Singapore Management University

Institutional Knowledge at Singapore Management University

Research Collection School Of Computing and Information Systems | School of Computing and Information Systems

1-2022

Action-centric relation transformer network for video question answering

Jipeng ZHANG
University of Electronic Science and Technology of China

Jie SHAO
University of Electronic Science and Technology of China

Rui CAO
Singapore Management University, ruicao.2020@phdcs.smu.edu.sg

Lianli GAO
University of Electronic Science and Technology of China

Xing XU
University of Electronic Science and Technology of China

See next page for additional authors

Follow this and additional works at: https://ink.library.smu.edu.sg/sis_research

 Part of the Broadcast and Video Studies Commons, Databases and Information Systems Commons, and the Numerical Analysis and Scientific Computing Commons

Citation

ZHANG, Jipeng; SHAO, Jie; CAO, Rui; GAO, Lianli; XU, Xing; and SHEN, Heng Tao. Action-centric relation transformer network for video question answering. (2022). *IEEE Transactions on Circuits and Systems for Video Technology*. 32, (1), 63-74.
Available at: https://ink.library.smu.edu.sg/sis_research/6020

This Journal Article is brought to you for free and open access by the School of Computing and Information Systems at Institutional Knowledge at Singapore Management University. It has been accepted for inclusion in Research Collection School Of Computing and Information Systems by an authorized administrator of Institutional Knowledge at Singapore Management University. For more information, please email cherylds@smu.edu.sg.

Author

Jipeng ZHANG, Jie SHAO, Rui CAO, Lianli GAO, Xing XU, and Heng Tao SHEN

This journal article is available at Institutional Knowledge at Singapore Management University: https://ink.library.smu.edu.sg/sis_research/6020

2020, Early access, DOI: 10.1109/TCSVT.2020.3048440

Action-Centric Relation Transformer Network for Video Question Answering

Jipeng Zhang, Jie Shao, Rui Cao, Lianli Gao, Xing Xu, and Heng Tao Shen

***Abstract*—Video question answering (VideoQA) has emerged as a popular research topic in recent years. Enormous efforts have been devoted to developing more effective fusion strategies and better intra-modal feature preparation. To explore these issues further, we identify two key problems. (1) Current works take almost no account of introducing action of interest in video representation. Additionally, there exists insufficient labeling data on where the action of interest is in many datasets. However, questions in VideoQA are usually action-centric. (2) Frame-to-frame relations, which can provide useful temporal attributes (e.g., state transition, action counting), lack relevant research. Based on these observations, we propose an action-centric relation transformer network (ACRTransformer) for VideoQA and make two significant improvements. (1) We explicitly consider the action recognition problem and present a visual feature encoding technique, action-based encoding (ABE), to emphasize the frames with high actionness probabilities (the probability that the frame has actions). (2) We better exploit the interplays between temporal frames using a relation transformer network (RTransformer). Experiments on popular benchmark datasets in VideoQA clearly establish our superiority over previous state-of-the-art models. Code could be found at https://github.com/op-multimodal/ACRTransformer.**

***Index Terms*—Video question answering, video representation, temporal action detection, multi-modal reasoning, relation reasoning.**

I. Introduction

DEVELOPING computer systems to automatically answer questions according to the content of an image or a video has attracted a large amount of attention [1], [2], [3], [4], [5]. The task is challenging as it requires the comprehension of visual and textual semantic information, as well as their complex dependencies. Research efforts on visual question answering fall into two categories: image question answering (ImageQA) and video question answering (VideoQA). In this paper, our main focus lies in VideoQA. Compared with ImageQA, VideoQA is more difficult. Its visual modality extends from one image to a long sequence of images and associated questions place more emphasis on the temporal dimension of videos. This additional temporal dimension consists of varieties of actions with their localization, classes and relations, which are referred to as temporal attributes. Here we mainly focus on action-centric issues [3], [6], which require temporal reasoning over atomic motions in videos to generate correct answers.

Question: What does the hockey player do after throw a hockey stick?

Answer: take off mask

Fig. 1. To solve this problem, we need to first recognize these action instances: the player throws a hockey stick, the player takes off mask, the player outspreads arms, etc. Then, we need to determine their temporal relationships correctly and focus on the action following "the player throws a hockey stick".

Illustrated in Figure 1 is an example concerning state transition. Correctly answering the question requires accurate recognition of atomic motions and good comprehension over the interplays among these actions. This is a common case in VideoQA and requires shallow semantic understanding. However, current VideoQA models cannot solve these action-centric questions very well.

Issues. One reason is that previous methods lack consideration over fine-grained video representation based on the action of interest, that is, action proposals in videos. More accurately, action proposals mean spans showing the beginnings and endings of actions. We argue that action proposals should be integrated into video representation and enhance visual encoding. Most existing models feed video frames to a pre-trained 2D convolutional neural network (CNN) and video clips through 3D convolutional networks or flow CNN networks, and extract features from a specific layer of the network. Features of frames are commonly combined by concatenation or summation to compute the representation vector of the whole video. This technique under-exploits the temporal attributes and leaves out action proposals in videos. However, the requirement of the VideoQA system is that the model should be able to recognize a large variety of actions and their interplays. Thus, it is necessary to introduce action proposals into the VideoQA system.

A challenging part is that the performance on questions that focus more on static attributes (e.g., objects, colors, locations) should not be compromised. This means we need an effective feature encoding method to achieve a trade-off between dynamic and static parts.

J. Zhang, J. Shao, L. Gao, X. Xu, and H. T. Shen are with the Center for Future Media, School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu, 611731, China. E-mail: zhangjipeng20@std.uestc.edu.cn, {shaojie,lianli.gao,xing.xu,shenhengtao}@uestc.edu.cn. J. Shao and H. T. Shen are also with Sichuan Artificial Intelligence Research Institute, Yibin, 644000, China.

R. Cao is with the School of Information Systems, Singapore Management University, 178902, Singapore. Email: caorui0503@gmail.com.

Corresponding author: Jie Shao.

This work is supported by National Natural Science Foundation of China (No. 61832001 and No. 61672133) and Sichuan Science and Technology Program (No. 2019YFG0535).

Recalling ImageQA, similar problems have been found and have been circumvented by using object-level visual features [7], which are computed based on pre-trained Faster R-CNN. Similarly, if we try to simply borrow this idea in the video domain, action detection methods can be utilized to produce several action features for each detected action. However, it is not wise to choose this solution in VideoQA right now. Not only are related annotations in many existing VideoQA datasets lacking but also videos have much richer information that cannot be totally expressed in few features corresponding to action spans. Therefore, we need a different schema for video representation in VideoQA.

Another reason is that the frame-between relation features are overlooked in current models. Machines are required to capture abundant information of every single frame as well as their interplays. In most cases, it is the difference between frames that matters in our action-centric task. Considering the scenes of recognizing state transition, we need to localize when the state changes. More specifically, it is the difference between the frame of finishing previous action and the start frame of current action that indicates this state transition. For the repetition count task, the action of interest is performed many times, and frame-between changes in the start and end time matter in correctly answering this kind of question. Thus, we argue that only extracting frame-wise features underutilizes video features and adding frame-between relation features can do us a favor in capturing frame-level visual signals scattered in the given video.

Solutions. To address the issues mentioned above, we first propose a visual encoding technique, action-based encoding (ABE), to compute video representations enriched with temporal dynamics information, that is, action proposals of videos. We first concatenate 2D CNN and flow CNN features, where ResNet [8] and BN-Inception network [9] are used as the 2D and flow CNNs, respectively. Boundary sensitive network (BSN) [10], which can locate high actionness probability locations in a given video to detect action, is used to grab spans of actions in videos. Instead of computing several features for each action span to represent the whole video, these action spans are utilized as a form of actionness sampling in our approach. Finally, integrating them into the concatenated features mentioned above helps to produce strong action-centric video representations as well as preserve most of the static features.

In addition, we introduce a relatively simple relation transformer network (RTransformer) to produce novel relation features with the encoded video representations. Compared with many existing methods [3], [11], [12], RTransformer differs in incorporating an efficient multi-head structure primed for exploiting frame-between interplays. We believe these interplays matter in temporal reasoning over video and grabbing accurate patterns. The proposed method deals with this requirement effectively by making full use of various relations lied in the temporal dimension of videos.

We evaluate our model on the large-scale TGIF-QA dataset [3] and ActivityNet-QA [6]. The experimental results clearly establish the superiority of our new framework when compared with many previous methods. The accuracy of our model outperforms the best competitor on ActivityNet-QA. On TGIF-QA, we achieve the best accuracy in transition. Additionally, on the other three tasks, our model also achieves comparable results. In addition, some additional experiments further prove the effectiveness of our model on repeating action, state transition and repetition count.

To summarize, we make the following contributions:

- We propose a visual encoding method that effectively embeds temporal attributes into the video features. The newly generated video features contain the detected action information that matters in this action-centric task.
- We propose a relation-based video model aiming at better utilization of temporal attributes scattered in frames and their interplay.
- Experimental results show that our proposed framework is remarkably better than many existing models.

The remainder of this paper is organized as follows. First, Section II reviews the related work. Next, Section III introduces the basic architecture of our proposed model. Section IV provides details for our two novel changes in visual encoding and feature extraction. In Section V, we compare our model with existing methods and conduct an ablation study to show the effect of each important component. Finally, Section VI makes a conclusion.

II. Related Work

In this section, we review relevant works in three aspects: video question answering, visual representation methods in vision and language tasks, and relation network.

A. Video Question Answering

Recently, VideoQA has sparked public interest. VideoQA requires the capability of understanding visual concepts. However, there is an additional temporal dimension compared with ImageQA and we need to consider visual content in a long sequence of images. Thus, VideoQA is more challenging. To evaluate the ability of understanding contents in multiple frames for different models, numerous datasets have been proposed [3], [13], [5], [4]. Various types of video clips (e.g., movie clips, game videos, cartoons, social media, etc.) have been collected in these datasets, and questions are specifically tailored to fit the video contents.

Early methods mainly focus on fusing multi-modal features, grabbing various information in the temporal dimension and more detailed spatio-temporal modeling. For more literature about fusion, please refer to the recent survey paper [14]. Except for element-wise addition and multiplication, heterogeneous memory [15] and graph networks [16] have been used to fuse features. As for grabbing temporal information, some methods choose to feed features of each frame into recurrent neural networks [3], [17]. In [18], a co-memory network with motion flow features is adopted to extract more expressive temporal information. Transformer [19] is also exploited to encode the video clips [12] and the complex dependencies of frames or objects [20]. In [11], a structured segment component is used to boost the performance. Additionally, the temporal attention mechanism has been widely used in

VideoQA [3], [11], [17], [21], [22], [23], [24], to identify the important frames according to their relevance to questions. Moreover, the adversarial method [25] is also explored in improving temporal understanding. Recently, more researchers focus on spatio-temporal reasoning [26], [27], [28], [29]. More specifically, there are three most important factors: spatial information of each frame, temporal connections across different frames and relations of objects/humans of different frames. In this view, our work proposes more efficient solutions for temporal connections across different frames and the relation of objects/humans of different frames.

B. Visual Representation Methods in Vision and Language Tasks

In ImageQA, the development of visual feature preparation techniques plays an important role in improving the performance of models. At the very beginning, grid features extracted by pre-trained CNN such as VGG [30] and ResNet [8] are often chosen as the visual features for image-level visual and language systems [31]. Later, the region/object level bottom-up and top-down features [7] help improve many systems. They almost become a substitute for grid features in the vision and language tasks. Recently, as grid features can help simplify the image-level visual and language system, some works [32] have proposed to develop grid features again with good results achieved.

For VideoQA, frame-level video features [3], [18] are usually extracted by 3D-CNN or flow-CNN networks. Similar to grid features in the image domain, these frame-level features are easy to deal with. However, current methods lack consideration over action related information. Some recent works [6], [33] have also attempted to use detected span of actions to obtain visual features. They propose to use temporal action detection techniques to obtain action proposals first and then compute an action feature for each action proposal. These action-level features can achieve good performance given sufficient annotations. However, it is very difficult to transfer the action feature methods to datasets without sufficient annotations. Additionally, this approach may do harm to some questions focused on static attributes (e.g., objects, colors, locations).

Our aim is to achieve a trade-off between action-level and frame-level features as well as partly solve the transferring issue for cases without sufficient labels. Specifically, our model also first detects action proposals. Later, we transform the proposals into actionness importance weights to compute the new ABE features, which keeps the shape of the original frame features.

C. Relation Network

Different from the task of recognizing, classifying and grounding, relational reasoning calls for the ability to clarify the relationships among a series of unstructured inputs. Aiming at reasoning about relations between objects, various methods have been proposed such as symbolic approaches, static learning and graph-based reasoning. The development of deep learning has inspired people to propose an end-to-end trainable relation network. In [34], a differentiable relation network is proposed to capture interactions. Its relational reasoning module exploits multi-layer perceptrons as a composite function for obtaining relationships between objects and is later integrated with the ImageQA framework. Experiments are conducted on CLEVR [35], which is an ImageQA dataset specifically designed for object-based relational reasoning. The superior results suggest that neural networks have good relational reasoning capability. Motivated by previous work, recurrent relational networks [36] is introduced to do multi-step relational reasoning. For research on relation network, our RTransformer actually extends the relation network framework with the multi-head structure.

III. Basic Temporal Attention Model

In this section, we briefly review a basic temporal attention model, where the following input information is given: a video v, a question sentence q, and an answer phrase a. The goal is to output either an answer phrase (open-ended questions) or a group of scores (multiple-choice questions). In addition, answer a is optional and provided only by multiple-choice questions. The full pipeline consists of three stages: (1) feature preprocessing, (2) feature extraction, and (3) feature fusion and answer generation.

A. Feature Preprocessing

Given a video, we first extract all frames according to the frame per second (FPS) provided by the video file. Then, frames are processed by ResNet-152 [8] to generate frame-level feature $V_{frame} = \{f_1, f_2, \cdots, f_T\}, f_i \in R^{d_i}$. Meanwhile, we employ a flow CNN network [37] to obtain sequence-level feature $V_{sequence} = \{s_1, s_2, \cdots, s_T\}, s_i \in R^{d_s}$. We set the sequence length as $T = 35$. If the number of frames is smaller than T, we pad zero features. If the number of frames is larger than T, we sample with the same intervals. Frame-level features V_{frame} and sequence-level features $V_{sequence}$ are combined by concatenation to compute the representation of the whole video $V_{combine}$ $(vc_i = [f_i; s_i])$, $vc_i \in R^{d_f}$.

For the input question and answer, we consider them as sequences of words and represent each word as a 300D vector using the GloVe word embedding [38]. We denote the question as $Q = \{q_1, q_2, \cdots, q_N\}$. For multiple choice questions, the answer candidates are characterized as $A_{candidate} = \{ac_1, ac_2, ac_3, ac_4, ac_5\}$. $ac_i = \{aci_1, aci_2, \cdots, aci_{Mi}\}$ denotes a single answer candidate with each aoi_j corresponding to a word. N is the sequence length of the question while Mi is the sequence length for the i-th answer phrase. Then, we combine Q with each answer candidate ac_i to compute the final feature set: $A_i = \{q_1, q_2, \cdots, q_N, aci_1, aci_2, \cdots, aci_{Mi}\}$. Thus, we obtain the answer feature sets, $A = \{A_1, A_2, A_3, A_4, A_5\}$, $A_i = \{ai_1, ai_2, \cdots, ai_{N+Mi}\}$.

B. Feature Extraction

Next, we adopt recurrent neural networks to perform feature extraction for processed textual and visual features mentioned above.

We encode text features of question Q using text encoding LSTM:

$$h_n^q = LSTM_q(q_n, h_{n-1}^q), Q_o = h_N^q. \tag{1}$$

Finally, the question is represented as a vector Q_o with the dimension of d_q.

We employ another LSTM with an additional attention mechanism to encode video features $V_{combine}$ with question feature Q_o. We begin with encoding video features with an LSTM.

$$h_t^v = LSTM_{ve}(ve_t, h_{t-1}^v). \tag{2}$$

Here, we obtain a new set of visual features $H_v = \{h_1^v, h_2^v, \cdots, h_T^v\}$. Next, we perform *Attention* operation on H_v regarding question feature Q_o:

$$av_i = W_{av} ReLU(W_{qv} Q_o + W_v h_i^v), \tag{3}$$

$$\gamma_i = \frac{exp(av_i)}{\sum_{k=1}^{T} exp(av_k)}, \tag{4}$$

$$VC = \sum_{j=1}^{T} \gamma_j h_j^v, \tag{5}$$

where av_i is the attention weight of the i-th new visual feature vector. First, we project the input visual vectors and question vector into the same dimension. Next, we copy the question vector T times, and perform element-wise addition between new visual and question feature vectors. Next, a single layer neural network is applied to generate the unnormalized attention distribution. Then, we normalize the sequence of γ_j using a softmax function and compute V_o as the weighted average of H_v. Here, V_o is the representation of the whole video.

C. Feature Fusion and Answer Generation

We fuse visual feature V_o and question feature Q_o into joint feature J by element-wise addition. When facing multiple-choice questions with answer candidates, we consider these candidates as complementary to the question. Therefore, for each candidate, the answer encoder takes the fusion feature J to initialize the answer LSTM. After that, the input answer features $A = \{A_1, A_2, A_3, A_4, A_5\}$ are fed to the answer LSTM and generate answer features $AF = \{AF_1, AF_2, AF_3, AF_4, AF_5\}$. We formulate this encoding process as:

$$h_t^{ai} = LSTM_{ai}(ai_t, h_{t-1}^{ai}), AF_i = h_{N+Mi}. \tag{6}$$

After the multiple-choice encoder, the i-th answer candidate is represented as a feature AF_i. It is worth noting that the i-th answer candidates is fed to i-th LSTM, i.e., $LSTM_{ai}$.

In regard to answer generation, following previous work [3], [18], we treat four sub-tasks (repeating action, state transition, repetition count and frame QA) in the TGIF-QA dataset as three different types of decoders: multiple-choice, open-ended numbers and open-ended words. ActivityNet-QA belongs to the form of open-ended words.

Multiple choice: Since repeating action and state transition tasks both belong to multiple-choice questions, we define a linear regression function that takes as input the encoded answer feature AF. We compute a real-valued score for each answer candidate in terms of this function,

$$s_i = W_s AF_i, \tag{7}$$

where W_s is the model parameter. We train the decoder by optimizing the binary cross-entropy loss.

Open-ended numbers: For the Count task, we also define a linear regression function that takes the final fusion representation J as the input and outputs an integer-valued answer,

$$s = [W_n J + b_n], \tag{8}$$

where $[.]$ means rounding. We adopt $l2$ loss between label value and the predicted value to train the model.

Open-ended words: For the frame question answering task, we treat it as a classification problem. A linear function takes the final fused representation followed by softmax,

$$o = softmax(W_w J + b_w), \tag{9}$$

where W_w is the model parameter. Cross-entropy loss is used in this task.

We separately train the model for the four tasks with corresponding answer decoders and loss functions. Additionally, the evaluation is performed independently.

IV. ACRTransformer

The proposed action-centric relation transformer network (ACRTransformer), as illustrated in Figure 2, follows the basic temporal attention model processing paradigm for VideoQA but features two novel changes (Section IV-A and Section IV-B).

A. Action-Based Encoding

In this section, we propose a new video encoding mechanism, action-based encoding (ABE), to embed temporal attributes into video representations. The main idea is illustrated in Figure 3.

Since the VideoQA task is action-centric and requires temporal reasoning, more focus should be placed on the temporal dimension of video. Hence, we argue that designing an effective visual encoding mechanism emphasizing frames with high actionness probabilities for VideoQA is of great importance.

In visual encoding, CNNs are usually made as default choices of encoding schemes in VideoQA, which are tremendously successful in capturing effective representations from visual data. However, the existing methods mainly focus on tailoring model architectures, while the literature mostly chooses primitive methods of using CNN features. They usually directly utilize 2D, 3D and flow CNN features or their concatenations for visual encoding. In this line of research, there is a lack of exploration toward effective methods.

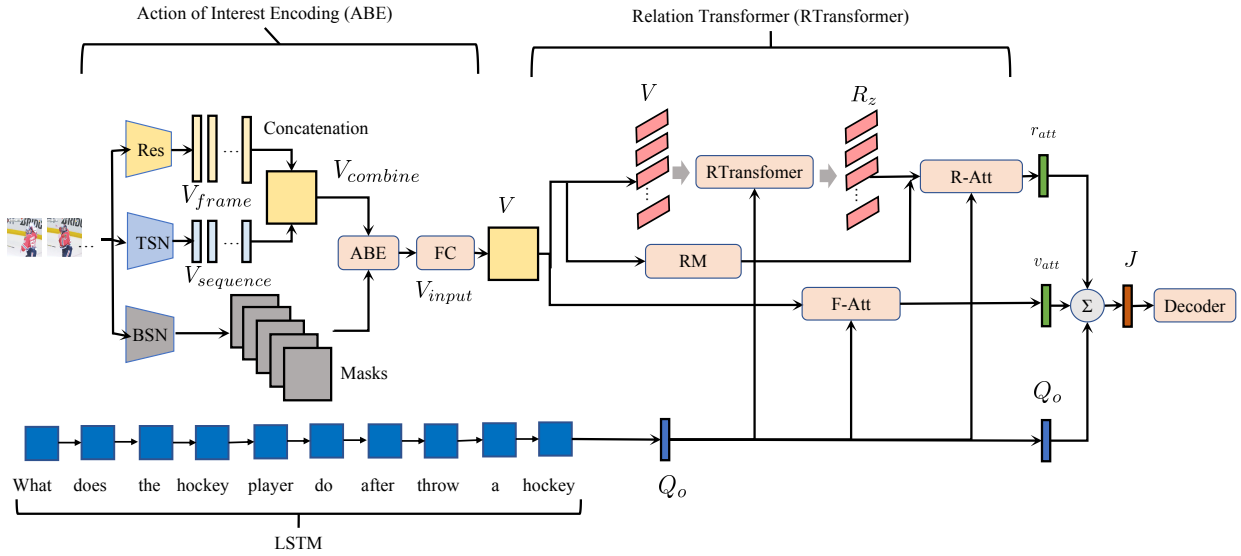


Fig. 2. Overview of the proposed model. V_{frame} and $V_{sequence}$ represent the feature extracted by Resnet and flow-CNN. These two group of features are fused together as $V_{combine}$. Action-based encoding (ABE) module handles $V_{combine}$ and action masks generated by BSN, and computes the representations V enriched with temporal attributes. RTransformer is utilized to grab relations across all pairs of frames, conditioned on the question embedding Q_o produced by an LSTM. After obtaining the relation features R_z, all these relations are integrated with the relation attention module (R-Att). Besides, the enriched video representations V are fused together with another attention module (F-Att). Finally, these features are fused into a single feature J with element-wise addition and are fed to the answer decoder to answer the question.

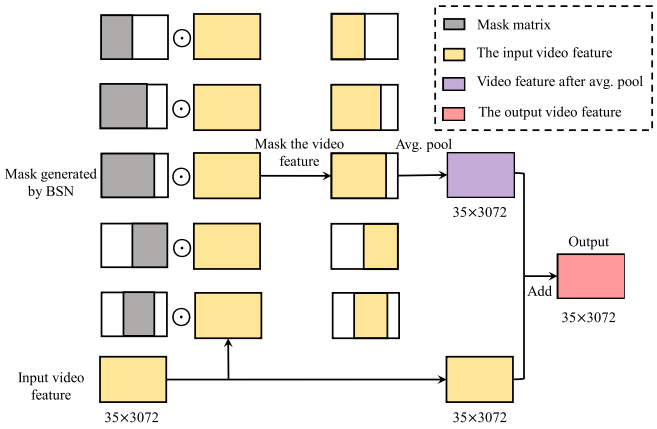


Fig. 3. Action-based encoding module. Masks are constructed using action spans derived by the boundary sensitive network (BSN). The input video features integrate all these masks to compute output features enriched with temporal attributes.

Our key idea for dealing with action recognition is to turn spans of actions generated by a temporal action detection model into masks that indicate ranges of high probabilities where actions exist. Next, we combine these masks with primary visual features and add all these masked features together to "zoom in" frames with higher probabilities of having actions and "zoom out" frames with lower probabilities of having actions .

In this work, BSN [10], which specializes in localizing high actionness frames in the given video, is chosen as our implementation of the temporal action detector. First, the demanding visual features are prepared according to [10]. Next, the prepared features are fed to the provided pre-trained BSN. Thus, action spans as well as their confidence scores are obtained. In the following, they are transformed into video feature masks.

Practically, we first choose the action spans with top-M high scores. Thus, we can obtain the start and end times of these spans: $\{(t_{s1}, t_{e1}), (t_{s2}, t_{e2}), \cdots, (t_{sM}, t_{eM})\}$. Then, the start time and end time are transformed into start and end frames $\{(tf_{s1}, tf_{e1}), (tf_{s2}, tf_{e2}), \cdots, (tf_{sM}, tf_{eM})\}$. Frames in these given spans have a higher actionness probability, which means we should focus more on these frames. By using these spans, we can generate M mask matrices that share the shape with $V_{combine}$. Since our visual feature contains T frames, the start and end frames of each action can be located.

Taking (tf_{s1}, tf_{e1}) as an example, we can obtain a subset of $V_{combine}$, $V_{combine1} = \{vc_{tf_{s1}}, \cdots, vc_{tfe1}\}$, which only contains visual features related to this detected action. In practice, for each action span, we define a zero matrix $Mask_i$ with the same shape of $V_{combine}$ and assign one to rows that locate in corresponding span of action. To make frame features locating in these spans matter more, element-wise multiplication is performed between the primary visual features and the mask matrices.

$$\begin{aligned} BSN_{f_1} &= V_{combine} \odot Mask_1, \\ BSN_{f_2} &= V_{combine} \odot Mask_2, \\ &\cdots \\ BSN_{f_M} &= V_{combine} \odot Mask_M, \end{aligned} \tag{10}$$

where each $BSN_{f_i} \in R^{T \times d_f}$. Hence, each BSN_{f_i} only retains frames in action spans. Later, these features are combined by average pooling:

$$BSN_f = \frac{1}{M} \sum_{i}^{M} BSN_{f_i}. \tag{11}$$

With the aim of avoiding losing information besides detected actions, this encoded feature is fused with the primary video representation feature $V_{combine}$ by element-wise addition:

$$V_{input} = BSN_f + V_{combine}. \tag{12}$$

Thus, we obtain the encoded representation $V_{input} \in R^{T \times d_f}$ of the given video.

B. Relation Transformer Network

In this section, a relation transformer (RTransformer) is introduced to exploit frame-between features as well as global information of the video.

Recall that in video feature extraction, previous works employed tailored LSTM or Transformer to encode video features into useful cues representing the whole video. They usually utilize an attention module, which focuses on attending to the relevant video frame features, to embed the visual features into one combined feature vector. However, we claim that the model should better exploit frame-level temporal reasoning and focus more on the relations between frames. The frame-between interplays are of great importance. It enables more accurate identification of state transitions of humans or objects and counting the repetition times of actions.

Consider a case in which a female model stops walking on the catwalk and pivots around to change direction after moving a long distance forward. The change in the model state is included in the frame-to-frame change. More precisely, the difference between the frame in which she finishes walking forward and the frame in which she begins pivoting around means a lot in questions about state transition. The ability to grab these frame-to-frame interplays is emphasized in our model. In our RTransformer, the capacity of reasoning about frame-between relations is included in the architecture without the need to be learned.

The goal of our RTransformer, as shown in Figure 4, is to generate a relation-aware video representation vector r_{att}.

Relation Module. $V_{input} = \{vi_1, vi_2, \cdots, vi_T\}$, $vi_i \in R^t$, the visual features produced by ABE, is the input sequence containing frame-wise features. Q_o is the final question feature computed by question encoder. The first step is to compute the fixed-dimensional representation vectors $V = \{v_1, v_2, \cdots, v_T\}$, $v_i \in R^{d_i}$ based on input video feature set V by a fully connected layer:

$$v_i = W_p vi_i + b_p, \tag{13}$$

where i ranges from 1 to T. Each of v_i is treated as a basic cell of RTransformer. We explore how many frames to choose in Table VII and finally choose to compute relations between two neighboring frames. We also argue that the existence and meaning of a frame-to-frame relation should be question dependent. For example, if a question asks about the state of a specific person, the relations belonging to other people are probably irrelevant. Therefore, we add the question feature Q_o into the input of RTransformer.

Given the frame-wise visual feature $V = \{v_1, v_2, \cdots, v_T\}$ and question feature Q_o, the relation feature between two consecutive frames is computed by:

$$r_i = W_r([v_i, v_{i+1}, \cdots, v_F, Q_o]) + b_r, \tag{14}$$

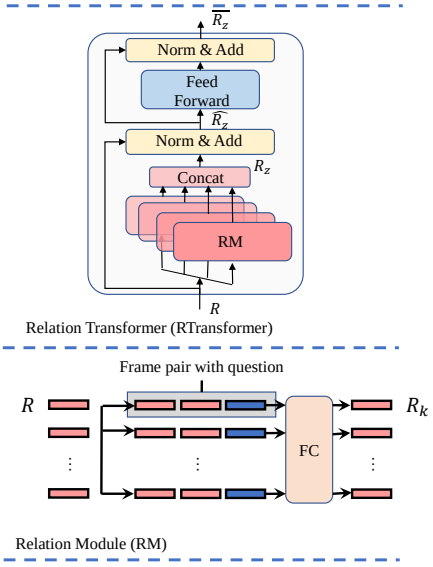


Fig. 4. Relation Transformer module. By introducing a multi-head architecture, the input relation features are enhanced to be aware of multiple patterns.

where r_i is the i-th relation feature containing relation information between v_i and v_{i+1}. [ ] indicates concatenation. For T visual features, we can obtain $T-(F-1)$ corresponding relation features $R = \{r_1, r_2, \cdots, r_{T-(F-1)}\}, r_i \in R^{d_r}$. Here, F means the number of frames considered in the relation module. The design philosophy behind this module is to constrain the functional form so that it focuses on frame-between relations.

RTransformer. Our relation transformer (RTransformer) utilizes the above relation module to learn the relation features. To identify complex dependencies among frames, we propose a K-head relation setup. This has many things in common with the Transformer model [19], where K separate relation networks are used and concatenated before a residual connection is applied.

First, we use a relation module to compute the initial relations features $R = \{r_1, r_2, \cdots, r_{T-1}\}, r_i \in R^{d_r}$.

Next, we input this R into the K-head architecture. Specifically, a single relation module has its parameter $W_{rk} \in R^{d_r \times d_k}$, where $d_k = d/K$. The exact learning of each head relation module has been mentioned above.

For each relation module head $\{RelationModule_k\}_{k=1}^{K}$, we perform learning of the relation module in parallel, yielding d_k-dimensional output values. The output values are concatenated and projected, resulting in the final values:

$$R_z = \bigoplus_{k=1}^{K} R_k, \tag{15}$$

$$R_k = RelationModule_k(V, Q_o). \tag{16}$$

Here, $\|$ denotes the concatenation of the K relation module heads.

The RTransformer then augments this K-head relation network with a feed-forward network, layer-norm layer, and

TABLE I
STATISTICS OF TGIF-QA. "NUMBER OF EXAMPLES" REPRESENTS THE NUMBER OF EXAMPLES OF DIFFERENT TASKS IN BOTH TRAINING AND TEST SET.

	Action	Trans.	Frame	Count	Total
Distribution	13.8%	35.7%	32.1%	18.4%	-
Number of Examples	22,749	58,936	53,083	30,379	165,147

TABLE II
STATISTICS OF ACTIVITYNET-QA. "NUMBER OF EXAMPLES" REPRESENTS THE NUMBER OF EXAMPLES OF DIFFERENT TASKS IN BOTH TRAINING AND TEST SET.

	Motion	Spat.	Temp.	Free	Total
Distribution	10.0%	10.0%	10.0%	70.0%	-
Number of Examples	5,800	5,800	5,800	40,600	58,000

residual connection:

$$\hat{R_z} = R_z + LayerNorm(R_z), \tag{17}$$

$$\overline{R_z} = \hat{R}_z + LayerNorm(FFN(\hat{R}_z)), \tag{18}$$

where $FFN(x)$ is a two-layer feed-forward network with a $ReLU$ function between layers:

$$FFN(x) = max(0, xW_{f1} + b_{f1})W_{f2} + b_{f2}. \tag{19}$$

The resulting relation representations $\overline{R_z}$ combine many different relation patterns. The final relation feature is:

$$R_f = \overline{R_z} + R, \tag{20}$$

To better capture the relationship between the question and these relation features, they are fed to an attention module that shares the paradigm with attention modules in Section III.

$$r_{att} = Attention_r(R_f, Q_o), \tag{21}$$

where r_{att} is the representation of all the frame-between relation features.

Considering frame-wise information as an important complement, we compute a frame-wise video feature conditioned on the question feature Q_o. Taking the projected video feature set $V = \{v_1, v_2, \cdots, v_T\}$ and question feature Q_o as input, the attention mechanism described in Section III is used to compute the frame-wise video feature v_{att}:

$$v_{att} = Attention(V, Q_o), \tag{22}$$

where v_{att} is the frame-wise video representation feature.

Finally, we combine the frame-wise feature v_{att}, relation feature r_{att} and question feature Q_o by adding them together, and then the combined feature J is obtained.

V. EXPERIMENTS

A. Datasets

As ActivityNet-QA [6] and TGIF-QA [3] are equipped with human-annotated and action-centric QA-pairs, we assume TGIF-QA and ActivityNet-QA are more suitable and thus evaluate our model on these two benchmarks.

ActivityNet-QA: ActivityNet-QA, which consists of 58K QA pairs on 5.8K videos, has large-scale open-ended QA pairs and relatively long videos. These videos come from the ActivityNet dataset [39], which contains 20K videos representing 200 action classes. Each video is annotated with 10 QA pairs using crowdsourcing to finally obtain 58K QA pairs. The detailed distribution of different types of questions is shown in Table II. There four types of questions in ActivityNet-QA:

Motion: Model needs to correctly recognize the action referred by the question so that it can give the right answer.

Spatial Relationship: Questions in this tasks only require spatial reasoning within one static frame.

Temporal Relationship: This task asks for exploring temporal relationship of objects across different frames.

Free: All the questions that are hard to classify into existing categories belong to this type.

TGIF-QA: TGIF-QA [3], a large dataset containing 72K animated GIFs and 165K template-based question-answer pairs, focuses more on action-centric temporal reasoning. The detailed distribution of different types of questions is shown in Table II. There are four types of tasks in TGIF-QA:

Action: It is defined as a multiple-choice question about recognizing actions repeating certain times in a video.

Transition (Trans.): This task asks us to determine specific state transition in a video.

FrameQA: Questions in this task can be answered from one of the frames in a video.

Count: In this task, it asks to give the number of repetitions of an action in the question. It can be regarded as an open-ended question with answers ranging from 0 to 10.

B. Evaluation Criteria

On ActivityNet-QA, as suggested by [6], the performance is evaluated using 2 common evaluation criteria, accuracy and Wu-Palmer metric score (WUPS) [40].

On TGIF-QA, following [3], for Action, Trans. and FrameQA, accuracy is adopted to estimate the predicted answer. In regard to the Count task, the mean square error (MSE) loss is used to evaluate the difference between the ground-truth answer and the predicted answer.

C. Parameter Setting

Given a GIF or video, we first extract all frames according to the frame per second (FPS) provided by the video file. For 2D CNN features, we apply ResNet-152 [8] to every frame to obtain output from "pool5" ($\in R^{2048}$). For flow CNN features, we first apply dense flow networks [41] to extract raw optical flow frames. These optical flow images are fed to the modified BN-Inception [9]. Through this process, we can obtain flow CNN features ($\in R^{1024}$).

For semantic representation, each word in the sentences is embedded into the same 300D with GloVe embedding [38].

During training, the optimizer from [19] is used in our model. The strategy of warm-up is also utilized during the training and the number of iterations is set as 2000. The training epoch is set as 30 and the size of minibatch is set as 128. To alleviate the issue of overfitting, dropout is applied to all the fully connected layers and word embedding layer. All implementations above are under the PyTorch library.

TABLE III
COMPARISON WITH THE STATE-OF-THE-ART METHODS ON THE ACTIVITYNET-QA DATASET.

Model	Accuracy (%)	WUPS@0.9 (%)	WUPS@0.0 (%)
E-VQA [6]	25.1	29.3	53.5
E-MN [6]	27.1	31.5	55.9
E-SA [6]	31.8	34.9	56.4
VQA-HMAL [25]	30.2	38.8	53.3
CAN [33]	35.4	40.5	60
ACRTransformer	**37.28**	**41.37**	**62.08**

TABLE IV
COMPARISON WITH THE STATE-OF-THE-ART METHODS ON THE TGIF-QA DATASET.

Model	Action	Trans.	Frame	Count
Yu et al. [42]	56.1	64.0	39.6	5.13
ST-TP (R+F) [3]	62.9	69.4	49.5	4.32
Co-memory (R+F) [18]	68.2	74.3	51.5	4.10
PSAC (R) [12]	70.4	76.9	55.7	4.27
STA (R) [11]	72.3	79.0	56.6	4.25
HME (R+C) [15]	73.9	77.8	53.8	4.02
Aggre-Diversity (R) [22]	72.0	80.7	58.2	4.24
Multi-Inter (R) [26]	72.7	80.9	57.1	4.17
HGA (R+C) [16]	75.4	81.0	55.1	4.09
Loc-GCN (R) [28]	74.3	81.1	56.3	3.95
QueST (R) [27]	**75.9**	81.0	**59.7**	4.19
HCRN [20]	75.0	81.4	55.9	**3.82**
ACRTransformer	75.81	**81.61**	57.68	4.08

Our implementation code is available at https://github.com/op-multimodal/ACRTransformer.

D. Comparison with State-of-the-Art Methods

We compare our ACRTransformer with recent deep learning models designed for VideoQA. Most of these methods first extract question and video representations by various feature extractors and then fuse them into one joint representation for answer decoder.

The experimental results are reported in Table III and Table IV. We denote our approach as ACRTransformer. We can observe from both Table III and Table IV that our model performs significantly better than the existing methods. On ActivityNet-QA, we improve the accuracy from 35.4 to 37.28 and obtain the corresponding increases in the WUPS metric. On TGIF-QA, our model achieves the best result in Trans. and obtains comparable performance on the other three tasks.

We suggest such significant progress results from the effectiveness of two modules in ACRTransformer. On the one hand, the process in ABE adds abundant temporal attributes to video representations as well as keeps static frame features. Existing methods merely use outputs from 2D, 3D or flow CNN as representations but ignore where the frames of interest are in temporal reasoning. Although temporal attention can alleviate this problem, with the lack of the ground-truth of which frame contains an action, it may be inaccurate. ABE clearly benefits questions concentrating on temporal reasoning because action attributes are extracted more accurately. On the other hand, the proposed ACRTransformer clarifies relationships between frames. Therefore, this implementation facilitates state transition comprehension, which is proven by the remarkable increase in accuracy in ActivityNet-QA.

TABLE V
ACCURACY FOR DIFFERENT INPUT FEATURES AND DIFFERENT MODELS WITH THE SAME INPUT FEATURES ON ACTIVITYNET-QA.

Model	Accuracy (%)	WUPS@0.9 (%)	WUPS@0.0 (%)
ACRTransformer (R+F)	35.99	40.12	61.41
ACRTransformer (Hard)	30.57	58.80	34.78
ACRTransformer (ABE)	37.28	41.37	62.08

TABLE VI
ACCURACY FOR DIFFERENT INPUT FEATURES AND DIFFERENT MODELS WITH THE SAME INPUT FEATURES ON TGIF-QA.

Model	Action	Trans.	Frame	Count
ACRTransformer (R+C)	70.67	78.88	53.46	4.17
ACRTransformer (R+C BSN)	72.47	79.44	52.76	4.13
ACRTransformer (R+F)	73.79	81.1	57.22	4.11
ACRTransformer (ABE)	75.81	81.61	57.68	4.08
ACRTransformer (Hard)	73.39	79.97	55.07	4.28
ST-TP (R+F)	62.9	69.4	49.5	4.32
ST-TP (ABE)	65.30	72.53	52.22	4.26
PSAC (R)	70.4	76.9	55.7	4.27
PSAC (ABE)	70.8	78.35	55.9	4.23

E. Ablation Studies

Our proposed ACRTransformer consists of several modules. To estimate the effectiveness of each module, an ablation study is conducted. The experimental results are shown in Table V, Table VI, Table VII and Table VIII.

Effectiveness of action-based encoding. We implement our basic LSTM model (LSTM) and attended frame-level relation network (ACRTransformer) with four different types of features. Here, "(R+C)" denotes ResNet-152 features and C3D features, "(R+F)" denotes ResNet-152 features and optical flow features, "(R+C BSN)" denotes ResNet-152 feature and C3D features with enriched temporal attributes, and "(R+F BSN)" denotes ResNet-152 and optical flow features with enriched action attributes, where $+$ means concatenation. It is notable that this "(R+F BSN)" equals our ABE features. As shown in Table V and Table VI, in comparison with "(R+F)" and "(Hard)", our model can achieve higher accuracy with our "(R+F BSN)" features. We believe the reason is that ABE performs better at identifying visual-related signals.

On ActivityNet-QA, as shown in Table V, our ABE features apparently achieve higher accuracy in all three metrics.

On TGIF-QA, we feed our ABE features to the ST-TP model in [3] and PSAC in [12]. The boosting of accuracies indicates the effectiveness of our ABE features. In other words, ABE can benefit the VideoQA model substantially.

Effectiveness of ACRTransformer. The results in the first two blocks of Table VI validate the power of our proposed ACRTransformer in modeling temporal relations. We can observe that our ACRTransformer-based models outperform LSTM-based models with all four kinds of features. We believe the reason is that our ACRTransformer module better utilizes features of frame-level action attributes.

Range of frames in relational reasoning. In relation modules of RTransformer, the relation within a distinct number of frames, which can be interpreted as different ranges of relational reasoning, can be varied. When the range of relational reasoning varies from 2 to 4, the result is provided

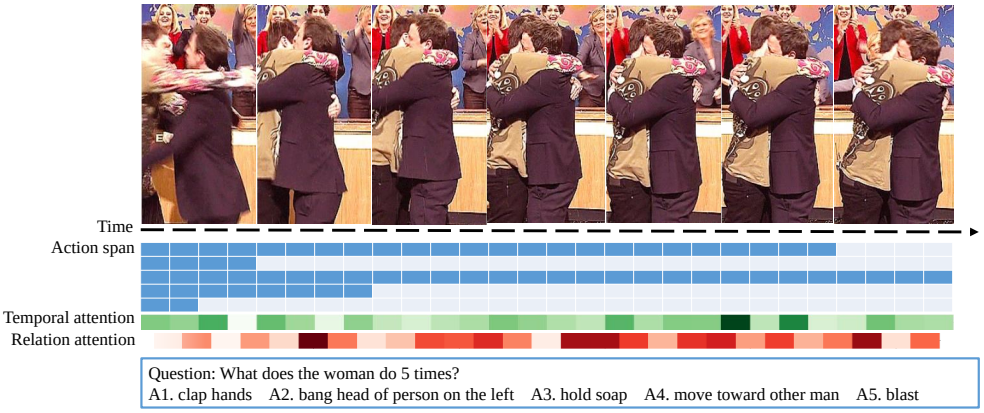


Fig. 5. **An example from Action** illustrates action span, temporal attention and relation attention. Temporal attention emphasizes frames that include the action "clapping hands". Relation attention weights are high when question-related state changes (the woman begins and stops clapping hands).

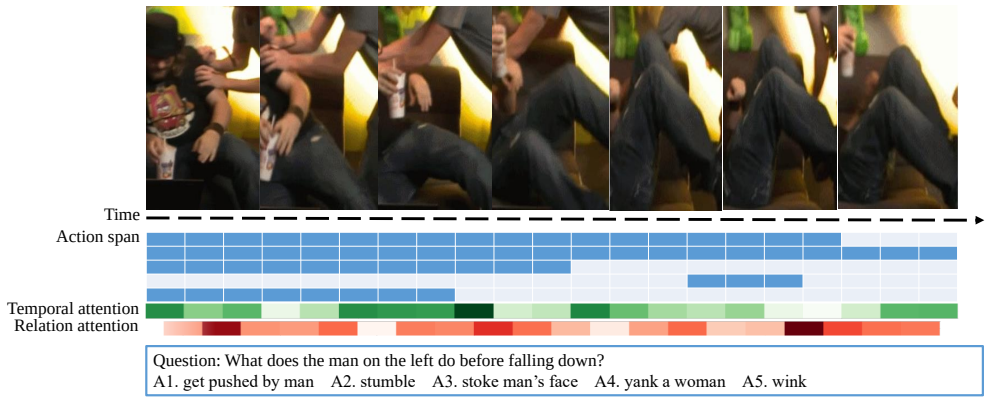


Fig. 6. **An example from Trans.** illustrates action span, temporal attention and relation attention. Temporal attention emphasizes frames which include the actions "falling down" and the action before it. Relation attention weights are high when question-related states change (the man begins to fall down), for illustration.

TABLE VII
ACCURACY FOR INCREASING RELATION FRAMES.

Model	Accuracy (%)	WUPS@0.9 (%)	WUPS@0.0 (%)
2-frame	37.28	41.37	62.08
3-frame	36.27	40.44	61.53
4-frame	36.17	40.31	61.29

TABLE VIII
ACCURACY FOR INCREASING RELATION TRANSFORMER HEADS.

Model	Accuracy (%)	WUPS@0.9 (%)	WUPS@0.0 (%)
1-head	36.73	40.62	61.80
2-head	37.28	41.37	62.08
4-head	36.33	40.50	61.56

in Table VII. We can see that using two frames gives the best performance.

Range of relational transformer heads. We explore the number of heads in the multi-head architecture. The range of the number of heads varies in $1, 2, 4$ and the corresponding results are shown in Table VIII. The results show that using 2 heads performs best.

TABLE IX
THE DETAILED ACCURACIES OF DIFFERENT TYPE QUESTIONS IN ACTIVITYNET-QA.

Model	Motion	Spat.	Temp.	Free	All
E-VQA	2.5	6.6	1.4	34.4	25.1
E-MN	3.0	8.1	1.6	36.9	27.1
E-SA	12.5	14.4	2.5	41.2	41.8
VQA-HMAL	-	-	-	-	30.2
CAN	21.1	17.3	3.6	44.5	35.4
Ours	**32.12**	**19.0**	**3.62**	**45.37**	**37.28**

TABLE X
ACCURACY FOR DIFFERENT DIFFERENT FRAME-TO-FRAME RELATION FEATURES EXTRACTION ON THE TGIF-QA AND ACTIVITYNET-QA DATASET.

Model	Action	Trans.	Frame	Count	Anet
RTransformer	75.81	81.61	57.68	4.08	37.28
Relation-RNN [36]	74.41	80.3	54.9	4.23	35.44
GCN	75.4	81.11	55.33	4.08	38.45

F. Detailed Accuracies for Questions about Actions

Table IX and Table IV give the detailed results for different type of questions in ActivityNet-QA and TGIF-QA. More exactly, these categories in datasets reflect the performance in questions related with action :"Action" and "Trans" in

TABLE XI
ACCURACY FOR DIFFERENT DIFFERENT VIDEO FEATURE PROCESSING METHODS ON THE TGIF-QA AND ACTIVITYNET-QA DATASET.

Model	Action	Trans.	Frame	Count	Anet
ACRTransformer	75.81	81.61	57.68	4.08	37.28
2D-TAN [43]	73.09	80.25	54.55	4.1	34.78
BMN [44]	75.2	80.95	55.28	4.08	37.81
self-attention	73.79	79.89	58.13	4.17	36.45

TGIF-QA, "Motion" and "Temp." in ActivityNet-QA. Our model still outperform other methods steadily, which indicates its effectiveness in questions action of interest need to be considered.

G. Further Exploration for Different Methods

As suggested by the reviewer, we have tried several different methods in modeling frame-to-frame relation and video feature extraction. The experimental results are shown in Table X and Table XI.

Frame-to-frame Relation: We have tried to replace the proposed RTransformer with Relation-RNN [36] and graph convolution network (GCN). Experiment results are shown in Table X.

Relation-RNN performs poor compared with our model. However, it is interesting to see GCN achieved impressive results in ActivityNet-QA while our model still outperform GCN in TGIF-QA. The reason should be that videos in ActivityNet-QA have richer actions, which means that the dependencies among different actions should be more complex. GCN may work better in this more complicated case.

Video Feature Extraction: We have used several different methods, like 2D-TAN [43], BMN [44] and self-attention [19] in processing video features.

Compared with 2D-TAN, our VideoQA datasets does not have natural language queries annotation with their corresponding moment location. Besides, we design ABE to extract frames with high possibility to have actions, which is different from "retrieving a specific moment from an untrimmed video by a query sentence". Thus, it is unreasonable to implement 2D-TAN directly in our model. However, we do agree that 2D map is an interesting idea to perform actionness sampling. Thus, we implement a new model. We first compute 2D relation map , and then add them together to get the score for each frame. Later, we use softmax to normalize these scores to sampling weights and multiply them with the feature of each frame. Here is the experiment results.

BMN is proposed to locate the action moments, which is similar to BSN used in our methods. Therefore, it can be utilized to recognize action spans in our model and replace BSN.

In experiment of self-attention, we implement self-attention layer and get the 2D attention map first. Next, we add them together in column to compute the score for each frame. Finally, these scores are normalized to get an attention vector with softmax function. This attention vector is used to replace BSN masks in ABE part.

Here, according to the results shown in Table XI, 2D-TAN does not achieve good results. We believe it is because the model does not provide effective prior. Besides, BMN performs really good and even outperform our model in ActivityNet-QA. We suggest that the reason is BMN can give more accurate action location than BSN in ActivityNet-QA, which has related annotations to show the start and end of action spans. However, in TGIF-QA, as there are no related annotations, BMN can not distinguish BSN in action detection. We believe BMN may need modifying some structures to get better results. We will keep exploring this in the future. In addition, self-attention performs bad in questions cares more about actions. It is interesting to observe that self-attention achieve great results in Frame, which means self-attention may be good at modeling static attributes.

H. Visual Verification

In this section, we visualize the spans of actions, temporal attention and relation attention. As illustrated in Figure 5 and Figure 6, we separately choose two examples from Trans. and Action. By visualizing the action proposal information, we can conclude that frames with higher actionness probabilities will be emphasized. Temporal attention refines such action attribute level attention by attending to question-related frames. For relation attention, places containing state transitions gain higher weights.

VI. CONCLUSION

We introduce two novel mechanisms, action-based encoding (ABE) and relation transformer (RTransformer), to help improve the VideoQA system. In addition to harnessing frame-level feature power in the static part, our method places more emphasis on dynamic temporal attributes and embeds them into primary video features with ABE. Additionally, instead of applying a recurrent neural network or Transformer to extract video representation vectors, we utilize an RTransformer to capture temporal attributes. Experiments are conducted on two large VideoQA benchmarks, TGIF-QA and ActivityNet-QA. The results show significant improvements over a collection of competitive baselines, verifying the value of our model. Although our model was originally designed for VideoQA, our method is task-agnostic. Our future direction is to explore how to use ABE and RTransformer on other video and language tasks.

REFERENCES

[1] S. Antol, A. Agrawal, J. Lu, M. Mitchell, D. Batra, C. L. Zitnick, and D. Parikh, "VQA: visual question answering," in *2015 IEEE International Conference on Computer Vision, ICCV 2015, Santiago, Chile, December 7-13, 2015*, 2015, pp. 2425–2433.

[2] Y. Goyal, T. Khot, D. Summers-Stay, D. Batra, and D. Parikh, "Making the V in VQA matter: Elevating the role of image understanding in visual question answering," in *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, 2017, pp. 6325–6334.

[3] Y. Jang, Y. Song, Y. Yu, Y. Kim, and G. Kim, "TGIF-QA: toward spatio-temporal reasoning in visual question answering," in *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, 2017, pp. 1359–1367.

[4] M. Tapaswi, Y. Zhu, R. Stiefelhagen, A. Torralba, R. Urtasun, and S. Fidler, "Movieqa: Understanding stories in movies through question-answering," in *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, 2016, pp. 4631–4640.

[5] J. Lei, L. Yu, M. Bansal, and T. L. Berg, "TVQA: localized, compositional video question answering," in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, 2018, pp. 1369–1379.

[6] Z. Yu, D. Xu, J. Yu, T. Yu, Z. Zhao, Y. Zhuang, and D. Tao, "Activitynet-qa: A dataset for understanding complex web videos via question answering," in *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019*, 2019, pp. 9127–9134.

[7] P. Anderson, X. He, C. Buehler, D. Teney, M. Johnson, S. Gould, and L. Zhang, "Bottom-up and top-down attention for image captioning and visual question answering," in *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, 2018, pp. 6077–6086.

[8] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, 2016, pp. 770–778.

[9] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015*, 2015, pp. 448–456.

[10] T. Lin, X. Zhao, H. Su, C. Wang, and M. Yang, "BSN: boundary sensitive network for temporal action proposal generation," in *Computer Vision - ECCV 2018 - 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part IV*. Springer, 2018, pp. 3–21.

[11] L. Gao, P. Zeng, J. Song, Y. Li, W. Liu, T. Mei, and H. T. Shen, "Structured two-stream attention network for video question answering," in *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019*, 2019, pp. 6391–6398.

[12] X. Li, J. Song, L. Gao, X. Liu, W. Huang, X. He, and C. Gan, "Beyond rnns: Positional self-attention with co-attention for video question answering," in *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019*, 2019, pp. 8658–8665.

[13] K. Kim, M. Heo, S. Choi, and B. Zhang, "Deepstory: Video story QA by deep embedded memory networks," in *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI 2017, Melbourne, Australia, August 19-25, 2017*, 2017, pp. 2016–2022.

[14] D. Zhang, R. Cao, and S. Wu, "Information fusion in visual question answering: A survey," *Inf. Fusion*, vol. 52, pp. 268–280, 2019.

[15] C. Fan, X. Zhang, S. Zhang, W. Wang, C. Zhang, and H. Huang, "Heterogeneous memory enhanced multimodal attention model for video question answering," in *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, 2019, pp. 1999–2007.

[16] P. Jiang and Y. Han, "Reasoning with heterogeneous graph alignment for video question answering," in *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, 2020, pp. 11 109–11 116.

[17] Z. Zhao, Z. Zhang, S. Xiao, Z. Yu, J. Yu, D. Cai, F. Wu, and Y. Zhuang, "Open-ended long-form video question answering via adaptive hierarchical reinforced networks," in *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI 2018, July 13-19, 2018, Stockholm, Sweden*, 2018, pp. 3683–3689.

[18] J. Gao, R. Ge, K. Chen, and R. Nevatia, "Motion-appearance co-memory networks for video question answering," in *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, 2018, pp. 6576–6585.

[19] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, 4-9 December 2017, Long Beach, CA, USA*, 2017, pp. 5998–6008.

[20] T. M. Le, V. Le, S. Venkatesh, and T. Tran, "Hierarchical conditional relation networks for video question answering," in *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, 2020, WA, USA, June 16-20, 2020*, 2020, pp. 9972–9981.

[21] D. Xu, Z. Zhao, J. Xiao, F. Wu, H. Zhang, X. He, and Y. Zhuang, "Video question answering via gradually refined attention over appearance and motion," in *Proceedings of the 2017 ACM on Multimedia Conference, MM 2017, Mountain View, CA, USA, October 23-27, 2017*, 2017, pp. 1645–1653.

[22] X. Li, L. Gao, X. Wang, W. Liu, X. Xu, H. T. Shen, and J. Song, "Learnable aggregating net with diversity learning for video question answering," in *Proceedings of the 27th ACM International Conference on Multimedia, MM 2019, Nice, France, October 21-25, 2019*, 2019, pp. 1166–1174.

[23] T. Yang, Z. Zha, H. Xie, M. Wang, and H. Zhang, "Question-aware tube-switch network for video question answering," in *Proceedings of the 27th ACM International Conference on Multimedia, MM 2019, Nice, France, October 21-25, 2019*, 2019, pp. 1184–1192.

[24] H. Song, X. Wu, B. Zhu, Y. Wu, M. Chen, and Y. Jia, "Temporal action localization in untrimmed videos using action pattern trees," *IEEE Trans. Multimedia*, vol. 21, no. 3, pp. 717–730, 2019.

[25] Z. Zhao, S. Xiao, Z. Song, C. Lu, J. Xiao, and Y. Zhuang, "Open-ended video question answering via multi-modal conditional adversarial networks," *IEEE Trans. Image Process.*, vol. 29, pp. 3859–3870, 2020.

[26] W. Jin, Z. Zhao, M. Gu, J. Yu, J. Xiao, and Y. Zhuang, "Multi-interaction network with object relation for video question answering," in *Proceedings of the 27th ACM International Conference on Multimedia, MM 2019, Nice, France, October 21-25, 2019*, 2019, pp. 1193–1201.

[27] J. Jiang, Z. Chen, H. Lin, X. Zhao, and Y. Gao, "Divide and conquer: Question-guided spatio-temporal contextual attention for video question answering," in *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, 2020, pp. 11 101–11 108.

[28] D. Huang, P. Chen, R. Zeng, Q. Du, M. Tan, and C. Gan, "Location-aware graph convolutional networks for video question answering," in *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, 2020, pp. 11 021–11 028.

[29] J. Kim, M. Ma, T. X. Pham, K. Kim, and C. D. Yoo, "Modality shifting attention network for multi-modal video question answering," in *CVPR, 2020*. IEEE, 2020, pp. 10 103–10 112.

[30] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015.

[31] Z. Yang, X. He, J. Gao, L. Deng, and A. J. Smola, "Stacked attention networks for image question answering," in *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, 2016, pp. 21–29.

[32] H. Jiang, I. Misra, M. Rohrbach, E. G. Learned-Miller, and X. Chen, "In defense of grid features for visual question answering," in *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 16-20, 2020*, 2020, pp. 10 267–10 276.

[33] T. Yu, J. Yu, Z. Yu, and D. Tao, "Compositional attention networks with two-stream fusion for video question answering," *IEEE Trans. Image Process.*, vol. 29, pp. 1204–1218, 2020.

[34] A. Santoro, D. Raposo, D. G. T. Barrett, M. Malinowski, R. Pascanu, P. W. Battaglia, and T. Lillicrap, "A simple neural network module for relational reasoning," in *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, 4-9 December 2017, Long Beach, CA, USA*, 2017, pp. 4967–4976.

[35] J. Johnson, B. Hariharan, L. van der Maaten, L. Fei-Fei, C. L. Zitnick, and R. B. Girshick, "CLEVR: A diagnostic dataset for compositional language and elementary visual reasoning," in *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, 2017, pp. 1988–1997.

[36] R. B. Palm, U. Paquet, and O. Winther, "Recurrent relational networks," in *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, 3-8 December 2018, Montréal, Canada*, 2018, pp. 3372–3382.

[37] L. Wang, Y. Xiong, Z. Wang, Y. Qiao, D. Lin, X. Tang, and L. V. Gool, "Temporal segment networks: Towards good practices for deep action recognition," in *Computer Vision - ECCV 2016 - 14th European Confer-*

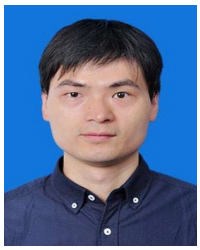
ence, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part VIII, 2016, pp. 20–36.
[38] J. Pennington, R. Socher, and C. D. Manning, "Glove: Global vectors for word representation," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL*, 2014, pp. 1532–1543.
[39] F. C. Heilbron, J. C. Niebles, and B. Ghanem, "Fast temporal activity proposals for efficient detection of human actions in untrimmed videos," in *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, 2016, pp. 1914–1923.
[40] M. Malinowski and M. Fritz, "A multi-world approach to question answering about real-world scenes based on uncertain input," in *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada*, 2014, pp. 1682–1690.
[41] Y. Xiong, L. Wang, Z. Wang, B. Zhang, H. Song, W. Li, D. Lin, Y. Qiao, L. V. Gool, and X. Tang, "CUHK & ETHZ & SIAT submission to activitynet challenge 2016," *CoRR*, vol. abs/1608.00797, 2016.
[42] Y. Yu, H. Ko, J. Choi, and G. Kim, "End-to-end concept word detection for video captioning, retrieval, and question answering," in *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, 2017, pp. 3261–3269.
[43] S. Zhang, H. Peng, J. Fu, and J. Luo, "Learning 2d temporal adjacent networks for moment localization with natural language," in *AAAI 2020*. AAAI Press, 2020, pp. 12 870–12 877.
[44] T. Lin, X. Liu, X. Li, E. Ding, and S. Wen, "BMN: boundary-matching network for temporal action proposal generation," in *ICCV, 2019*. IEEE, 2019, pp. 3888–3897.

Jipeng Zhang is currently a master student with the Center for Future Media, School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu, China. His research interests include visual question answering and semantic parsing.

Jie Shao received the B.E. degree in computer science from Southeast University, Nanjing, China, in 2004 and the Ph.D. degree in computer science from The University of Queensland, Brisbane, Australia, in 2009. He is currently a Professor with the Center for Future Media, School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu, China. He worked as a Research Fellow at the University of Melbourne from 2008 to 2011, and at National University of Singapore from 2012 to 2014. His research interests include spatial databases and multimedia information retrieval.

Rui Cao received the B.E. degree from University of Electronic Science and Technology of China, Chengdu, China in 2020. She is currently a Ph.D student in the School of Information Systems in Singapore Management University. Her research interests are visual question answering and online anti-social speech detection.

Lianli Gao received the Ph.D. degree from the University of Queensland, Australia, in 2014. She is currently a Professor with the Center for Future Media, School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu, China, teaching subjects on semantic Web and machine learning theory, etc. Her research interests include data mining, machine learning, multimedia analysis, and semantic Web.



Xing Xu received the B.E. and M.E. degrees from the Huazhong University of Science and Technology, China, in 2009 and 2012, respectively, and the Ph.D. degree from Kyushu University, Japan, in 2015. He is currently with the School of Computer Science and Engineering, University of Electronic Science and Technology of China, China. His research interests include multimedia information retrieval and pattern recognition.

Heng Tao Shen received the B.Sc. (First Class Hons.) and Ph.D. degrees in computer science from the Department of Computer Science at National University of Singapore in 2000 and 2004 respectively. He is currently a Professor, the Dean of School of Computer Science and Engineering, the Executive Dean of Artificial Intelligence Research Institute, and the Director of Center for Future Media at University of Electronic Science and Technology of China. His current research interests include multimedia search, computer vision, artificial intelligence, and big data management. He has published over 280 peer-reviewed papers, and received 7 best paper awards from international conferences, including the Best Paper Award from ACM Multimedia 2017 and Best Paper Award-Honourable Mention from ACM SIGIR 2017. He has served as General Co-chair for ACM Multimedia 2021 and Program Committee Co-Chair for ACM Multimedia 2015, and is/was an Associate Editor of ACM Transactions of Data Science, IEEE Transactions on Image Processing, IEEE Transactions on Multimedia, and IEEE Transactions on Knowledge and Data Engineering.