Open-Ended Long-Form Video Question Answering via Hierarchical Convolutional Self-Attention Networks

Zhu Zhang[1] , **Zhou Zhao**[*1] , **Zhijie Lin**[1] , **Jingkuan Song**[2] and **Xiaofei He**[3]

[1]College of Computer Science, Zhejiang University, China
[2]University of Electronic Science and Technology of China, China
[3]State Key Lab of CAD&CG, Zhejiang University, China

{zhangzhu, zhaozhou, linzhijie}@zju.edu.cn, {jingkuan.song, xiaofeihe}@gmail.com

Abstract

Open-ended video question answering aims to automatically generate the natural-language answer from referenced video contents according to the given question. Currently, most existing approaches focus on short-form video question answering with multi-modal recurrent encoder-decoder networks. Although these works have achieved promising performance, they may still be ineffectively applied to long-form video question answering due to the lack of long-range dependency modeling and the suffering from the heavy computational cost. To tackle these problems, we propose a fast Hierarchical Convolutional Self-Attention encoder-decoder network(HCSA). Concretely, we first develop a hierarchical convolutional self-attention encoder to efficiently model long-form video contents, which builds the hierarchical structure for video sequences and captures question-aware long-range dependencies from video context. We then devise a multi-scale attentive decoder to incorporate multi-layer video representations for answer generation, which avoids the information missing of the top encoder layer. The extensive experiments show the effectiveness and efficiency of our method.

1 Introduction

Open-ended video question answering is an important problem in visual information retrieval, which automatically generates the accurate answer from referenced video contents according to the given question. Most existing works employ the multi-modal recurrent encoder-decoder framework, which first encodes the multi-modal video and question contents into a joint representation, and then generates the natural-language answer [Xue *et al.*, 2017; Zhao *et al.*, 2018]. Although these approaches have achieved excellent performance in short-form video question answering, they may still be ineffectively applied to the long-form video question answering due to the lack of long-range dependency modeling and the suffering from the heavy computational cost.

---

*Zhou Zhao is the corresponding author.

Q: What does the girl do after putting the fruit into a blender?
A: She pushes the fruit down further into the blender.

Figure 1: Open-Ended Long-Form Video Question Answering.

The long-form video often contains the complex information of the targeted objects that evolve over time [Krishna *et al.*, 2017]. We illustrate a concrete example in Figure 1. The answer generation to the question "What does the girl do after putting the fruit into a blender?" requires the accurate information extraction from long-form video contents. However, the sequential processing of long-form video contents by recurrent neural networks remains the network depth-in-length, which causes the heavy computational cost and insufficiency of long-range dependency modeling. Therefore, the simple extension of multi-modal recurrent encoder-decoder networks is difficult to generate a satisfactory answer for open-ended long-form video question answering.

Recently, the convolutional sequence-to-sequence method has been applied in fast conditional text generation [Gehring *et al.*, 2017]. Based on the convolutional sequence modeling, we are able to improve the computational efficiency of long-form video encoding. However, the original convolutional sequence-to-sequence method still has some limitations. Firstly, the temporal convolution keeps the sequential length unchanged, so the multi-layer convolution operation leads to the large memory requirement, especially for long-form video sequences. Secondly, as shown in Figure 1, the crucial clues for question answering often are contained in continuous segments rather than discrete frames, but the original convolutional sequence-to-sequence strategy ignores the hierarchical structure of long-form video contents. Thirdly, this method still remains in local modeling and fails to capture long-range dependencies from video context.

We tackle these problems by a hierarchical convolution self-attention encoder, which can efficiently learn multi-layer video semantic representations. Specifically, the encoder consists of multiple convolution self-attention layers, and

each layer contains two convolution units, an attentive segmentation unit and a question-aware self-attention unit. The attentive segmentation unit splits long-form video contents into different segments and leverages question information to learn attentive segment-level representations, which builds the hierarchical structure of long-form video contents and explores multi-scale visual clues for question answering. Moreover, this method reduces the video sequential length layer by layer and is helpful to alleviate memory load. On the other hand, the question-aware self-attention unit exploits the question evidence to capture long-range dependencies from video context. Concretely, different from the conventional self-attention developed for machine translation [Vaswani *et al.*, 2017], we consider that a large portion of contents in long-form videos are irrelevant to the given question and even interfere with question answering, thus utilize the question information as guidance for filtering the unnecessary context.

After encoding long-form video contents, we consider the decoder network for answer generation. Previous works [Bahdanau *et al.*, 2014; Anderson *et al.*, 2018] often adopt the recurrent neural network with attention mechanism to generate the natural-language answer. Different from them, we develop a multi-scale attentive recurrent decoder to incorporate multi-layer semantic representations from the hierarchical video encoder according to the given question, which avoids the information missing of the top layer.

In summary, the main contributions of this paper are as follows:

- we develop a hierarchical convolutional self-attention encoder to efficiently explore the hierarchical structure of long-form video contents and capture question-aware long-range dependencies from video context.

- We devise a multi-scale attentive decoder to incorporate multi-layer video semantic representations for answer generation, which avoids information missing only from the top layer of the hierarchical encoder.

- The extensive experiments on a public long-form video question answering dataset validate the effectiveness and efficiency of our method.

2 Related Work

In this section, we briefly review some related work on visual question answering.

Given a question to an image, the image-based question answering task is to return the accurate answer for the given question [Antol *et al.*, 2015]. [Li and Jia, 2016] propose the QRU method that updates the question representation iteratively by selecting image regions. [Malinowski and Fritz, 2014] combine discrete reasoning with uncertain predictions in a bayesian framework for automatically answering questions. [Lu *et al.*, 2016] devise the co-attention mechanism for image question answering that generates spatial maps. A detailed survey of existing image question answering approaches can be found in [Wu *et al.*, 2017].

As a natural extension, the video-based question answering has been proposed as a more challenging task [Tapaswi *et al.*,

2016; Jang *et al.*, 2017]. The fill-in-the-blank methods [Maharaj *et al.*, 2017; Mazaheri *et al.*, 2016] complete the missing entries in the video description based on both visual and textual contents. [Zhu *et al.*, 2017] present the encoder-decoder approach to learn temporal structures of videos. [Gao and Han, 2017] encode the video by LSTM and then decode the answer by a language model. [Xue *et al.*, 2017] apply the sequential video attention and temporal question attention for open-ended video question answering. [Jang *et al.*, 2017] propose the dual-LSTM based approach to perform spatiotemporal reasoning from videos to answer questions. [Zeng *et al.*, 2017] extend the end-to-end memory network with additional LSTM layer for video question answering. [Zhao *et al.*, 2018] adaptively divide entire videos into several segments to model video presentation for answer generation. [Gao *et al.*, 2018] devise the dynamic memory network to learn temporal video representations for question answering. Besides single-turn video question answering, [Zhao *et al.*, 2019] propose a hierarchical attention method to solve the multi-turn video question answering task.

Although these methods have achieved promising performance, they may still suffer from the heavy computational cost and the ineffectiveness of long-range dependency modeling for long-form video contents.

3 Hierarchical Convolutional Self-Attention Encoder-Decoder Networks

3.1 The Problem

We present a video as a sequence of frames $\mathbf{v} = \{\mathbf{v}_i\}_{i=1}^{n} \in V$, where $\mathbf{v}_i$ is the feature of the i-th frame and n is the feature number. Each video is associated with a natural language question, denoted by $\mathbf{q} = \{\mathbf{q}_i\}_{i=1}^{m} \in Q$, where $\mathbf{q}_i$ is the feature of the i-th word and m is the word number. And the ground-truth answer for each question is denoted by $\mathbf{a} = \{a_i\}_{i=1}^{r} \in A$ of length r, where a_i is the i-th word token. Our goal is to generate a natural-language answer from referenced long-form video contents according to the given question. The overall framework is shown in Figure 2.

3.2 Hierarchical Convolutional Self-Attention Encoder

In this section, we introduce the hierarchical convolutional self-attention encoder that unifies the convolutional sequence modeling, attentive segmentation and question-aware self-attention mechanism into a common framework.

We first extract the visual features $\mathbf{v} = \{\mathbf{v}_i\}_{i=1}^{n}$ using a pre-trained 3D-ConvNet [Tran *et al.*, 2015] and then apply a linear projection for dimensionality reduction. Compared with RNN-based encoder, convolutional encoder lacks precise position modeling. Hence, we add position encoding [Vaswani *et al.*, 2017] into the initial video sequence to supplement the temporal information. After that, we employ a pre-trained word2vec [Mikolov *et al.*, 2013] to extract the word features $\mathbf{q} = \{\mathbf{q}_i\}_{i=1}^{m}$, and then develop a bi-directional GRU networks (BiGRU) to learn the question semantic representations. The BiGRU networks incorporate contextual in-
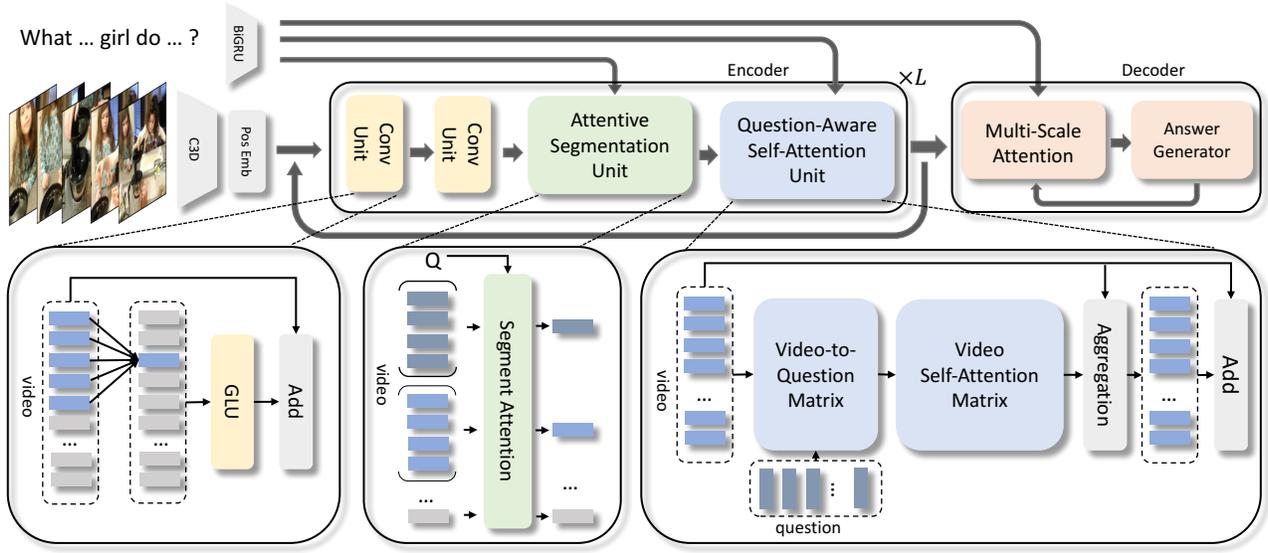
Figure 2: The Framework of Hierarchical Convolutional Self-Attention Encoder-Decoder Networks.

formation for each word, given by

$$\mathbf{h}_i^f = \mathrm{GRU}_q^f(\mathbf{q}_i, \mathbf{h}_{i-1}^f), \mathbf{h}_i^b = \mathrm{GRU}_q^b(\mathbf{q}_i, \mathbf{h}_{i+1}^b),$$

$$\mathbf{h}_i^q = [\mathbf{h}_i^f; \mathbf{h}_i^b], \mathbf{h}^Q = [\mathbf{h}_m^f; \mathbf{h}_1^b],$$

where GRU_q^f and GRU_q^b represent the forward and backward GRU, respectively. And the contextual representation $\mathbf{h}_i^q$ is the concatenation of the forward and backward hidden state at the i-th step. Thus, we get the question semantic representations $\mathbf{h}^q = (\mathbf{h}_1^q, \mathbf{h}_2^q, \dots, \mathbf{h}_m^q)$ and global question representation $\mathbf{h}^Q$.

Next, we introduce the hierarchical convolutional self-attention encoder, composed by L convolutional self-attention layers. The l-th layer takes the sequence representations $\mathbf{h}^{l-1} = (\mathbf{h}_1^{l-1}, \mathbf{h}_2^{l-1} \dots, \mathbf{h}_{n_{l-1}}^{l-1})$ of length n_{l-1} as input, and output another sequence representations $\mathbf{h}^l = (\mathbf{h}_1^l, \mathbf{h}_2^l, \dots, \mathbf{h}_{n_l}^l)$ of length n_l, where $n_{l-1} = H \times n_l$ and the sequence is reduced by a factor of $1/H$ per layer. The input of the first layer is the initial video sequence $\mathbf{v} = \{\mathbf{v}_i\}_{i=1}^n$ with n elements. Concretely, each layer consists of two convolution units, an attentive segmentation unit and a question-aware self-attention unit.

Convolution Unit

The convolution unit can efficiently model the local context of long-form video contents. We first conduct a one-dimensional convolution for the input sequence. Specifically, we parameterize the convolution kernel by $\mathbf{W}^l \in \mathbb{R}^{2d \times kd}$ and $\mathbf{b}^l \in \mathbb{R}^{2d}$, where k is the kernel width and d represents the dimensionality of sequence elements. We then concatenate successive k elements and map them into a single output $\mathbf{Y} \in \mathbb{R}^{2d}$ with the convolutional kernel, given by $\mathbf{Y} = \mathbf{W}^l[\mathbf{h}_{i-k/2}^{l-1}; \dots; \mathbf{h}_{i+k/2}^{l-1}] + \mathbf{b}^l$. We pad the input sequences by $k/2$ zero vectors on the both sides to keep sequence length invariable.

We then devise a special non-linearity operation by the gated linear unit, called as GLU [Gehring *et al.*, 2017], on the output $\mathbf{Y} = [\mathbf{A}; \mathbf{B}] \in \mathbb{R}^{2d}$:

$$\mathrm{GLU}([\mathbf{A}, \mathbf{B}]) = \mathbf{A} \otimes \delta(\mathbf{B}),$$

where $\mathbf{A}, \mathbf{B} \in \mathbb{R}^d$ are the inputs to the GLU, δ is the sigmoid function, $\otimes$ represents the point-wise multiplication and the output $\mathrm{GLU}([\mathbf{A}, \mathbf{B}]) \in \mathbb{R}^d$ has the same dimensionality as the input element. Furthermore, to build deeper convolutional encoder, we add residual connection for each convolutional unit, given by

$$\mathbf{o}_i^{l1} = \mathrm{GLU}(\mathbf{W}^l[\mathbf{h}_{i-k/2}^{l-1}; \dots; \mathbf{h}_{i+k/2}^{l-1}] + \mathbf{b}^l) + \mathbf{h}_i^{l-1}.$$

We note that $\mathbf{o}_i^{l1}$ contains information over k input elements and the receptive field of final output can be expanded rapidly by stacking several units. And the output of the two convolutional units of the l-th layer is $(\mathbf{o}_1^{l2}, \mathbf{o}_2^{l2}, \cdots; \mathbf{o}_{n_{l-1}}^{l2})$.

Attentive Segmentation Unit

The attentive segmentation unit splits long-form video contents into different segments and leverages question information to learn attentive segment-level representations. Thus, this method can reduces the video sequential length layer by layer to build the hierarchical video structure and also alleviate memory load.

Given the convolution output $(\mathbf{o}_1^{l2}, \mathbf{o}_2^{l2}, \cdots; \mathbf{o}_{n_{l-1}}^{l2})$, we first devide sequence elements into n_l segments, each containing H elements (i.e. $n_{l-1} = H \times n_l$). We then devise an attention mechanism to aggregate element representations for each segment, where we utilize the question representation as guidance to highlight critical elements. Different from mean-pooling aggregation, this method can effectively filter the irrelevant video contents for subsequent modeling.

Concretely, for the i-th segment $(\mathbf{o}_{(i-1)H+1}^{l2}, \dots, \mathbf{o}_{iH}^{l2})$ with H elements, we compute the attention weight of j-th

element $\mathbf{o}^{l2}_{(i-1)H+j}$ according to the global question representation $\mathbf{h}^Q$, given by

$$\alpha_{ij} = \mathbf{w}_s^\top \tanh(\mathbf{W}_s^1 \mathbf{o}^{l2}_{(i-1)H+j} + \mathbf{W}_s^2 \mathbf{h}^Q + \mathbf{b}_s),$$

where $\mathbf{W}_s^1$, $\mathbf{W}_s^2$ are parameter matrices, $\mathbf{b}_s$ is the bias vector and the $\mathbf{w}_s^\top$ is the row vector for computing the attention weight. We then conduct the softmax operation on attention weights and obtain the segment representation $\mathbf{s}_i^l$ by $\mathbf{s}_i^l = \sum_{j=1}^{H} \text{softmax}(\alpha_{ij}) \mathbf{o}^{l2}_{(i-1)H+j}$.

Hence, by the attentive segmentation unit, we learn the segment-level video representations $\mathbf{s}^l = (\mathbf{s}_1^l, \mathbf{s}_2^l \ldots, \mathbf{s}_{n_l}^l)$. This segmentation strategy builds the hierarchical structure of long-form video contents, which is helpful for exploring multi-scale viusal clues for question answering. Moreover, this method reduces the video sequential length layer by layer to alleviate the memory load.

Question-Aware Self-Attention Unit

The question-aware self-attention unit captures the long-range dependencies from long-form video context with question information as guidance, which filters the unnecessary context.

Given the ouput of the attentive segmentation unit $\mathbf{s}^l = (\mathbf{s}_1^l, \mathbf{s}_2^l, \ldots, \mathbf{s}_{n_l}^l)$ and question semantic representations $\mathbf{h}^q = (\mathbf{h}_1^q, \mathbf{h}_2^q, \ldots, \mathbf{h}_m^q)$, we can compute the video-to-question attention weights between each pair of sequence element and word, and obtain a video-to-question attention matrix $\mathbf{M} \in \mathbb{R}^{n_l \times m}$. Specifically, the attention weight of the i-th element and j-th word is calculated by

$$\mathbf{M}_{ij} = \mathbf{w}_m^\top \tanh(\mathbf{W}_m^1 \mathbf{s}_i^l + \mathbf{W}_m^2 \mathbf{h}_j^q + \mathbf{b}_m).$$

We then calculate the self-attention matrix $\mathbf{D}$ for video contents based on $\mathbf{M}$ as below:

$$\mathbf{D} = \mathbf{M} \cdot \mathbf{M}^\top, \ \mathbf{D} \in \mathbb{R}^{n_l \times n_l}$$

where each value in $\mathbf{D}$ represents the correlation of two video elements. Specifically, each value in $\mathbf{D}$ is calculated by $\mathbf{D}_{ij} = \sum_{k=1}^{m} \mathbf{S}_{ik} \mathbf{S}_{kj}^\top$, where k represents the index of the k-th word in the question. That is, we regard question semantic representations as the middle layer while establishing element-to-element correlation of the video sequence. Compared with the conventional self-attention method, we filter the ineffective video context with question contents as guidance while capturing long-range dependencies.

We then conduct the softmax operation for each row in the self-attention matrix $\mathbf{D}$ and compute self-attention representations for each element, followed by an additive residual connection:

$$\mathbf{h}_i^l = \mathbf{s}_i^l + \sum_{j=1}^{n} \text{softmax}(\mathbf{D}_{ij}) \mathbf{s}_j^l.$$

Therefore, we obtain the final output of the l-th convolutional self-attention layer $\mathbf{h}^l = (\mathbf{h}_1^l, \mathbf{h}_2^l \ldots, \mathbf{h}_{n_l}^l)$.

By stacking L convolutional self-attention layers, we learn multi-layer video semantic representations $(\mathbf{h}^1, \mathbf{h}^2, \ldots, \mathbf{h}^L)$.

3.3 Multi-Scale Attentive Decoder Network

In this section, we introduce the multi-scale attentive decoder to consider the multi-scale visual clues from the hierarchical encoder for natural-language answer generation.

Our multi-scale attentive decoder is based on a GRU answer generator. At each time step, we conduct a recurrent operation as follows:

$$\mathbf{h}_t^a = \text{GRU}(\mathbf{x}_t, \mathbf{h}_{t-1}^a),$$

where $\mathbf{x}_t$ is the input vector and $\mathbf{h}_t^a$ is the output vector of the t-th step. The $\mathbf{x}_t$ consists of the embedding $\mathbf{w}_t$ of the input word at the t-th step, global question representation $\mathbf{h}^Q$ and multi-scale video representation $\mathbf{h}_t^v$], given by $\mathbf{x}_t = [\mathbf{w}_t; \mathbf{h}^Q; \mathbf{h}_t^v]$. We then introduce how to develop the multi-scale video representations $\mathbf{h}_t^v$. By hierarchical encoder, the long-form video sequences are exponentially shorten layer by layer, and the output from the top layer of the hierarchical encoder aggregates informative segment-level representations for question answering. However, only considering the top-layer output may ignore fine-grained features in preceding layers and lead to information missing. Thus, after trade-off consideration, we devise a multi-scale attention on encoder outputs of top-K layers. Specifically, for the l-layer video representations $\mathbf{h}^l = (\mathbf{h}_1^l, \mathbf{h}_2^l, \ldots, \mathbf{h}_{n_l}^l)$, we compute the attention weight of i-th element with the hidden state $\mathbf{h}_{t-1}^a$ and glabal question representation $\mathbf{h}^Q$, given by

$$\beta_{li} = \mathbf{w}_g^\top \tanh(\mathbf{W}_g^1 \mathbf{h}_i^l + \mathbf{W}_g^2 \mathbf{h}_{t-1}^a + \mathbf{W}_g^3 \mathbf{h}^Q + \mathbf{b}_g).$$

We then conduct the softmax operation on attention weights and obtain the l-layer attentive representation $\mathbf{v}^l$ by

$$\mathbf{v}^l = \sum_{i=1}^{n_l} \text{softmax}(\beta_{li}) \mathbf{h}_i^l.$$

Based on top-K attentive representations, we finally get the multi-scale video representation $\mathbf{h}_t^v$ by a mean pooling:

$$\mathbf{h}_t^v = \frac{1}{K} \sum_{l=L-K+1}^{L} \mathbf{v}^l.$$

With $\mathbf{x_t}$ as input, we get the output vector $\mathbf{h}_t^a$ of GRU networks at the t-th time step, and further generate the conditional distribution over possible words, given by

$$p_\theta(\hat{a}_t | \hat{a}_{1:t-1}) = \text{softmax}(\mathbf{W}_a \mathbf{h}_t^a + \mathbf{b}_a) \in \mathbb{R}^T,$$

where $\hat{a}_t$ is the t-th generated word and T is the size of the vocabulary. Finally, we apply the maximum likelihood estimation to train our encoder-decoder networks in an end-to-end manner, given by

$$\mathcal{L}_{ML} = - \sum_{t=1}^{r} \log p_\theta(\hat{a}_t | \hat{a}_{1:t-1}).$$

4 Experiments

4.1 Dataset

We conduct experiments on an open-ended long-form video question answering dataset [Zhao *et al.*, 2018], which is constructed from the ActivityCaption dataset [Krishna *et al.*, 2017] with natural-language descriptions. The average video time in the dataset is approximately 180 seconds and the longest video even lasts over 10 minutes. The question-answer pairs contain five types corresponding to the object, number, color, location and action for the video contents. The first four types of questions mainly focus on the appearance visual features in videos but motion features are necessary to the action related question answering. The details of this dataset are summarized in Table 1.

Data	Question Types				
	Object	Number	Color	Location	Action
Train	10,338	2,605	3,680	7,438	28,543
Valid	1,327	289	447	1,082	3,692
Test	1,296	355	501	971	3,826
All	12,961	3,249	4,628	9,491	36,061

Table 1: Summaries of the Dataset

4.2 Implementation Details

In this section, we introduce data preprocessing and model settings.

We first resize each frame to 112×112 and then employ the pre-trained 3D-ConvNet [Tran *et al.*, 2015] to extract the 4,096-d feature for each unit, which contains 16 frames and overlaps 8 frames with adjacent units. We then reduce the dimensionality of the feature from 4,096 to 500 using PCA. Next, we set the maximum sequence length to 512 and down-sample overlong sequences to this length. As for the question, we employ the pre-trained word2vec model [Mikolov *et al.*, 2013] to extract the semantic features word by word. The dimension of word features is 300 and the vocabulary size T is 10000.

In our HCSA, we set the layer number L of the hierarchical convolutional self-attention encoder to 3. And the segmentation factor H in the attentive segmentation unit is set to 4. To avoid heavy computational cost, we only consider top-2 layers ($K = 2$) of the hierarchical encoder for the multi-scale attentive decoder. Moreover, we set convolution kernel width k to 5, convolution dimension to 256 and the dimension of the hidden state of GRU networks to 256 (512 for BiGRU while question encoding). And the dimensions of the linear matrice in all kinds of attention are set to 256. During training, we adopt an adam optimizer to minimize the loss and the learning rate is set to 0.001.

4.3 Performance Criteria

We evaluate the performance of open-ended video question answering based on evaluation criteria BLEU-1, WUPS@0.0 and WUPS@0.9. Since the length of answers is relatively short, we mainly focus on word-level evaluation. The BLEU-1 is for accurate word matching and WUPS [Malinowski and Fritz, 2014] accounts for word-level ambiguities. Given the ground-truth answer $\mathbf{a} = \{a_1, a_2, \ldots, a_r\}$ and the generated answer $\hat{\mathbf{a}} = \{\hat{a}_1, \hat{a}_2, \ldots, \hat{a}_r\}$, the WUPS is given by

$$\text{WUPS} = \frac{1}{|Q|} \sum_{\mathbf{q} \in Q} \min\{\frac{1}{r} \sum_{a_i \in \mathbf{a}} \max_{\hat{a}_j \in \hat{\mathbf{a}}} WUP_\gamma(a_i, \hat{a}_j), \\ \frac{1}{r} \sum_{\hat{a}_i \in \hat{\mathbf{a}}} \max_{a_j \in \mathbf{a}} WUP_\gamma(\hat{a}_i, a_j)\},$$

where the $\text{WUP}_\gamma(\cdot)$ [Wu and Palmer, 1994] calculates word similarity by the WordNet [Fellbaum, 1998], given by

$$\text{WUP}_\gamma(a_i, \hat{a}_j) = \begin{cases} \text{WUP}(a_i, \hat{a}_j), & \text{WUP}(a_i, \hat{a}_j) \geq \gamma \\ 0.1 \cdot \text{WUP}(a_i, \hat{a}_j). & \text{WUP}(a_i, \hat{a}_j) < \gamma \end{cases}$$

We set the threshold γ to 0 and 0.9 and denote the two criteria by WUPS@0.0 and WUPS@0.9, respectively.

Method	BLEU-1	WUPS@0.9	WUPS@0.0
MN+	19.86	28.37	56.87
UNIFY	24.13	29.85	58.56
STVQA+	24.64	33.37	58.97
CDMN+	25.38	34.53	59.20
AHN	25.81	34.14	59.66
HCSA	**28.83**	**36.90**	**61.74**

Table 2: Experimental results on BLEU-1, WUPS@0.9 and WUPS@0.0 with all types of visual questions.

Method	train(s)/epo	infer(s)/epo	epoch	params
MN+	372.12	83.54	17	6.27M
UNIFY	253.71	61.29	14	7.17M
STVQA+	207.43	38.37	11	9.50M
CDMN+	276.53	66.45	10	8.97M
AHN	231.94	52.76	12	8.24M
HCSA	**45.13**	**12.82**	**8**	6.37M

Table 3: Experimental results of training time, inference time, training epoch and model parameters.

4.4 Performance Comparisons

The open-ended video question answering is an emerging task, thus we compare our HCSA method with existing open-ended methods and meanwhile extend some conventional multi-choice methods for performance evaluation. Specifically, we add a GRU recurrent answer generator with attention mechanism to the end of those non-open-ended models.

- **MN+** method [Zeng *et al.*, 2017] is the extension of end-to-end memory network algorithm, where we add a bi-LSTM network to encode the sequence of video frames.

- **STVQA+** method [Jang *et al.*, 2017] utilizes the spatial and temporal attention strategies on videos to answer related questions.

- **CDMN+** method [Gao *et al.*, 2018] proposes a motion-appearance co-memory network to simultaneously learn the motion and appearance features.

- **UNIFY** method [Xue *et al.*, 2017] applies the sequential video attention and temporal question attention for open-ended video question answering.

- **AHN** method [Zhao *et al.*, 2018] divides entire videos into several segments and adopts a hierarchical attention to model video presentations for answer generation.

The former three approaches are originally developed for multi-choice video question answering and we extend them into the open-ended form. Table 2 shows the overall experimental results of all methods on three criteria BLEU-1, WUPS@0.9 and WUPS@0.0. Table 4 and 5 demonstrate the evaluation results of different question types on BLEU-1 and WUPS@0.9, respectively. Moreover, we adjust the parameter number in different methods at the same magnitude for fairly evaluating the time consumption, shown in Table 3. The training and inference time for each epoch only contain the network execution time. The experiment results reveal some interesting points:

- The methods based on attention mechanism, UNIFY, STVQA+, CDMN+, AHN, and HCSA achieve better

Method	BLEU-1				
	Object	Number	Color	Location	Action
MN+	25.02	59.58	22.53	31.09	11.21
UNIFY	31.01	74.98	26.21	33.26	14.52
STVQA+	30.53	75.56	26.62	33.77	15.35
CDMN+	31.06	76.43	24.20	35.92	15.57
AHN	31.48	78.72	24.63	36.35	16.42
HCSA	**34.48**	**79.49**	**26.91**	**37.23**	**20.33**

Table 4: Results on BLEU-1 with different question types.

Method	WUPS@0.9				
	Object	Number	Color	Location	Action
MN+	29.22	70.43	46.69	31.73	20.93
UNIFY	38.63	85.04	46.81	36.12	23.54
STVQA+	38.75	87.70	46.86	36.73	23.89
CDMN+	38.79	87.13	46.40	37.72	24.11
AHN	39.43	87.85	47.12	38.21	24.70
HCSA	**41.17**	**88.06**	**47.52**	**40.12**	**28.49**

Table 5: Results on WUPS@0.9 with different question types.

performance than MN+, which demonstrates that the joint representation learning of video and question is critical for high-quality answer generation.

- Our HCSA method achieves the best performance on three evaluation criteria and all question types, especially for action type. It suggests that our hierarchical convolutional self-attention encoder and the multi-scale attentive decoder are effective for the problem.

- From the perspective of time consumption, our HCSA method reduces plenty of training and testing time per epoch compared with all baselines, and can converge more quickly using quite a few epochs. This suggests that the convolution-based sequence modeling is more efficient than RNN-based sequence modeling.

4.5 Ablation Study and Visualization

To prove the contribution of each component of our HCSA method, we next conduct some ablation studies. Concretely, we discard or change one component at a time to generate an ablation model. We first replace the segment attention in the attentive segmentation unit with a conventional mean-pooling, denoted by **ASU(MP)**, and further remove attentive segmentation unit as **w/o. ASU**. We then replace the question-aware self-attention with a typical self-attention, denoted by **QSU(SA)**, and thoroughly discard the question-aware self-attention unit as **w/o. QSU**. We finally denote the model that only considers the top-layer video representations in the decoder by **w/o. MA**. The ablation results are shown in Table 6.

By analyzing the ablation results, we find the full method outperforms **w/o. ASU**, **w/o. QSU** and **w/o. MA** models, indicating the attentive segmentation, question-aware self-attention and multi-scale attention are all helpful for high-quality answer generation. Additionally, the **ASU(MP)** and **QSU(SA)** still achieve worse performance than the full model, which demonstrates our proposed segment attention and question-aware self-attention is more effective than conventional mean-pooling and self-attention. Moreover, the

Configuration	BLEU-1	WUPS@0.9	WUPS@0.0
ASU(MP)	26.73	34.84	59.71
w/o. ASU	25.64	33.79	58.98
QSU(SA)	28.06	36.11	61.12
w/o. QSU	27.32	35.40	60.44
w/o. MA	27.65	35.61	60.70
HCSA	**28.83**	**36.90**	**61.74**

Table 6: Experimental Results of Ablation Study

Question: What does the boy do after bringing in water?
Ground Truth: Pours water into the sand.   Generated Answer: Pour water.
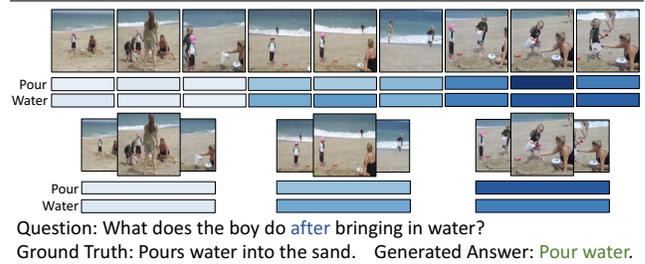
Figure 3: The Multi-Scale Attention Results in the Decoder.

worst performance of **w/o. ASU** suggests the hierarchical structure is crucial for long-form video modeling.

Furthermore, we demonstrate how the multi-scale attentive decoder works by a visualization example. As shown in Figure 3, we display the multi-scale attention results for video representations using a thermodynamic diagram, where the darker color represents the higher correlation. For each step of answer generation, the multi-scale attention produces a weight distribution over the video semantic representations of top-2 layers, We note that our proposed method can attend the semantically related visual contents and ignore these irrelevant features at each step. It suggests our proposed decoder effectively incorporates the multi-scale visual clues for high-quality open-ended long-form video question answering.

5 Conclusion

In this paper, we propose a fast hierarchical convolutional self-attention encoder-decoder network for open-ended long-form video question answering. We first propose a hierarchical convolutional self-attention encoder to efficiently model long-form video contents, which builds the hierarchical structure for video sequences and captures question-aware long-range dependencies from video context. We then devise a multi-scale attentive decoder to incorporate multi-layer video representations for natural-language answer generation. The extensive experiments show the effectiveness and efficiency of our method.

Acknowledgments

References

[Anderson *et al.*, 2018] Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang. Bottom-up and top-down attention for image captioning and visual question answering. In *CVPR*, pages 6077–6086, 2018.

[Antol *et al.*, 2015] Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C Lawrence Zitnick, and Devi Parikh. Vqa: Visual question answering. In *ICCV*, pages 2425–2433, 2015.

[Bahdanau *et al.*, 2014] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*, 2014.

[Fellbaum, 1998] Christiane Fellbaum. *WordNet*. Wiley Online Library, 1998.

[Gao and Han, 2017] Kun Gao and Yahong Han. Spatiotemporal context networks for video question answering. In *Pacific Rim Conference on Multimedia*, pages 108–118. Springer, 2017.

[Gao *et al.*, 2018] Jiyang Gao, Runzhou Ge, Kan Chen, and Ram Nevatia. Motion-appearance co-memory networks for video question answering. *CVPR*, 2018.

[Gehring *et al.*, 2017] Jonas Gehring, Michael Auli, David Grangier, Denis Yarats, and Yann N Dauphin. Convolutional sequence to sequence learning. *arXiv preprint arXiv:1705.03122*, 2017.

[Jang *et al.*, 2017] Yunseok Jang, Yale Song, Youngjae Yu, Youngjin Kim, and Gunhee Kim. Tgif-qa: Toward spatiotemporal reasoning in visual question answering. In *CVPR*, pages 2680–8, 2017.

[Krishna *et al.*, 2017] Ranjay Krishna, Kenji Hata, Frederic Ren, Li Fei-Fei, and Juan Carlos Niebles. Densecaptioning events in videos. In *ICCV*, pages 706–715, 2017.

[Li and Jia, 2016] Ruiyu Li and Jiaya Jia. Visual question answering with question representation update (qru). In *NIPS*, pages 4655–4663, 2016.

[Lu *et al.*, 2016] Jiasen Lu, Jianwei Yang, Dhruv Batra, and Devi Parikh. Hierarchical question-image co-attention for visual question answering. In *NIPS*, pages 289–297, 2016.

[Maharaj *et al.*, 2017] Tegan Maharaj, Nicolas Ballas, Anna Rohrbach, Aaron C Courville, and Christopher Joseph Pal. A dataset and exploration of models for understanding video data through fill-in-the-blank question-answering. In *CVPR*, pages 7359–7368, 2017.

[Malinowski and Fritz, 2014] Mateusz Malinowski and Mario Fritz. A multi-world approach to question answering about real-world scenes based on uncertain input. In *NIPS*, pages 1682–1690, 2014.

[Mazaheri *et al.*, 2016] Amir Mazaheri, Dong Zhang, and Mubarak Shah. Video fill in the blank with merging lstms. *arXiv preprint arXiv:1610.04062*, 2016.

[Mikolov *et al.*, 2013] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.

[Tapaswi *et al.*, 2016] Makarand Tapaswi, Yukun Zhu, Rainer Stiefelhagen, Antonio Torralba, Raquel Urtasun, and Sanja Fidler. Movieqa: Understanding stories in movies through question-answering. In *CVPR*, pages 4631–4640, 2016.

[Tran *et al.*, 2015] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. Learning spatiotemporal features with 3d convolutional networks. In *ICCV*, pages 4489–4497, 2015.

[Vaswani *et al.*, 2017] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NIPS*, pages 5998–6008, 2017.

[Wu and Palmer, 1994] Zhibiao Wu and Martha Palmer. Verbs semantics and lexical selection. In *Proceedings of the 32nd annual meeting on Association for Computational Linguistics*, pages 133–138. ACL, 1994.

[Wu *et al.*, 2017] Qi Wu, Damien Teney, Peng Wang, Chunhua Shen, Anthony Dick, and Anton van den Hengel. Visual question answering: A survey of methods and datasets. *Computer Vision and Image Understanding*, 163:21–40, 2017.

[Xue *et al.*, 2017] Hongyang Xue, Zhou Zhao, and Deng Cai. Unifying the video and question attentions for open-ended video question answering. *IEEE Transactions on Image Processing*, 26(12):5656–5666, 2017.

[Zeng *et al.*, 2017] Kuo-Hao Zeng, Tseng-Hung Chen, Ching-Yao Chuang, Yuan-Hong Liao, Juan Carlos Niebles, and Min Sun. Leveraging video descriptions to learn video question answering. In *AAAI*, pages 4334–4340, 2017.

[Zhao *et al.*, 2018] Zhou Zhao, Zhu Zhang, Shuwen Xiao, Zhou Yu, Jun Yu, Deng Cai, Fei Wu, and Yueting Zhuang. Open-ended long-form video question answering via adaptive hierarchical reinforced networks. In *IJCAI*, pages 3683–3689, 2018.

[Zhao *et al.*, 2019] Zhou Zhao, Zhu Zhang, Xinghua Jiang, and Deng Cai. Multi-turn video question answering via hierarchical attention context reinforced networks. *IEEE Transactions on Image Processing*, 2019.

[Zhu *et al.*, 2017] Linchao Zhu, Zhongwen Xu, Yi Yang, and Alexander G Hauptmann. Uncovering the temporal context for video question answering. *IJCV*, 124(3):409–421, 2017.