Open-Ended Long-form Video Question Answering via Adaptive Hierarchical Reinforced Networks

Zhou Zhao¹, Zhu Zhang¹, Shuwen Xiao¹, Zhou Yu², Jun Yu², Deng Cai³, Fei Wu¹, Yueting Zhuang¹

¹ College of Computer Science, Zhejiang University
² College of Computer Science, Hangzhou Dianzi University
³ State Key Lab of CAD&CG, Zhejiang University

{csezhaozhou,dengcai}@gmail.com, {zz,xsw,wufei,yzhuang}@zju.edu.cn, {yuz,yujun}@hdu.edu.cn

Abstract

Open-ended long-form video question answering is a challenging problem in visual information retrieval, which automatically generates the natural language answer from the referenced long-form video content according to the question. However, the existing video question answering works mainly focus on the short-form video question answering, due to the lack of modeling the semantic representation of long-form video contents. In this paper, we consider the problem of long-form video question answering from the viewpoint of adaptive hierarchical reinforced encoder-decoder network learning. We propose the adaptive hierarchical encoder network to learn the joint representation of the long-form video contents according to the question with adaptive video segmentation. We then develop the reinforced decoder network to generate the natural language answer for open-ended video question answering. We construct a large-scale long-form video question answering dataset. The extensive experiments show the effectiveness of our method.

1 Introduction

Video question answering is the visual information delivery mechanism that enables users to issue their queries and then collect the answers from the referenced visual contents. Open-ended video question answering is the essential problem of visual question answering, which automatically generates the natural language answer from the referenced video contents according to the given question. Currently, most of the video question answering approaches mainly focus on the problem of short-form video question answering [Zeng et al., 2017; Zhao et al., 2017; Jung et al., 2017], which learn the semantic video representation from LSTM network layer, and then generate the answer to the given question. Although the existing works have achieved promising performance in short-form video question answering, they may still be inefficiently applied to the long-form video question answering due to the lack of modeling the semantic representation of long-form video contents.

The long-form video contents often contain the evolving complex object interactions through frames, which have long-term semantic dependencies [Lezama et al., 2011]. We illustrate a simple example in Figure 1. We show that the answer generation to question “what is the woman holding?” requires the semantic localization of the targeted segment from the long-form video contents. Thus, the simple extension of the existing video question answering works based on frame-level LSTM networks is difficult for modeling the semantic representation of long-form video contents according to the given question [Krishnan and Sitaraman, 2013]. Recently, hierarchical neural encoder [Pan et al., 2016] has been proposed to learn the segment-level video semantic representation with fixed segment length. We then employ the hierarchical neural encoder with attention mechanisms to learn the joint representation of video contents according to the given question. On the other hand, although the video frames are topically consistent, they have different semantic contents [Lezama et al., 2011]. Inspired by binary neurons [Bengio et al., 2013], we then develop the hierarchical neural encoder that unifies the adaptive video segmentation and joint representation modeling of video content according to the given question into a joint learning framework. Thus, leverage the hierarchical neural encoder with adaptive video segmentation is critical for modeling the semantic representation of long-form video contents for video question answering.

In this paper, we study the problem of open-ended long-form video question answering from the viewpoint of adaptive hierarchical reinforced encoder-decoder network learning. We propose the hierarchical neural encoder with adaptive recurrent network to learn segment-level question-aware
video representation with adaptive video segmentation. We devise the reinforced decoder network to generate the natural language answer for open-ended video question answering. We then develop the adaptive hierarchical reinforced network learning framework, named as AHN. When a certain question is issued, AHN can generate natural language answer for it based on the referenced video contents. The main contribution of this paper are as follows:

- Unlike the previous studies, we study the problem of open-ended long-form video question answering from the viewpoint of adaptive hierarchical reinforced encoder-decoder network learning.
- We develop the adaptive hierarchical encoder to learn the segment-level question-aware video representation with adaptive video segmentation. We then devise the reinforced decoder to generate the answer for open-ended video question answering.
- We construct a large-scale dataset for open-ended long-form video question answering and validate the effectiveness of our propose method through extensive experiments.

2 Video Question Answering via Adaptive Hierarchical Reinforced Networks

2.1 The Problem

Before presenting the learning framework, we first introduce some basic notions and terminologies. We denote the question by \( q \in Q \), the video \( v \in V \) and the answer by \( a \in A \), where \( Q, V \) and \( A \) are the sets of question, videos and answers, respectively. Since the video is composed of sequential frames, the frame-level representation of video \( v \) is given by \( v = (v_1, v_2, \ldots, v_N) \) of length \( N \), where \( v_2 \) is the second frame. We then encode the word-level representation of natural language answer by \( a = (a_1, a_2, \ldots, a_M) \) of length \( M \), where \( a_M \) is the \( M \)-th word token. We then denote the collection of video segments by \( \{S_1, S_2, \ldots, S_K\} \) of size \( K \), where \( S_2 \) is the set of segmented frames and \( v_s \in S_2 \) means that the \( i \)-th frame belongs to segment \( S_2 \). The semantic representation of segment \( S_k \) is denoted by \( s_k \), and the sequential representation of video segments is given by \( (s_1, s_2, \ldots, s_K) \). Since both the video and answer are sequential data with variant length, it is natural to choose the variant recurrent neural network called long-short term memory network (LSTM) [Hochreiter and Schmidhuber, 1997] to learn their feature representations by

\[
\begin{align*}
i_t &= \delta(W_{ix}x_t + G_i h_{t-1} + b_i), \\
c_t &= \tanh(W_{xc}x_t + G_t h_{t-1} + b_f), \\
f_t &= \delta(W_f x_t + G_f h_{t-1} + b_f), \\
c_t &= i_t \cdot c_t + f_t \cdot c_t, \\
o_t &= \delta(W_o x_t + G_o h_{t-1} + V_o c_t + b_o), \\
h_t &= o_t \cdot \tanh(c_t),
\end{align*}
\]

where \( \delta \) represents the sigmoid activation function; \( W_{ix}, G_i \) and \( V_o \) are the weight matrices, and \( b_i, b_o, b_f, b_o \) are the bias vectors. The memory cell \( c_t \) maintains the history of the inputs observed up to the timestep. Update operations on the memory cell are modulated by three gates \( i_t, f_t, o_t \), which are all computed as a combination of the current input \( x_t \), and of the previous hidden state \( h_{t-1} \), followed by a sigmoid activation. Specifically, we denote the semantic representation of video \( v \) by \( h^{(v)} = (h^{(v)}_1, h^{(v)}_2, \ldots, h^{(v)}_N) \) and that of answer \( a \) by \( h^{(a)} = (h^{(a)}_1, h^{(a)}_2, \ldots, h^{(a)}_M) \) using LSTM networks.

Using the notations above, the problem of open-ended video question answering is formulated as follows. Given the set of videos \( V \), questions \( Q \) and answers \( A \), our goal is to learn the encoder-decoder network model \( g(f(v, q)) \) where the encoder network \( f(v, q) \) that learns the joint representation of the video and question, and the decoder network \( \hat{a} = g(f(v, q)) \) generates the answer \( \hat{a} \) for open-ended video question answering.

2.2 Adaptive Encoder Network Learning

In this section, we propose the adaptive encoder network \( f(v, q) \) that unifies the hierarchical video segmentation and semantic representation learning into a common framework.

The video contents often contain a number of frames with the targeted objects to the given question that evolves over time [Yao et al., 2015]. Inspired by binary neuron [Bengio et al., 2013], we propose the adaptive attentional recurrent neural networks that segment the complex events in video to capture their long-term semantic dependencies, and learn the joint representation of relevant frames and segments according to the question. We first extract the frame-level video feature using ConvNet [Simonyan and Zisserman, 2014] by \( v^{(j)}(j) = (v^{(j)}_1, v^{(j)}_2, \ldots, v^{(j)}_N) \), and then learn their semantic representation using LSTM networks. When the end frame of a video event is estimated, we reset the LSTM parameters of the next frame in order to segment the video. To enable the video segmentation, we define an adaptive recurrent neural networks with binary gate function, which decides whether to transfer the LSTM parameters (i.e., hidden state \( h^{(v)}_t \) and memory cell \( c^{(v)}_t \) of the current frame to update the LSTM parameters of the next frame (i.e., hidden state \( h^{(v)}_{t+1} \) and memory cell \( c^{(v)}_{t+1} \) by Equations (1), (2), (3), (4) and (5) or reinitialize them. Formally, the \( t \)-th binary gate is defined as a step function, which is computed as a non-linear combination of the feature of the \( t \) and \( t+1 \) frame from ConvNet (i.e., \( v^{(j)}_{t+1} \) and the semantic representation of the \( t \) frame from LSTM networks (i.e., \( h^{(v)}_t \), given by

\[
\gamma_t(v^{(j)}_{t+1}, h^{(v)}_t) = 1[\delta(w_{\gamma}v^{(j)}_{t+1} + w_{\gamma}h^{(v)}_t + b_{\gamma}) > \tau].
\]

The \( 1[\cdot] \) is a step function and \( \delta(\cdot) \) is a sigmoid function. The \( w_{\gamma} \) is a learnable row vector, \( W_{\gamma} \), \( W_{\gamma}h \) and \( b_{\gamma} \) are the learnable weights and bias. The \( \tau \) is the threshold parameter of the step function. For example, the inputs of the 3-th binary gate \( \gamma_3 \) is the \( h^{(v)}_3 \) (i.e., the hidden state of the 3-th frame from LSTM networks) and \( v^{(j)}_3 \) (i.e., the video feature of the 4-th frame) in Figure 2. That is, the binary gate function \( \gamma_t \) decides whether to transfer the parameters of the \( t \) frame from LSTM networks by \( h^{(v)}_t \leftarrow h^{(v)}_t \cdot (1 - \gamma_t) \) and \( c^{(v)}_t \leftarrow c^{(v)}_t \cdot (1 - \gamma_t) \), respectively.
Given the semantic representation of frames \((h_1^{(s)}, h_2^{(s)}, \ldots, h_N^{(s)})\) with binary gate values \((\gamma_1, \gamma_2, \ldots, \gamma_{N-1})\), we then learn the joint question-aware video segment representation. If the value of binary gate \(\gamma_t = 1\), the question-aware representation of current segment \(S_k\) is computed and then passed to the segment-level LSTM networks. Given the question representation \(h^{(q)}\), the frame-level attention for the \(t\)-th frame \(v_t \in S_k\) is given by

\[
a_t^{(v)} = P(f)^{\text{tanh}}(W_h^{(f)} h_t^{(s)} + W_q^{(f)} h^{(q)} + b^{(v)}),
\]

where \(W_h^{(f)}, W_q^{(f)}\) are parameter matrices and \(b^{(v)}\) is the bias vector. The \(P(f)\) is the parameter vector for computing the frame-level attention score. For each frame \(v_t \in S_k\), its activation by the softmax function is given by \(\beta_t^{(v)} = \frac{\exp(a_t^{(v)})}{\sum_{v_t \in S_k} \exp(a_t^{(v)})}\). Thus, the attentional semantic representation for segment \(S_k\) is then given by \(s_k = \sum_{v_t \in S_k} \beta_t^{(v)} h_t^{(s)}\). Given the representation of segments \((s_1, s_2, \ldots, s_K)\), we devise the segment-level LSTM networks to learn their semantic representation, denoted by \((h_1^{(s)}, h_2^{(s)}, \ldots, h_K^{(s)})\). Therefore, the adaptive encoder network is given by \(f(v, q) = (h_1^{(s)}, h_2^{(s)}, \ldots, h_K^{(s)})\).

### 2.3 Reinforced Decoder Network Learning

In this section, we propose the reinforced decoder network \(g(\cdot)\) based on segment-level LSTM networks to generate the open-ended answer for video question answering.

Given the segment-level semantic representation of video contents \(f(v, q) = (h_1^{(s)}, h_2^{(s)}, \ldots, h_K^{(s)})\) from segment-level LSTM networks, the question-aware attentional LSTM networks predict the next word in answer by sampling \(a_t \sim p_\theta(a_t|a_{1:t-1}, f(v, q)) = g_t(f(v, q), h_t^{(s)}, e_t)\), where \(g_t(\cdot)\) is the recurrent answer generator at time \(t\). The \(h_t^{(a)}\) is the decoder state and \(e_t\) is the context vector at time \(t\). Given question representation \(h^{(q)}\), the segment-level attention score for the \(k\)-th segment and \(t\)-th decoder state for answer generation is given by

\[
a_k^{(a)} = P^{(a)} \text{tanh}(W_h^{(a)} h_k^{(s)} + W_q^{(a)} h^{(q)} + W_h^{(a)} h_t^{(a)} + b^{(a)}),
\]

where \(W_h^{(a)}, W_q^{(a)}, W_h^{(a)}\) are parameter matrices and \(b^{(a)}\) is the bias vector. The \(P^{(a)}\) is the parameter vector for computing the question-aware segment-level attention score. For each segment \(h_k^{(s)}\) with decoder state \(h_t^{(a)}\), its activation by the softmax function is given by \(\beta_k^{(a)} = \frac{\exp(a_k^{(a)})}{\sum_{k} \exp(a_k^{(a)})}\). Thus, the attentional question-aware semantic video representation at decoder state \(h_t^{(a)}\) is given by \(e_t = \sum_k \beta_k^{(a)} h_k^{(s)}\). One common approach to train the proposed decoder network is under the framework of maximum likelihood estimation, given by

\[
\mathcal{L}_{ML}(g(f(v, q))) = \sum_{t=1}^{M} \log p_\theta(a_t|a_{1:t-1}, f(v, q)).
\]

However, the training based on maximum likelihood estimation makes the learnt decoder network suboptimal [Bahdanau et al., 2016]. In this work, we train the proposed decoder network under the framework of reinforcement learning. In the setting of reinforcement learning, we define the generation of next answer word as action, and the decoding probability \(p_\theta(a_t|a_{1:t-1}, f(v, q))\) as the policy. Following the existing visual-semantic embedding works [Ren et al., 2017], we choose the reward function based on the embedding similarity between the ground-truth answer \(a\) and the generated answer \(a\), given by \(R_a(\hat{a}) = \| h^{(a)} - h^{(a)} \|^2 \). Specifically, we define the expected cumulative reward at each decoding step using value function by

![Figure 2: The Framework of Adaptive Hierarchical Reinforced Networks for Open-ended Long-form Video Question Answering.](image-url)
Table 1: Summary of Dataset

<table>
<thead>
<tr>
<th>Data</th>
<th>Object</th>
<th>Number</th>
<th>Color</th>
<th>Location</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>All</td>
<td>13,588</td>
<td>2,960</td>
<td>5,273</td>
<td>8,959</td>
<td>5,105</td>
</tr>
<tr>
<td>Train</td>
<td>10,121</td>
<td>2,228</td>
<td>3,642</td>
<td>6,303</td>
<td>3,705</td>
</tr>
<tr>
<td>Valid</td>
<td>1,335</td>
<td>287</td>
<td>547</td>
<td>336</td>
<td>509</td>
</tr>
<tr>
<td>Test</td>
<td>2,132</td>
<td>445</td>
<td>1,084</td>
<td>1,754</td>
<td>891</td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th>Method</th>
<th>Accuracy</th>
<th>WUPS@0.0</th>
<th>WUPS@0.9</th>
</tr>
</thead>
<tbody>
<tr>
<td>VQA+</td>
<td>0.254</td>
<td>0.4831</td>
<td>0.3546</td>
</tr>
<tr>
<td>MM+</td>
<td>0.2657</td>
<td>0.5162</td>
<td>0.3665</td>
</tr>
<tr>
<td>STAN</td>
<td>0.2641</td>
<td>0.5151</td>
<td>0.3696</td>
</tr>
<tr>
<td>STVQA</td>
<td>0.3022</td>
<td>0.5376</td>
<td>0.3879</td>
</tr>
<tr>
<td>AHN(ml)</td>
<td>0.3317</td>
<td>0.5696</td>
<td>0.4103</td>
</tr>
<tr>
<td>AHN(rtl)</td>
<td><strong>0.3429</strong></td>
<td><strong>0.6081</strong></td>
<td><strong>0.4313</strong></td>
</tr>
</tbody>
</table>

3.2 Performance Criteria

We evaluate the performance of our proposed AHN method based on two widely-used evaluation criteria for open-ended video question answering, i.e., Accuracy [Antol et al., 2015] and WUPS [Malinowski and Fritz, 2014]. Given the testing question \( q \in Q_t \) with its ground-truth answer \( a \), we denote the generated answers from our AHN method by \( \hat{a} \). We now introduce the evaluation criteria below.

- **Accuracy.** The Accuracy is the normalized criteria of accessing the quality of the generated answer based on the testing question set \( Q_t \), given by

\[
\text{Accuracy} = \frac{1}{|Q_t|} \sum_{q \in Q_t} (1 - \prod_{i=1}^{M} 1[\hat{a}_i \neq a_i]),
\]

where \( \text{Accuracy} = 1 \) (best) means that the generated answer and the ground-truth ones are exactly the same, while \( \text{Accuracy} = 0 \) means the opposite.

- **WUPS.** The WUPS is the soft measure based on the WUP [Wu and Palmer, 1994] score to evaluate the quality of the generated answer. The WUP measures word similarity based on WordNet [Fallbaum, 1998]. Thus, given the set of generated answer words \( O_\hat{a} = \{a_1, a_2, \ldots, a_M\} \) and the ground-truth ones \( A_q = \{a_1, a_2, \ldots, a_M\} \) for testing question \( q \), the WUPS score with the threshold \( \gamma \) is given by

\[
\text{WUPS} = \frac{1}{|Q_t|} \sum_{q \in Q_t} \min \{ \prod_{a_i \in A_q} \max_{o_j \in O_\hat{a}} \text{WUP}\gamma(a_i, o_j), \prod_{a_i \in O_\hat{a}} \max_{o_j \in A_q} \text{WUP}\gamma(o_i, a_j) \},
\]

where the \( \text{WUP}_\gamma(\cdot) \) score is given by

\[
\text{WUP}_\gamma(a_i, o_j) = \begin{cases} \text{WUP}(a_i, o_j) & \text{WUP}(a_i, o_j) \geq \gamma \\ 0.1 \cdot \text{WUP}(a_i, o_j) & \text{WUP}(a_i, o_j) < \gamma \end{cases}
\]

Following the experimental setting in [Malinowski and Fritz, 2014], we choose two WUPS evaluation criteria with the parameter \( \gamma \) to be 0 and 0.9, denoted by WUPS@0.0 and WUPS@0.9, respectively.

3.3 Performance Comparisons

We compare our proposed method with other four state-of-the-art methods for the problem of open-ended video question answering as follows:
Table 3: Experimental results on Accuracy with different types of visual questions.

<table>
<thead>
<tr>
<th>Method</th>
<th>Accuracy</th>
<th>Object</th>
<th>Number</th>
<th>Color</th>
<th>Location</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>VQA+</td>
<td>0.2469</td>
<td>0.7563</td>
<td>0.3089</td>
<td>0.133</td>
<td>0.1954</td>
<td></td>
</tr>
<tr>
<td>MM+</td>
<td>0.2537</td>
<td>0.7851</td>
<td>0.3217</td>
<td>0.143</td>
<td>0.2011</td>
<td></td>
</tr>
<tr>
<td>STAN</td>
<td>0.2515</td>
<td>0.7995</td>
<td>0.3452</td>
<td>0.125</td>
<td>0.2169</td>
<td></td>
</tr>
<tr>
<td>STVQA</td>
<td>0.3063</td>
<td>0.7918</td>
<td>0.3327</td>
<td>0.166</td>
<td>0.2829</td>
<td></td>
</tr>
<tr>
<td>AHN_{ml}</td>
<td>0.3596</td>
<td>0.8021</td>
<td>0.3417</td>
<td>0.189</td>
<td>0.3014</td>
<td></td>
</tr>
<tr>
<td>AHN_{rl}</td>
<td>0.3735</td>
<td>0.8055</td>
<td>0.3241</td>
<td>0.2285</td>
<td>0.2905</td>
<td></td>
</tr>
</tbody>
</table>

Table 4: Experimental results on WUPS@0.0 with different types of visual questions.

<table>
<thead>
<tr>
<th>Method</th>
<th>WUPS@0.0</th>
<th>Object</th>
<th>Number</th>
<th>Color</th>
<th>Location</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>VQA+</td>
<td>0.5489</td>
<td>0.9478</td>
<td>0.8311</td>
<td>0.1647</td>
<td>0.3017</td>
<td></td>
</tr>
<tr>
<td>MM+</td>
<td>0.5513</td>
<td>0.9602</td>
<td>0.8215</td>
<td>0.2715</td>
<td>0.3178</td>
<td></td>
</tr>
<tr>
<td>STAN</td>
<td>0.5612</td>
<td>0.9517</td>
<td>0.8421</td>
<td>0.251</td>
<td>0.321</td>
<td></td>
</tr>
<tr>
<td>STVQA</td>
<td>0.5847</td>
<td>0.9689</td>
<td>0.8632</td>
<td>0.3001</td>
<td>0.3657</td>
<td></td>
</tr>
<tr>
<td>AHN_{ml}</td>
<td>0.6154</td>
<td>0.9701</td>
<td>0.8712</td>
<td>0.3217</td>
<td>0.3874</td>
<td></td>
</tr>
<tr>
<td>AHN_{rl}</td>
<td>0.6234</td>
<td>0.9657</td>
<td>0.9042</td>
<td>0.4192</td>
<td>0.4091</td>
<td></td>
</tr>
</tbody>
</table>

Table 5: Experimental results on WUPS@0.9 with different types of visual questions.

<table>
<thead>
<tr>
<th>Method</th>
<th>WUPS@0.9</th>
<th>Object</th>
<th>Number</th>
<th>Color</th>
<th>Location</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>VQA+</td>
<td>0.3614</td>
<td>0.8165</td>
<td>0.4947</td>
<td>0.2039</td>
<td>0.2378</td>
<td></td>
</tr>
<tr>
<td>MM+</td>
<td>0.3741</td>
<td>0.8411</td>
<td>0.5201</td>
<td>0.2187</td>
<td>0.2412</td>
<td></td>
</tr>
<tr>
<td>STAN</td>
<td>0.3716</td>
<td>0.8315</td>
<td>0.5142</td>
<td>0.2101</td>
<td>0.254</td>
<td></td>
</tr>
<tr>
<td>STVQA</td>
<td>0.3959</td>
<td>0.8478</td>
<td>0.5514</td>
<td>0.2287</td>
<td>0.2578</td>
<td></td>
</tr>
<tr>
<td>AHN_{ml}</td>
<td>0.4214</td>
<td>0.8643</td>
<td>0.5743</td>
<td>0.2462</td>
<td>0.2847</td>
<td></td>
</tr>
<tr>
<td>AHN_{rl}</td>
<td>0.45</td>
<td>0.8762</td>
<td>0.5844</td>
<td>0.2782</td>
<td>0.2937</td>
<td></td>
</tr>
</tbody>
</table>

- VQA+ method is the extension of image question algorithm [Malinowski and Fritz, 2014], where the mean-pooling layer is added to encode the video.
- MM+ method [Zeng et al., 2017] is the extension of end-to-end memory network algorithm, which one-layer bi-LSTM network is added to encode the sequence of video frames for answer generation.
- STAN method [Zhao et al., 2017] is based on the hierarchical spatio-temporal attention network for learning the joint representation of the dynamic video contents according to the given question.
- STVQA method [Jang et al., 2017] is based on the spatio-temporal reasoning algorithm, which employ the spatial and temporal attention on video to answer questions.

Unlike the previous video question answering works, our AHN method learns the hierarchical attentional video representation with adaptive recurrent encoder networks, and then generates the natural language answer with reinforced decoder networks for open-ended video question answering. To exploit the effect of reinforced decoder networks, we denote the our AHN method with reinforced decoder networks by AHN_{rl} and the one without reinforced decoder networks by AHN_{ml}. The weights of both frame-level LSTM networks and segment-level LSTM networks are randomly by a Gaussian distribution with zero mean.

Table 2 shows the overall experimental results of the methods on all types of questions based on three evaluation criteria. Tables 3, 4 and 5 illustrate the evaluation results on Accuracy, WUPS@0.0 and WUPS@0.9 with different types of questions, respectively. The hyperparameters and parameters which achieve the best performance on the validation set are chosen to conduct the testing evaluation. We report the average value of all the methods on three evaluation criteria. The experiments reveal a number of interesting points:

- The methods based on LSTM networks, MM+, STAN, STVQA, AHN outperform the mean-pooling based method VQA+, which suggests that the sequential frame-level representation is critical for the problem.
- The attention based methods STAN, STVQA and AHN achieves better performance than other baselines. This suggests that the joint representation learning of video and question can also improve the performance of open-ended video question answering.
- In all the cases, our AHN method achieves the best performance. This fact shows that the hierarchical attentional video representation with adaptive recurrent encoder networks, and reinforced decoder networks are effective for the problem.

In our approach, there are three essential parameters, which are the dimension of hidden state in frame-level LSTM networks, the dimension of hidden state in segment-level LSTM networks and the threshold $\tau$ of the step function. We investigate the effect of these parameters on our method by varying both the dimension of hidden state in frame-level LSTM networks and segment-level LSTM networks from 128 to 1024, and the threshold $\tau$ of step function from 0.1 to 0.9 on Accuracy in Figures 3(a), 3(b) and 3(c). We then vary these parameters to show their effect on our method using WPUS@0.9 in Figures 4(a), 4(b) and 4(c). Our method achieves the best performance when the dimension of hidden state in frame-level LSTM networks is set to 512, the dimension of hidden state in segment-level LSTM networks is set to 256 and the threshold is set to 0.3.

4 Related Work

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

The existing approaches for visual question answering can be categorized into image-based question answering methods [Antol et al., 2015; Lu et al., 2016; Li and Jia, 2016; Malinowski and Fritz, 2014; Shih et al., 2016] and video-based question answering ones [Mazaheri et al., 2016; Tapaswi et al., 2016; Zhu et al., 2015; Zeng et al., 2016; Zhao et al., 2017]. Given a natural-language question for the image, the task of image-based question answering is to provide the accurate answer for the given question [Antol et al.,...
As a natural extension of image-based question answering, the video-based question answering has been proposed as a more challenging task [Zeng et al., 2016]. The fill-in-the-blank approaches [Zhu et al., 2015; Mazaheri et al., 2016] complete the missing entry in the video description by ranking candidate answers based on both visual content and contextual video description. Tapaswi et al. [Tapaswi et al., 2016] propose the three-way scoring function for movie question answering based on both the relevance between given question and textual movie subtitles, and textual movie subtitles and answers. Zhao et al. [Zhao et al., 2017] propose the hierarchical spatio-temporal attention networks for video question answering. Jang et al. [Jang et al., 2017] develop the spatio-temporal reasoning algorithm, which employ the spatial and temporal attention on video to answer questions. Zeng et al. [Zeng et al., 2017] extend the end-to-end memory network with additional LSTM layer for video question answering. Although these works have achieved promising 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 modeling the semantic representation of long-form video contents. Unlike the previous studies, we study the problem of open-ended long-form video question answering from the viewpoint of adaptive hierarchical network learning.

5 Conclusion

In this paper, we present the problem of open-ended long-form video question answering from the viewpoint of adaptive hierarchical reinforced encoder-decoder network learning. We first propose the adaptive hierarchical encoder network to learn the video representation from the critical segments of the targeted objects with adaptive video segmentation. We then develop the reinforced decoder network to generate the natural language answer for open-ended video question answering. We construct a large-scale long-form video question answering dataset and evaluate the effectiveness of our proposed method through extensive experiments.

Acknowledgments

This work was supported by the National Natural Science Foundation of China under Grant No.61602405, No.61702143, No.61622205 and No.61472110, Sponsored by CCF-Tencent Open Research Fund and the China Knowledge Centre for Engineering Sciences and Technology. This work is also Supported by Zhejiang Natural Science Foundation(LZ17F020001) and Key R&D Program of Zhejiang Province(2015C01027).
References


