fig. 2 The figure illustrates our proposed Object Detection and Relationship Acquisition (ODRA) framework

pre-trained on ImageNet [20] to extract the image features $F_I = F_{I1}, \cdots, F_{IL}$, where $F_{Ii} \in \mathbb{R}^{d_I}$. We also utilized the I3D [25] pre-trained on Kinetics [26] to extract the motion features $F_M = F_{M1}, \cdots, F_{ML}$, where $F_{Mi} \in \mathbb{R}^{d_M}$.

Moreover, the object modality in the video exhibits a dynamic behavior compared to those in the picture, with links between them. As illustrated in Fig. 2 using the Object Detection and Relationship Acquisition (ODRA) framework, we first leverage the Mask R-CNN [27] object detector to extract the semantic object features $F_O = F_{O1}, \cdots, F_{OL}$, where $F_{Oi} \in \mathbb{R}^{d_O}$ (such as people, buildings, and cars). We then acquire relationship features $F_R = F_{R1}, \cdots, F_{RL}$, where $F_{Ri} \in \mathbb{R}^{d_R}$, between the objects using the TransE [28] knowledge graph model based on the knowledge representation learning framework OpenKE [29].

Object detection. We extract the location and eigenvectors of objects using Mask R-CNN pre-trained on the COCO dataset [30], and obtain the categories of the objects from a set of 50 common categories. Specifically, we first input a video V_i into the Feature Pyramid Network [31] to obtain its feature maps. We then use the Region Proposal Network [32] to filter out a series of Regions of Interest (ROI). Finally, as it extracts ROI of different sizes, we employ an ROI pooling layer to obtain the exact ROI size, namely Object Regions (OR). The position coordinates of the OR, the eigenvectors, and the categories are expressed as $R_{li} = [l_1, \cdots, l_k]$, $R_{vi} = [v_1, \cdots, v_k]$ and $R_{oi} = [o_1, \cdots, o_k]$, respectively, where $R_{li} \in \mathbb{R}^{d_l}, R_{vi} \in \mathbb{R}^{d_v}, R_{oi} \in \mathbb{R}^{d_o}$,$k$ represents the k-th detected object, $l_j = [\frac{x_j}{w_f}, \frac{y_j}{h_f}, \frac{w_j}{w_f}, \frac{h_j}{h_f}]$ provides the center coordinates $(\frac{x_j}{w_f}, \frac{y_j}{h_f})$, width $\frac{w_j}{w_f}$ and height $\frac{h_j}{h_f}$ of the j-th object area after normalization according to the size of the video frame with width w_f and height h_f, and v_j represents the eigenvector of the j-th object. Finally, we concatenate R_{li} and R_{vi} to obtain the object features, denoted as F_{Oi}, as follows:

$$F_{Oi} = [R_{li}; R_{vi}], \tag{4}$$

where $F_{Oi} \in \mathbb{R}^{d_l + d_v}$, the $[\cdot;\cdot]$ denotes the concatenation of two matrices.

Relationships acquisition based on knowledge graph. We first use the categories R_{oi} as input for TransE to predict relationships $R_{ri} = [r_1, \cdots, r_{\frac{(k-1)k}{2}}]$ between the objects and represented the relationships using 300-dimensional GLOVE vectors, where $R_{ri} \in \mathbb{R}^{d_{glove}}$. Then we encode R_{oi} and R_{ri} into word vectors using Word2Vec [33]. The encoding process is as follows:

$$R'_{oi} = R_{oi} W_o, \tag{5}$$

$$R'_{ri} = R_{ri} W_r, \tag{6}$$

where the embedding matrix $W_o \in \mathbb{R}^{d_{glove} \times d_{model}}$ and $W_r \in \mathbb{R}^{d_o \times d_{model}}$ is pre-trained by Word2Vec. Finally, we synthesize the relationship features using R'_{oi} and R'_{ri} as:

$$F_{Ri} = [R'_{oi}; R'_{ri}], \tag{7}$$

where $F_{Ri} \in \mathbb{R}^{2 \times d_{model}}$.

3.2 Framework: multimodal-enhanced hierarchical attention network

As shown in Fig. 3, our proposed Multimodal-enhanced Hierarchical Attention Network (MHAN) mainly consists of an Objects and Relationships Encoder (ORE) and a Bidirectional Decoder (BDD).

3.2.1 Objects and relationships encoder (ORE)

The objects and relationships encoder is primarily responsible for processing the four feature modalities at two levels. In the first stage, each modality is encoded separately using different encoders. In the second stage, our model can select keyframes from all sampled frames of each modality.

3.2.2 Self-encoding of feature modalities

In Section 3.1.2, given a sequence of video frames $V = \{V_1, \cdots, V_L\}$ with length L, we obtain image features $F_I = \{F_{I1}, \cdots, F_{IL}\} \in \mathbb{R}^{L \times d_I}$, motion features $F_M \in \mathbb{R}^{L \times d_M}$, object features $F_O \in \mathbb{R}^{L \times d_O}$, and relationship features $F_R \in \mathbb{R}^{L \times d_R}$ in the video. To unify the dimensions of the four feature modalities, we apply linear transformations to map their dimensions to d_{model}, where the calculation formula is:

$$F'_{Ii} = w_{Ii} F_{Ii} + b_{Ii}, \; b_{Ii} \in \mathbb{R}^{d_{model}}, \tag{8}$$

$$F'_{Mi} = w_{Mi} F_{Mi} + b_{Mi}, \; b_{Mi} \in \mathbb{R}^{d_{model}}, \tag{9}$$

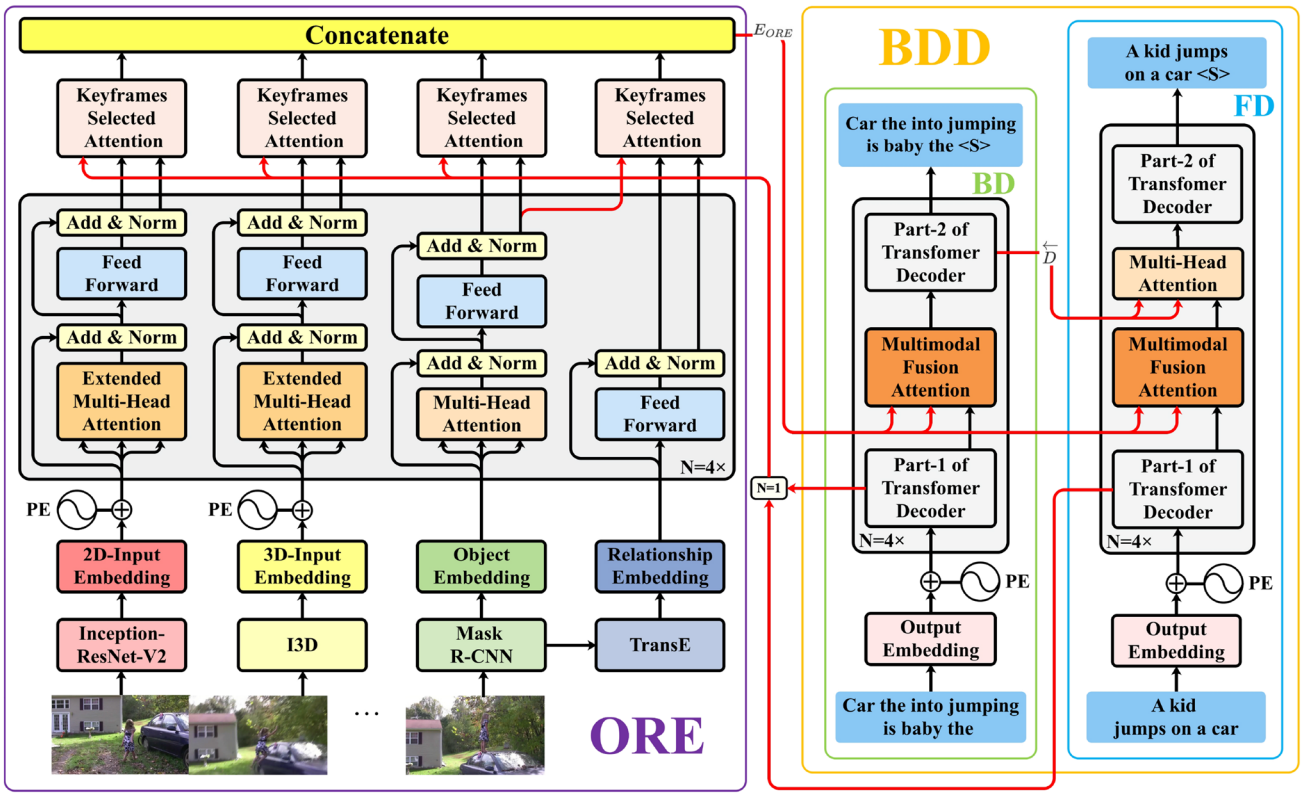

Fig. 3 The figure illustrates our proposed Multimodal-enhanced Hierarchical Attention Network(MHAN)based on the bidirectional decoding transformer in detail. It consists of two modules: (1) the Objects and Relationships Encoder (ORE), which is divided into two parts and encodes four feature modalities to obtain the output E_{ORE}. The first part is self-encoding (Transformer Encoder), which captures the association between frames to enhance the vector representation of each modality. The second part is the keyframes selected attention, which selects keyframes from all sampled frames for each modality. (2) The Bidirectional Decoder (BDD), which consists of a backward decoder (BD) and a forward decoder (FD). The backward decoder generates a reverse caption, and the forward decoder adds cross multi-head attention to integrate the context $\overleftarrow{D}$ of the reverse caption. Each decoder assigns weights to different modalities through multimodal fusion attention (Multimodal fusion attention is one of the internal structures of the transformer decoder), according to the generated captions and the output E_{ORE}, to affect the generation of the current word

$$F'_{Oi} = w_{Oi}F_{Oi} + b_{Oi},\ b_{Oi} \in \mathbb{R}^{d_{model}}, \tag{10}$$

$$F'_{Ri} = w_{Ri}F_{Ri} + b_{Ri},\ b_{Ri} \in \mathbb{R}^{d_{model}}, \tag{11}$$

where $w_{Ii} \in \mathbb{R}^{d_{model} \times d_I}$, $w_{Mi} \in \mathbb{R}^{d_{model} \times d_M}$, $w_{Oi} \in \mathbb{R}^{d_{model} \times d_O}$ and $w_{Ri} \in \mathbb{R}^{d_{model} \times d_R}$.We then obtain $F'_I = \{F'_{I1}, \cdots, F'_{IL}\} \in \mathbb{R}^{L \times d_{model}}$, F'_M, F'_O, and F'_R, respectively.

Next, to strengthen the vector representation of these modalities through the connection between frames, we send F'_I and F'_M into different encoders of Transformer as inputs to acquire $E_I = E_{I1}, \cdots, E_{IL} \in \mathbb{R}^{L \times d_{model}}$ and E_M. The calculation formula for attention in the encoder is (taking E_I as an example):

$$\text{Attention}(E_I, E_I, E_I) = \text{softmax}\left(\frac{(E_I W_Q^T)(E_I W_K^T)^T}{\sqrt{d_k}}\right)(E_I W_V^T), \tag{12}$$

where W_Q, W_K, and W_V are learned parameters, and $W_Q, W_K, W_V \in \mathbb{R}^{d_{model} \times d_{model}}$. We use extended multi-head attention in the above two encoders, as multi-head attention (Vaswani et al. utilize eight heads) allows a model to jointly attend to information from different representation subspaces at different positions [22]. When we employ the extended multi-head attention (our model uses 64 heads, that is, $h = 64$ in Eq. (1)), MHAN can capture the correlation of different positions in time clips smaller than video frames.

Finally, similar to the process of encoding image and motion features, we use multi-head attention with eight heads to encode F_O, resulting in the object encoding $E_O = E_{O1}, \cdots, E_{OL}$. However, there is no relevant order between F'_{Ri} and F'_{Rj}, so we employ an encoder without an attention layer to encode F_R, resulting in the relationship encoding E_r. We do not use positional encoding to encode the object and relationship modalities because they are not contextually ordered.

3.2.3 Keyframes selected attention

In Sect. 3.2.2, we obtain the encoding of four feature modalities, namely E_I, E_M, E_O, and E_R, respectively. To reduce redundancy between frames, we select K keyframes from each modality of a video. These K keyframes are the most dissimilar visually.

Taking the image encoding $E_I = E_{I1}, \cdots, E_{IL} \in \mathbb{R}^{L \times d_{model}}$ as an example, where L is the number of all frames in a video, when we predict the word of caption y_t, we obtain the word embedding $E_{t-1} \in \mathbb{R}^{L \times d_{model}}$. This is an embedded representation of the first $t-1$ words through the masked multi-head attention of layer one decoder (N=1). We then calculate the similarity $\alpha \in \mathbb{R}^{L \times L}$ between E_{t-1} and E_I using the following equation:

$$\alpha = softmax\left(\frac{E_{t-1}E_I^T}{\sqrt{d_k}}\right). \tag{13}$$

For N videos in a batch, we calculate the similarity and concatenate the resulting α_i:

$$A = [\alpha_1; \alpha_2; \cdots; \alpha_N], \tag{14}$$

where $A \in \mathbb{R}^{N \times L \times L}$.

We then count the number N_0 of non-zero elements using PyTorch on A and use K_I as the number of keyframes:

$$\begin{aligned} N_0 &= \text{torch.nonzero}(A), \\ K_I &= N_0 \parallel N. \end{aligned} \tag{15}$$

For each α_i in the batch, we take out the indices of the top K_I elements $\alpha_{i1}, \cdots, \alpha_{iK_I}$ and obtain the corresponding frames $E_I^{'} = E_{I1}, \cdots, E_{IK_I}$ in the image encoding E_I according to these indices. $E_I^{'} \in \mathbb{R}^{K_I \times d_{model}}$ is a new vector representation of the image modality after selecting K_I keyframes. The method of selecting keyframes for motion and object modalities is the same, while for the relationship modality, we replace E_{t-1} with the object encoding E_O, and the other steps are the same.

Finally, we concatenate the four feature modalities of the selected keyframes as the output E_{ORE} of ORE:

$$E_{ORE} = [E_I^{'}; E_M^{'}; E_O^{'}; E_R^{'}], \tag{16}$$

where Eq.(15) being a dynamic frame selection process, the number of frames selected in the same video may vary for each different modality, so $E_{ORE} \in \mathbb{R}^{(K_I+K_M+K_O+K_R) \times d_{model}}$.

3.3 Bidirectional secoder (BDD)

The Bidirectional decoder consists of a backward decoder and a forward decoder. The backward decoder generates captions in reverse order. Then, the forward decoder adds cross multi-head attention to integrate the context of the reverse caption.

3.3.1 Backward decoder (BD)

The Backward Decoder (BD) generates a reverse caption from right to left by integrating the output E_{ORE}. When the predicted word is the end marker $< \mathrm{S} >$, the prediction of the reverse caption ends. We express it as $\overleftarrow{C} = [s_1, \cdots, s_L, < \mathrm{S}s >]$. As shown in the BD of Fig. 3, we obtain the context $\overleftarrow{D}$ of the reverse caption as:

$$\overleftarrow{D} = \overleftarrow{D_L}, \tag{17}$$

where $\overleftarrow{D_L}$ is the hidden state obtained from the Linear layer when the BD generates the last word s_L.

3.3.2 Forward decoder (FD)

The Forward Decoder (FD) predicts a word from left to right by integrating the output E_{ORE}. Then, we integrate the context of the reverse caption generated by the BD. Compared with the BD, the FD adds cross multi-head attention, integrating the context $\overleftarrow{D}$, so that the forward decoder can take into account the context of the ground-truth caption every time it predicts the next word. When the predicted word is the end marker $< \mathrm{S} >$, the prediction of the forward caption is completed, which is expressed as $\overrightarrow{C} = [s_1, \cdots, s_T, < \mathrm{S} >]$.

3.3.3 Multimodal fusion attention

In the above two decoders, multimodal fusion attention assigns weights to different modalities through the E_{ORE} and $E_{t-1} \in \mathbb{R}^{(K_I+K_M+K_O+K_R) \times d_{model}}$ computed attention value $E_{t-1}^{'}$ to reduce redundancy between feature modalities, which is calculated as follows:

$$\begin{aligned} E_{t-1}^{'} &= \text{Attention}(E_{t-1}, E_{ORE}, E_{ORE}) \\ &= \text{softmax}\left(\frac{E_{t-1}E_{ORE}^T}{\sqrt{d_k}}\right)E_{ORE}. \end{aligned} \tag{18}$$

3.4 Training

During the training of MHAN, one of the most crucial aspects is the introduction of bidirectional decoding. To achieve this, we utilize a two-stage method that generates video captions with bidirectional encoding. Specifically, we first obtain the reverse caption through the backward decoder

and then integrate it into the forward decoder to generate a forward caption.

For bidirectional decoding, we introduce the typical cross-entropy losses $\mathcal{L}_{bd}$ and $\mathcal{L}_{fd}$ for the backward and forward decoders, respectively. Given a video V, its ground-truth caption $\overrightarrow{Y} = [y_1, \cdots, y_L]$ of length L, and the pseudo reverse caption $\overleftarrow{Y} = [y_1, \cdots, y_T]$ of length T from a training dataset $\mathcal{D}$, the loss formula of the bidirectional decoder $\mathcal{L}$ is as follows:

$$\mathcal{L} = (1 - \lambda)\mathcal{L}_{bd} + \lambda\mathcal{L}_{fd}, \tag{19}$$

where, $\lambda \in [0, 1]$ is a hyper-parameter used to balance the preferences between the two decoders. Next, we will explain the pseudo reverse caption and these two losses in detail.

3.4.1 Pseudo reverse caption

It is noteworthy that we generate pseudo reverse captions to mitigate the information leakage of the bidirectional decoder, which results in inconsistent lengths of $\overrightarrow{Y}$ and $\overleftarrow{Y}$. Specifically, we reverse all the ground-truth captions of a video to obtain the corresponding reverse captions. Then, we randomly shuffle these reverse captions to obtain the pseudo reverse captions that correspond to the ground-truth captions. As a result, the final reverse caption is not the inversion of the ground-truth caption.

3.4.2 Backward decoder loss

The backward decoder loss is defined as the negative log-likelihood to generate the reverse caption:

$$\mathcal{L}_{bd} = \sum_{(V,\overleftarrow{Y})\in\mathcal{D}} \sum_{t} (-\log p(y_t|V, y_1, \cdots, y_{t-1}; \theta_{ore}; \theta_{bd})), \tag{20}$$

where, y_t is the word being predicted, and θ_{ore} and θ_{bd} are the learnable parameters of the Objects and Relationships Encoder and the Backward Decoder, respectively.

3.4.3 Forward decoder loss

The traditional transformer decoder masks its input so that the decoder cannot refer to the following words of the ground-truth caption when predicting a word of the caption. Therefore, we utilize a bidirectional decoder that aims to obtain the following words from $\overleftarrow{D}$ generated by the backward decoder. Then, we integrate $\overleftarrow{D}$ into the word generation of the forward caption when the forward decoder generates the current word. The cross-entropy loss of the forward decoder is defined similarly to Eq. (20):

$$\mathcal{L}_{fd} = \sum_{(V,\overrightarrow{Y})\in\mathcal{D}} \sum_{l} (-\log p(y_l|V, y_1, \cdots, y_{l-1}; \theta_{bd}; \theta_{ore}; \theta_{fd}), \tag{21}$$

where θ_{fd} is the learnable parameters of the forward decoder.

4 Experiments

In this section, we will first describe four experimental settings: Datasets, Feature Extraction, Evaluation Metrics, and Parameter Settings. We will then introduce the performance comparison of our method.

4.1 Experimental settings

4.1.1 Datasets

Various experiments were conducted using the two most popular benchmark datasets to demonstrate the effectiveness of our proposed MHAN: the Microsoft Research Video Captioning Corpus (MSVD) [48] and the Microsoft Research Video to Text (MSR-VTT) [49].

MSVD: This dataset consists of 1970 YouTube video clips ranging from 10 to 25 s in length. Each clip describes a single activity and has approximately 40 English captions. As in [50, 51], the training, validation, and test sets contain 1200, 100, and 670 clips, respectively.

MSR-VTT: This dataset contains 10,000 video clips, each of which has 20 captions and a category tag annotated by 1327 workers from Amazon Mechanical Turk. Following the split settings in previous works [34], we allocated 6513, 497, and 2990 clips to the training, validation, and test sets, respectively.

4.1.2 Evaluation metrics

We have used standard automatic evaluation metrics to test the performance of our model, including BLEU [52], METEOR [53], CIDEr-D [54], and ROUGE-L [55]. BLEU and METEOR were initially designed for machine translation evaluation and are commonly used to assess the quality of machine-generated text. CIDEr-D is a recently introduced evaluation metric for image caption tasks, designed specifically for caption system evaluation, while ROUGE-L was used to evaluate the extracted text summarization proposed in 2004.

4.1.3 Feature extraction

To extract the image features from the videos in MSVD and MSR-VTT, we first sample the videos at 5 and 3 frames per second, respectively. We then feed the sampled results into the Inception-ResNet-V2 model. After that, we re-sample the videos at the rate of 25 and 15 frames per second, respectively. We extract the motion features using the I3D model, which takes as input 64 overlapping continuous frames and extracts features at intervals of 5 frames. The dimensions of the image and motion features are 2048-D and 1024-D, respectively. In addition, we use GLOVE [56] vectors of the auxiliary video category labels to help with feature encoding. We extract 50 and 60 frames from the visual features in MSVD and MSR-VTT at equal intervals, respectively. We then project both features to $d_{model} = 512$ in this paper.

4.1.4 Parameter settings

To obtain the objects and relationships between them, we set the confidence of Mask R-CNN to 0.7 and the minimum detection size to 224×224. We used a dataset of 613 non-repeated triples for TransE and represented the relationships using 300-dimensional GLOVE vectors. For the encoders and decoders used by MSVD and MSR-VTT, we respectively used 2/4 layers, 512/640 word embedding dimensions, 512/512 model dimensions, 2048/2048 hidden dimensions, and 8/10 attention heads per layer. However, we set the number of heads to 64 for the extended multi-head attention. During training, we set the hyperparameter λ used to balance the preferences between the two decoders to 0.6, and the number of training epochs to 25 and 15 for MSVD and MSR-VTT, respectively. We used Adam optimizer with a batch size of 32 and 16, and a learning rate of 1e-4 and 3e-5 for MSVD and MSR-VTT, respectively. To generate a better caption, we used beam search with a size of 5, and set the dropout rate to 0.15 and 0.1 for MSVD and MSR-VTT, respectively. All experiments were conducted on RTX3060 GPUs.

4.2 Performance comparison

The performance of our proposed MHAN will be compared with the following state-of-the-art methods: RecNet [34], PickNet [36], MARN [40], mg-LSTM [37], TDConvED [38] DRPN [35], ASL+BL [39], NACF [41], SHAN [44], MGRMP [42], GSB + CoSB [43], TVRD+OAG [46], ORG-TRL [45] and HMN [47]. Unless otherwise stated, we report the results directly from the original papers. The test results of the MSVD and MSR-VTT datasets are shown in Table 1. Our MHAN outperforms most state-of-the-art methods on both datasets.

Notably, our MHAN achieves significant improvement on the CIDEr-D metric, which is specifically designed for captioning tasks. For example, it shows a relative improvement of 1.5% on MSVD compared with HMN, and 1.3% on MSR-VTT compared with SHAN. This is because our model can focus on key frames and assign higher weights

Table 1 Experimental results on MSR-VTT and MSVD datasets are presented in this paper

Method	Features	RS	Dataset: MSVD				Dataset: MSR-VTT			
			B@4	M	R	C	B@4	M	R	C
RecNet [34]	IV4	✗	52.3	34.1	69.8	80.3	39.1	26.6	59.3	42.7
DRPN [35]	IV4	✗	*57.3*	34.3	72.0	86.4	39.5	27.7	61.0	49.2
PickNet [36]	R152	✗	52.3	33.3	69.6	76.5	41.3	27.7	59.8	44.1
mg-LSTM [37]	R152	✗	53.0	32.9	69.8	75.1	40.8	27.5	60.7	45.4
TDConvED [38]	R152	✗	53.3	33.8	–	76.4	39.5	27.5	–	42.8
ASL+BL [39]	R152	✗	50.4	34.2	70.4	73.7	38.4	27.2	59.7	44.1
MARN [40]	R101+3DR	✗	48.6	35.1	71.9	92.2	40.4	28.1	60.7	47.1
NACF [41]	R101+3DR	✗	55.6	36.2	–	96.3	42.0	28.7	–	51.4
MGRMP [42]	IRv2+3DR	✗	55.8	36.9	74.5	98.5	41.7	28.9	62.1	51.4
GSB+CoSB [43]	R101+3DR	✗	50.7	35.3	72.1	97.8	41.4	27.8	61.0	46.5
SHAN [44]	IRV2+I	✓	50.9	35.1	72.4	94.5	40.3	28.8	61.2	*54.1*
ORG-TRL [45]	IRV2+C+F	✓	54.3	36.4	73.9	95.2	**43.6**	28.8	62.1	50.9
TVRD+OAG [46]	IRV2+C+F	✓	50.5	34.5	71.7	84.3	*43.0*	28.7	*62.2*	51.8
HMN [47]	IRV2+C+F	✓	**59.2**	*37.7*	**75.1**	*104.0*	43.5	*29.0*	**62.7**	51.5
MHAN (Ours)	IRV2+I+M	✓	55.6	**38.5**	*74.9*	**105.6**	42.3	**29.8**	62.0	**54.8**

The best and suboptimal results of all methods are indicated by the bold and italic, respectively. ROUGE-L, BLEU4, METEOR, and CIDEr-D are abbreviated as R, B@4, M, and C, respectively. IV4, R152, R101, IRV2, 3DR, I, C, F and M denote Inception-V4, ResNet-152, ResNet-101, InceptionResNetV2, 3D ResNeXt-101, I3D, C3D, Faster-RCNN and Mask-RCNN, respectively. And the relationship (RS) column denotes whether this method uses relationships to enhance performance

Table 2 Ablation study of hierarchical attention network. (✓) indicates that the modal is added to the encoder, and (✗) indicates that it is not used

Image	Motion	Object	Relationship	B@4	M	R	C
✓	✓	✗	✗	53.2	37.0	73.8	99.5
✓	✓	✓	✗	53.4	37.9	74.6	100.6
✓	✓	✓	✓	55.8	38.5	75.1	101.1
MHAN (Ours)				55.6	38.5	74.9	105.6

to the more important modals through the generated words before generating the final caption. However, our model does not perform best on the BLEU metric, which is designed for machine translation and word-level-based matching. Additionally, Novicoca et al. [57] have shown that BLEU is not consistent with human judgement, while CIDEr-D is more in line with human writing habits [16]. Therefore, it is difficult for CIDEr-D and BLEU to improve the same model significantly. For example, while the BLEU4 of HMN is the highest in MSVD and MSR-VTT, the CIDEr-D score is not the highest.

5 Analysis

This section provides a comprehensive analysis of the MSVD dataset from various perspectives to enhance our understanding of our approach. We begin with a quantitative analysis, followed by a case study.

5.1 Quantitative analysis

We will first investigate the feasibility of our model after implementing a hierarchical attention network by conducting an ablation study. Then, we will discuss three comparative experiments as follows:

- Feasibility analysis of the extended multi-head attention,
- Selection of the number of keyframes,
- Necessity of the bidirectional decoder.

5.1.1 Ablation study of hierarchical attention network

For evaluating the effectiveness of our proposed MHAN, Table 2 shows that when we integrate more feature modalities into the encoder using a concatenation method of element-wise addition, the model's performance improves gradually. However, when our MHAN uses the hierarchical attention network to reduce the redundancy between modality frames and fuse four modalities through multimodal fusion attention, our proposed MHAN shows a significant improvement of 4.5% from 101.1 to 105.6 in CIDEr-D.

Table 3 Feasibility analysis of the extended and multi-head attention

Number of heads	B@4	M	R	C
8	54.6	37.9	74.4	104.5
16	53.0	38.1	74.2	100.8
32	53.6	38.1	74.5	101.8
64	55.6	38.5	74.9	105.6
128	54.7	38.1	74.4	101.1

Table 4 Selection of the number of keyframes

Number of keyframes	B@4	M	R	C
20	52.5	37.6	74.6	101.0
30	53.7	38.0	74.3	101.9
40	55.7	38.5	75.4	104.5
50	54.9	38.7	75.4	103.8
Dynamic selection	55.6	38.5	74.9	105.6

5.1.2 Feasibility analysis of the extended and multi-head attention

In Section 3.2.1, we investigate the closer correlations between smaller video clips rather than frames through the use of extended multi-head attention. As shown in Table 3, when increasing the number of heads to 64, the encoder can capture sufficient relevance between the video clips, resulting in a significant improvement in performance. However, if the number of heads is increased to 128, the model is prone to overfitting. Therefore, we ultimately adopted 64 heads.

5.1.3 Selection of the number of keyframes

In Eq.(15), we automatically obtain the number of keyframes by focusing on the proportion of non-zero elements generated after attention value calculation. This method of dynamically obtaining the number of keyframes of each modality is better than fixing this value. As shown in Table 4, when we select a small number of keyframes, such as 20 or 30, the video information obtained is insufficient, and the performance is inferior.

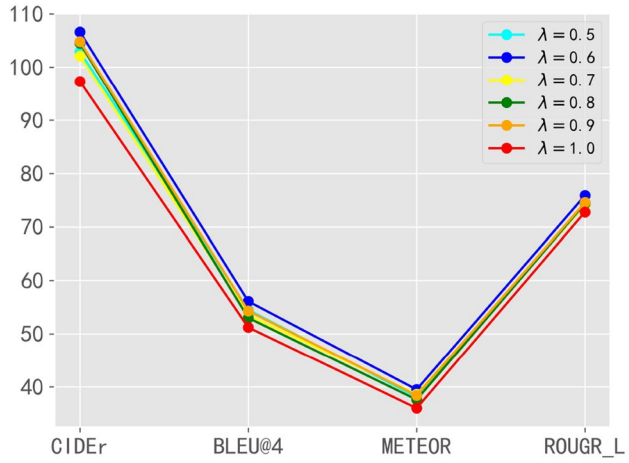

Fig. 4 Necessity of the bidirectional decoder. When $\lambda = 1$, the bidirectional decoder degenerates into a unidirectional decoder

When we use all 50 frames as keyframes, the performance is not as good as when we use 40 keyframes due to the redundancy between frames. Finally, when we use the method of automatically obtaining the number of keyframes, the model achieves the highest CIDEr-D score of 105.6, demonstrating the effectiveness of our approach.

5.1.4 Necessity of the bidirectional decoder

We introduce the hyperparameter λ to balance the preferences of the forward and backward decoders during training (see Eq.(19)). In Fig. 4, we trained MHAN on the MSVD dataset using different values of λ to determine its optimal value. When $\lambda = 1$ (red line), the bidirectional decoder degenerates into a unidirectional decoder, where the backward decoder does not participate in training, resulting in the worst performance of the model. When $0.5 \leq \lambda < 1$, there is a significant improvement in all four metrics compared to $\lambda = 1$, indicating that the backward decoder affects the generation of captions in the forward decoder and proving the effectiveness of the bidirectional decoder. When we set $\lambda = 0.6$ (blue line), a stable balance is achieved between the two decoders, resulting in the best performance across all four metrics.

5.2 Case study

Although we can evaluate our model and summarize its performance through the evaluation mechanism described in Section 4.2, the scores may not directly reflect the quality of the sentences generated by MHAN. Therefore, we present a few visualized examples of captions generated by fusing modalities through element-wise addition (EWA), our proposed MHAN, and human annotation (Ground Truth) in Fig. 5 to analyze and verify the findings in the Performance Comparison section.

As shown in Fig. 5, MHAN accurately predicted the behavior of the cat and the man in the *No.1* and *No.2* videos, respectively, while EWA used "playing" and ended the prediction with an incorrect action, "drinking." The model with object detection is expected to use specific words to describe the video. However, in *No.3*, EWA used the abstract word "a vegetable" instead of "an onion." When EWA tried to use the specific word "a potato," the described object was "carrots" in *No.4*. Additionally, we found that EWA failed to fully learn the video content when using a concatenation method of element-wise addition to fuse these modalities, which may lead to inaccurate predictions. For example, in the EWA of *No.5*, "cleaning" was replaced by an inaccurate action, "standing." Even in the *No.6* video, where a man and a monkey were fighting (practicing karate), EWA ignored the other subject, "a man."

In summary, our proposed MHAN effectively integrates multiple modalities into a bidirectional decoding transformer through a hierarchical attention network, accurately describing the subjects' actions in the video. The captions generated by MHAN are also more consistent with human language habits. This further explains why our MHAN achieved a higher CIDEr-D but only a minor increase in BLEU, as described in Section 4.2.

6 Conclusions

This paper proposes a Multimodal-enhanced Hierarchical Attention Network (MHAN) based on a bidirectional decoding transformer for video captioning. MHAN not only considers the context when generating the description through a bidirectional decoder but also strengthens the vector representation of modalities, selects keyframes for each modality, and reduces redundancy between modalities to effectively fuse multiple modalities through the hierarchical attention network. Additionally, experiments on two mainstream benchmark datasets, MSVD and MSR-VTT, demonstrate the effectiveness of the proposed method, which achieves state-of-the-art performance in significant metrics. The captions generated by MHAN are also more in line with human language habits.

Fig. 5 Visualized examples of captions generated by fusing modalities through element-wise addition (EWA), our proposed MHAN, and human annotation (Ground Truth)

Acknowledgements The authors would like to express their gratitude to the anonymous reviewers for their valuable comments, which have helped to improve the quality of the paper. This research has been partially supported by the National Natural Science Foundation of China (Grant No. 61877031) and the Jiangxi Normal University Graduate Innovation Fund (Grant No. YJS2022029).

Author contributions YC completed the main code writing and experiments and the rest of the people participated in the compilation of part of the code and the design of the experiment. MZ and YC wrote the main manuscript text. All authors reviewed the manuscript.

Data availability This paper uses two common datasets in the field of video captioning, MSVD and MSR-VTT. Data availability is not applicable to this article as no new data were created or analyzed in this study.

Declarations

Conflict of interest The authors declare no competing interests.

References

1. Donahue, J., Anne Hendricks, L., Guadarrama, S., Rohrbach, M., Venugopalan, S., Saenko, K., Darrell, T.: Long-term recurrent convolutional networks for visual recognition and description. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 2625-2634) (2015)
2. Venugopalan, S., Xu, H., Donahue, J., Rohrbach, M., Mooney, R., Saenko, K.: Translating videos to natural language using deep recurrent neural networks. arXiv preprint arXiv:1412.4729 (2014)
3. Yao, L., Torabi, A., Cho, K., Ballas, N., Pal, C., Larochelle, H., Courville, A.: Describing videos by exploiting temporal structure. In *Proceedings of the IEEE international conference on computer vision* (pp. 4507-4515) (2015)
4. Xu, J., Yao, T., Zhang, Y., Mei, T.: Learning multimodal attention LSTM networks for video captioning. In *Proceedings of the 25th ACM international conference on Multimedia* (pp. 537-545) (2017, October)
5. Singh, A., Singh, T.D., Bandyopadhyay, S.: Attention based video captioning framework for hindi. Multimedia Syst. **28**(1), 195–207 (2022)
6. Zhong, M., Zhang, H., Xiong, H., Chen, Y., Wang, M., Zhou, X.: Kgvideo: A Video Captioning Method Based on Object Detection and Knowledge Graph. *Available at SSRN 4017055*
7. Zhong, M., Zhang, H., Wang, Y., Xiong, H.: BiTransformer: augmenting semantic context in video captioning via bidirectional decoder. Mach. Vis. Appl. **33**(5), 1–9 (2022)
8. Yang, B., Zhang, T., Zou, Y.: (2022) CLIP Meets Video Captioning: Concept-Aware Representation Learning Does Matter. In: Pattern Recognition and Computer Vision: 5th Chinese Conference. PRCV,: Shenzhen, China, November 4?7, 2022, Proceedings, Part I, pp. 368–381. Springer International Publishing, Cham (2022)

9. Hori, C., Hori, T., Lee, T. Y., Zhang, Z., Harsham, B., Hershey, J. R., ... Sumi, K.: Attention-based multimodal fusion for video description. In *Proceedings of the IEEE international conference on computer vision* (pp. 4193-4202) (2017)
10. Ji, S., Xu, W., Yang, M., Yu, K.: 3D convolutional neural networks for human action recognition. IEEE Trans. Pattern Anal. Mach. Intell. **35**(1), 221–231 (2012)
11. Aafaq, N., Akhtar, N., Liu, W., Gilani, S. Z., & Mian, A.: Spatio-temporal dynamics and semantic attribute enriched visual encoding for video captioning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 12487-12496) (2019)
12. Anderson, P., He, X., Buehler, C., Teney, D., Johnson, M., Gould, S., Zhang, L.: Bottom-up and top-down attention for image captioning and visual question answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 6077-6086) (2018)
13. Lee, J.Y.: Deep multimodal embedding for video captioning. Multimedia Tools Appl. **78**(22), 31793–31805 (2019)
14. Liu, A.A., Xu, N., Wong, Y., Li, J., Su, Y.T., Kankanhalli, M.: Hierarchical & multimodal video captioning: Discovering and transferring multimodal knowledge for vision to language. Comput. Vis. Image Underst. **163**, 113–125 (2017)
15. Jin, Q., Chen, J., Chen, S., Xiong, Y., & Hauptmann, A.: Describing videos using multi-modal fusion. In *Proceedings of the 24th ACM international conference on Multimedia* (pp. 1087-1091) (2016, October)
16. Jiang, Y.: Multi-feature fusion for video captioning. Int. J. Comput. Appl. **181**(48), 975–8887 (2019)
17. Li, L., Zhang, Y., Tang, S., Xie, L., Li, X., Tian, Q.: Adaptive spatial location with balanced loss for video captioning. IEEE Trans. Circuits Syst. Video Technol. **32**(1), 17–30 (2020)
18. Huang, Y., Cai, Q., Xu, S., Chen, J.: Xlanv model with adaptively multi-modality feature fusing for video captioning. In *Proceedings of the 28th ACM International Conference on Multimedia* (pp. 4600-4604) (2020, October)
19. Yan, Z., Chen, Y., Song, J., Zhu, J.: Multimodal feature fusion based on object relation for video captioning. CAAI Trans. Intell. Technol. **8**(1), 247–259 (2023)
20. Krizhevsky, A., Sutskever, I., Hinton, G. E.: ImageNet classification with deep convolutional neural networks. Commun ACM **60**(6), 84–90 (2017)
21. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural Comput. **9**(8), 1735–1780 (1997)
22. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... Polosukhin, I.: Attention is all you need. In *Proceedings of the 31st International Conference on Neural Information Processing Systems* (pp. 6000–6010) (2017)
23. Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., ... Sutskever, I.: Learning transferable visual models from natural language supervision. In *International conference on machine learning* (pp. 8748-8763). PMLR (2021, July)
24. Szegedy, C., Ioffe, S., Vanhoucke, V., Alemi, A. A.: Inception-v4, inception-resnet and the impact of residual connections on learning. In *Thirty-first AAAI conference on artificial intelligence* (2017, February)
25. Carreira, J., Zisserman, A.: Quo vadis, action recognition? a new model and the kinetics dataset. In *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 6299-6308) (2017)
26. Kay, W., Carreira, J., Simonyan, K., Zhang, B., Hillier, C., Vijayanarasimhan, S., ... Zisserman, A.: The kinetics human action video dataset. arXiv preprint arXiv:1705.06950 (2017)
27. He, K., Gkioxari, G., Dollr, P., Girshick, R.: Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision* (pp. 2961–2969) (2017)
28. Bordes, A., Usunier, N., Garcia-Durán, A., Weston, J., & Yakhnenko, O.: Translating embeddings for modeling multi-relational data. In *Proceedings of the 26th International Conference on Neural Information Processing Systems-Volume 2* (pp. 2787–2795) (2013)
29. Han, X., Cao, S., Lv, X., Lin, Y., Liu, Z., Sun, M., Li, J.: Openke: An open toolkit for knowledge embedding. In *Proceedings of the 2018 conference on empirical methods in natural language processing: system demonstrations* (pp. 139-144) (2018, November)
30. Lin, T. Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., ... Zitnick, C. L.: Microsoft coco: Common objects in context. In *European conference on computer vision* (pp. 740-755). Springer, Cham (2014, September)
31. Lin, T. Y., Dollr, P., Girshick, R., He, K., Hariharan, B., Belongie, S.: Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 2117-2125) (2017)
32. Ren, S., He, K., Girshick, R., & Sun, J.: Faster R-CNN: towards Real-Time Object Detection with Region Proposal Networks. IEEE Trans Pattern Anal Mach Intell **39**(6), 1137–1149 (2017)
33. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781 (2013)
34. Wang, B., Ma, L., Zhang, W., Liu, W.: Reconstruction network for video captioning. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 7622-7631) (2018)
35. Xu, W., Yu, J., Miao, Z., Wan, L., Tian, Y., Ji, Q.: Deep reinforcement polishing network for video captioning. IEEE Trans. Multimedia **23**, 1772–1784 (2020)
36. Chen, Y., Wang, S., Zhang, W., Huang, Q.: Less is more: Picking informative frames for video captioning. In *Proceedings of the European conference on computer vision (ECCV)* (pp. 358-373) (2018)
37. Xu, N., Liu, A.A., Nie, W., Su, Y.: Multi-guiding long short-term memory for video captioning. Multimedia Syst. **25**, 663–672 (2019)
38. Chen, J., Pan, Y., Li, Y., Yao, T., Chao, H., Mei, T.: Temporal deformable convolutional encoder-decoder networks for video captioning. In *Proceedings of the AAAI conference on artificial intelligence* (Vol. 33, No. 01, pp. 8167-8174) (2019, July)
39. Li, L., Zhang, Y., Tang, S., Xie, L., Li, X., Tian, Q.: Adaptive spatial location with balanced loss for video captioning. IEEE Trans. Circuits Syst. Video Technol. **32**(1), 17–30 (2022)
40. Wenjie, Pei., Jiyuan, Zhang., Xiangrong, Wang., Lei, Ke., Xiaoyong, Shen., Yu-Wing, Tai.: Memory-attended recurrentnetwork for video captioning. In CVPR, pages 8347?8356, (2019)
41. Yang, B., Zou, Y., Liu, F., Zhang, C.: Non-autoregressive coarse-to-fine video captioning. Proc. AAAI Conf. Artif. Intell. **35**(4), 3119–3127 (2021). https://doi.org/10.1609/aaai.v35i4.16421
42. Chen, S., & Jiang, Y. G.: Motion guided region message passing for video captioning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision* (pp. 1543-1552) (2021)
43. Vaidya, J., Subramaniam, A., Mittal, A.: Co-Segmentation Aided Two-Stream Architecture for Video Captioning. In *Proceedings of the IEEE/CVF Win ter Conference on Applications of Computer Vision* (pp. 2774-2784) (2022)
44. Deng, J., Li, L., Zhang, B., Wang, S., Zha, Z., Huang, Q.: Syntax-guided hierarchical attention network for video captioning. IEEE Trans. Circuits Syst. Video Technol. **32**(2), 880–892 (2022). https://doi.org/10.1109/TCSVT.2021.3063423
45. Zhang, Z., Shi, Y., Yuan, C., Li, B., Wang, P., Hu, W., Zha, Z. J.: Object relational graph with teacher-recommended learning for video captioning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 13278-13288) (2020)

46. Wu, B., Niu, G., Yu, J., Xiao, X., Zhang, J., Wu, H.: Towards Knowledge-aware Video Captioning via Transitive Visual Relationship Detection. *IEEE Transactions on Circuits and Systems for Video Technology.* (2022)
47. Ye, H., Li, G., Qi, Y., Wang, S., Huang, Q., Yang, M.: Hierarchical Modular Network for Video Captioning. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) **2022**, 17918–17927 (2022)
48. Chen, D., Dolan, W. B.: Collecting highly parallel data for paraphrase evaluation. In *Proceedings of the 49th annual meeting of the association for computational linguistics: human language technologies* (pp. 190-200) (2011, June)
49. Xu, J., Mei, T., Yao, T., Rui, Y.: Msr-vtt: A large video description dataset for bridging video and language. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 5288-5296) (2016)
50. Pei, W., Zhang, J., Wang, X., Ke, L., Shen, X., Tai, Y. W.: Memory-attended recurrent network for video captioning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 8347-8356) (2019)
51. Pan, B., Cai, H., Huang, D. A., Lee, K. H., Gaidon, A., Adeli, E., Niebles, J.C.: Spatio-temporal graph for video captioning with knowledge distillation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 10870-10879) (2020)
52. Papineni, K., Roukos, S., Ward, T., Zhu, W. J.: Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics* (pp. 311-318) (2002, July)
53. Banerjee, S., Lavie, A.: 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* (pp. 65-72) (2005, June)
54. Vedantam, R., Lawrence Zitnick, C., Parikh, D.: Cider: Consensus-based image description evaluation. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 4566-4575) (2015)
55. Lin, C. Y.: Rouge: a package for automatic evaluation of summaries. In *Text summarization branches out* (pp. 74–81) (2004)
56. Pennington, J., Socher, R., Manning, C. D.: Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)* (pp. 1532-1543) (2014)
57. Novikova, J., Du?ek, O., Curry, A. C., Rieser, V.: Why We Need New Evaluation Metrics for NLG. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing* (pp. 2241-2252) (2017)