Query-Biased Self-Attentive Network for Query-Focused Video Summarization

Shuwen Xiao, Zhou Zhao, Zijian Zhang, Ziyu Guan, and Deng Cai

***Abstract*— This paper addresses the task of query-focused video summarization, which takes user queries and long videos as inputs and generates query-focused video summaries. Compared to video summarization, which mainly concentrates on finding the most diverse and representative visual contents as a summary, the task of query-focused video summarization considers the user's intent and the semantic meaning of generated summary. In this paper, we propose a method, named query-biased self-attentive network (QSAN) to tackle this challenge. Our key idea is to utilize the semantic information from video descriptions to generate a generic summary and then to combine the information from the query to generate a query-focused summary. Specifically, we first propose a hierarchical self-attentive network to model the relative relationship at three levels, which are different frames from a segment, different segments of the same video, textual information of video description and its related visual contents. We train the model on video caption dataset and employ a reinforced caption generator to generate a video description, which can help us locate important frames or shots. Then we build a query-aware scoring module to compute the query-relevant score for each shot and generate the query-focused summary. Extensive experiments on the benchmark dataset demonstrate the competitive performance of our approach compared to some methods.**

***Index Terms*— Video summarization, vision and language, self-attention mechanism.**

I. Introduction

THE goal of video summarization is to generate a condensed synopsis which remains the important information from the original video and removes the trivial contents.

Manuscript received May 17, 2019; revised October 16, 2019 and February 2, 2020; accepted March 18, 2020. Date of publication April 10, 2020; date of current version April 28, 2020. This work was supported in part by the Zhejiang Natural Science Foundation under Grant LR19F020006, in part by the National Key Research and Development Program of China under Grant 2018AAA0101400, in part by the National Nature Science Foundation of China under Grant 61672409, Grant 61751209, Grant 61836002, Grant 61936006, and Grant U1611461, and in part by the Joint Research Program of ZJU and Hikvision Research Institute and China Knowledge Center for Engineering Sciences. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Senem Velipasalar. *(Corresponding author: Zhou Zhao.)*

Shuwen Xiao and Zijian Zhang are with the College of Computer Science, Zhejiang University, Hangzhou 310027, China (e-mail: 21721140@zju.edu.cn; ckczzj@zju.edu.cn).

Zhou Zhao is with the College of Computer Science, Zhejiang University, Hangzhou 310027, China, and also with the Alibaba-Zhejiang University Joint Research Institute of Frontier Technologies, Hangzhou 310058, China (e-mail: zhaozhou@zju.edu.cn).

Ziyu Guan is with the School of Information and Technology, Northwest University, Xi'an 710127, China (e-mail: ziyuguan@nwu.edu.cn).

Deng Cai is with the State Key Laboratory of CAD&CG, Zhejiang University, Hangzhou 310027, China (e-mail: dengcai@cad.zju.edu.cn).

Digital Object Identifier 10.1109/TIP.2020.2985868

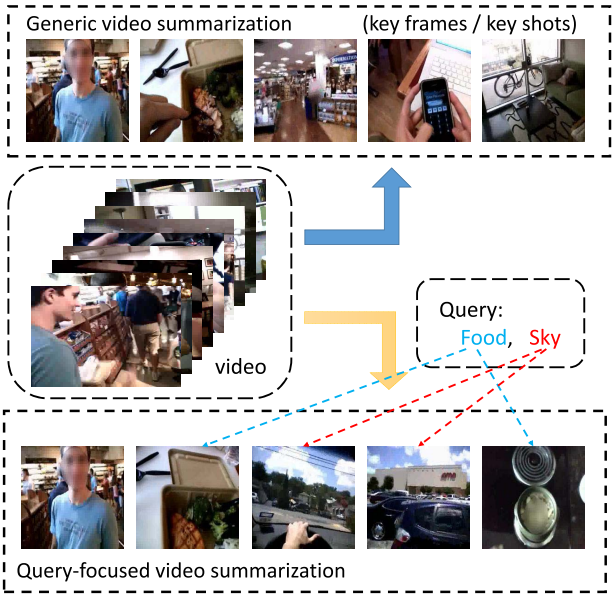


Fig. 1. The goal of generic video summarization is to select a compact subset of frames or shots that remains the important information from the original video, while the task of query-focused video summarization considers user's query and generates query-related video summaries. For example, the shots indicated by the blue arrows relate to the concept of food.

Automated video summarization is very helpful for applications such as action recognition, analysis of surveillance videos, and creation of visual diaries for personal lifelog videos.

The research directions in the task of video summarization can be divided into three aspects: domain-specific video summarization, generic video summarization and query-focused video summarization. The source videos in domain-specific video summarization are usually from some categories such as sports, which are more structured videos. Generic video summarization has been addressed at three level: shot-level [1], [2], frame level [3], [4] and object level [5]. These approaches select shots and frames with high interestingness score as the short summaries. However, both domain-specific video summarization and generic video summarization do not take user preferences into account. Moreover, the trained video summarizers cannot meet all the users' preferences and the performance evaluation is often to measure the temporal overlap, making it hard to capture the semantic similarity between summary and original videos. To address this

problem, Sharghi *et al.* [6] introduced a new task, named query-focused video summarization, which is taking the long video and user's query as inputs and generating query-oriented video summary. Query-focused video summarization can be useful in such scenarios when a user wants to customize a summary from his daily video logs.

There have been some researches studying the task of query-focused video summarization. In [7], they first employed a sequence scoring algorithm based on DPP(determinantal point process) which can extract frames that are representative and query-related. Later in [6], they extended SeqDPP [8] by combining memory network and constructed a new dataset for this task. The dataset is based on the existing UT Egocentric(UTE) dataset [9], which collects hours of egocentric video, containing a diverse set of events. However, these approaches are all trained on a limited number of data, for example, there are only four videos in the UTE dataset, which may lead to deviations and cannot be applied to other videos.

In this paper, we formulate the task of query-focused video summarization as a scoring problem. More specifically, with the additional input of users' queries, we first generate a general importance score for each video shot, then calculates the relevant score between each video shot and the given query. Finally, we obtain the query-relevant score for each video shot so that we can generate a query-related video summary. The main idea of this paper is that to eliminate the bias of parameters when the model is trained on a small dataset, we first train the model on another larger video dataset and then transfer the parameters to deal with the task of query-focused video summarization. We employ a novel method, named Query-biased Self-Attentive Network(QSAN), to generate query-related video summary.

Our model is consist of three parts: 1) a hierarchical self-attentive network to compute the general important score for each frame/shots; 2) a reinforced caption generator to construct the video description; 3) a query-aware scoring module (QS module) to select video content related to the given query and compute the query-relevant score for each video shot.

As shown in figure 2, given a video and its corresponding description, "*A man is playing the instrument*.", the part related to the description is the picture of *human* or *instrument*, which is also the part with more semantic information in the video, compared with the part of natural scenery. That is to say, given video descriptions and all segments of the video, the description-related segments or frames are more often to be semantically meaningful visual content. We propose a hierarchical self-attentive network to deal with both segment-level and frame-level visual features and select important frames. The selected frames are weighted by the importance score and then used to construct the video descriptions through a reinforced caption generator. By reducing the distance between generated sentence and video description, the model can capture the relationship between important visual content and textual information from segment-level and frame-level. These two modules are jointly trained and can compute the importance score for each video shot from intra-segment and inter-segment level. Then we transfer the parameters in the HSAN to the task of query-focused video summarization and we propose a query-aware scoring module, which can handle the modality between video and text query and compute the similarity score for each shot. This module will return a relevant score given a video shot and a query. Finally, we obtain the query-biased score and then generate a query-focused summary for a long video. Experiments on the benchmark datasets [6], which include videos showing events in the first-person perspective, show the ability of our model to locate key contents associated with queries.

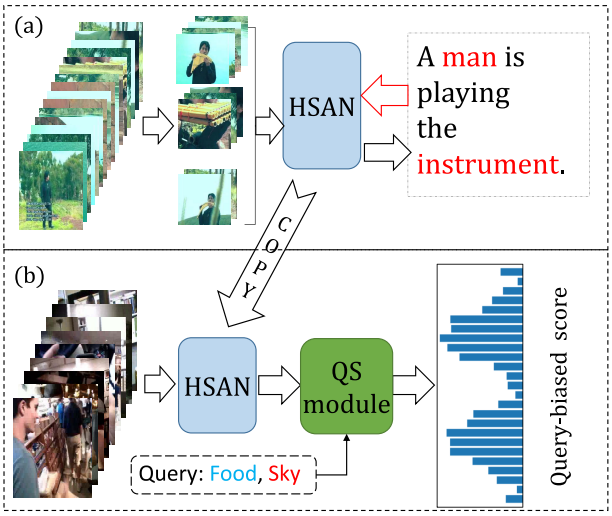


Fig. 2. Our network is first trained on video description dataset to learn the semantic relationship between visual contents and textual information. Then the network is transferred and the query information is added as an input, to generate query-related video summaries for the task of query-focused video summarization.

The main contributions of this paper are summarized as follows:

- Instead of training the model on limited data, we utilize a large video dataset to train our model for getting richer semantic information and then transfer the trained parameters to the task of query-focused video summarization.
- We present the hierarchical self-attentive network to learn the frame-level and segment-level semantic representation of video content. A reinforced caption generator is introduced to generate descriptions given the video summary and then measure the distance between generated descriptions and ground truth. We utilize reinforcement learning to optimize the model. Through these two modules, we can compute the general importance score for each frame in the video.
- We employ a query-aware scoring module which takes video shots and query as input and then selects the most query-related shots. We can compute the query-relevant score and finally generate the query-related video summary.
- We perform extensive experiments on several benchmark datasets to demonstrate the effectiveness of our model.

The rest of this paper is organized as follows. In Section II, we provide a review of the related work about generic video summarization and query-focused video summarization. In the

following, Section III introduces each component of our approach. A variety of experimental results are presented in Section IV. Finally, we provide some concluding remarks in Section V.

II. Related Work

In this section, we will introduce some related works on the task of generic video summarization, query-focused video summarization and video description model.

A. Generic Video Summarization

The task of generic video summarization has been studied for years and several approaches have been proposed so far. They can be roughly divided into three aspects: unsupervised, supervised and weakly supervised methods.

Unsupervised video summarization approaches [1], [3], [9]–[20] makes use of specific selection criteria to measure the importance or interestingness of video content and then generate frame-based or shot-based video summaries. These methods have conventional methods, that is, methods that use some low-level video features for calculation, as well as more recent methods, which combine deep learning. Conventional unsupervised video summarization approaches mostly use hand-crafted heuristics [13], [21] (such as diversity and representativeness) or frame clustering method [16], [18], [22] to decide whether to choose a video frame as a keyframe. Some methods [3], [23]–[26] use the web images as auxiliary information to generate user-oriented video summaries, by leveraging the extra visual information brought by web images which contains people's interest. In [27], Kanehira *et al.* take $viewpoint$ into consideration and calculate the video-level similarity between multiple videos to generate video summary. More recently, some methods based on deep learning have been proposed. In [12], Mahasseni *et al.* propose a generative adversarial network to select a subset of keyframes by minimizing the distance between the information of generated summaries and original videos. Zhou *et al.* [15] propose a reinforcement learning algorithm to summarize videos, by designing the diversity-representativeness reward and applying algorithm based on policy gradient to optimize the model.

Researches on supervised video summarization methods [4], [28]–[36] have appeared in recent years. These methods are trained on video content with human-created ground-truth annotations. With the semantic information from ground-truth annotations, these methods usually capture the video content with more semantic information and achieve better performance than unsupervised methods. In [28], video summarization is formulated as an interestingness scoring problem and frames with higher scores are selected as summaries. Some approaches utilize the additional information from web images [3], categories [29], and titles [30] to improve the quality of video summaries. In [31], Gygli *et al.* propose an adapted submodular function with multiple objects to tackle the problem of video summarization. Yao *et al.* [33] employ a deep rank model, taking a pair of highlight segment and non-highlight segment as inputs, to rank each segment. In [4], Zhang *et al.* implement the dppLSTM model by combining determinantal point process and bi-directional LSTM to predict whether a frame is a keyframe. In [34], Rochan *et al.* propose a fully convolutional sequence network, of which the structure is based on convolutional layer and deconvolutional layer, consequently the network runs in parallel. In [35], they improve SeqDPP method with large-margin algorithm and design a new probabilistic distribution. Although most supervised video summarization methods achieve better performance, there are still some shortcomings. Firstly, it is time-consuming and labor-intensive to label ground-truth annotations for a video summarization dataset. Secondly, models trained on annotated data can lead to overfitting and be difficult to generalize, but in reality, results returned by the video summary generator should not be unique. To overcome these weaknesses, some weakly supervised methods have been proposed. These methods typically utilize readily available label, such as video categories, as additional information to improve model performance. In [25], a 3D ConvNet is trained to predict the category of the video. The importance score of each frame is obtained by calculating the back-propagated gradient returned by the true category. In [37], an encoder-attention-decoder structure is built, where the encoder learns the latent semantics from web videos and the decoder generate the summary. Compared to weakly labels, video descriptions contain more semantic information which can help the model to generate the story-telling summary.

B. Query-Focused Video Summarization

Compared to the generic video summarization, the task of query-focused video summarization takes user's query into account. Specifically, the dataset provides concept annotations, which are able to describe the image, for each shot in the video and convey more semantic information than the general one.

Some previous works take semantics into consideration when dealing with the task of video summarization. In [38], they label video shots with texts and generate a text representation for each video based on the generated summary, then measure its semantic distance to ground truth. In [39], they create video descriptions for videos in summarization datasets and train model based on the descriptions to leverage the additional textual information. In [7], Sharghi *et al.* introduce a DPP-based algorithm, which takes user query as input and generates query-focused video summary. In [6] explore query-focused video summarization, which generates a summary based on video content and user query and proposes a memory network based model. Vasudevan *et al.* [40] introduce a quality-aware relevance model with submodular maximization to select important frames. In [41], the author proposes generative adversarial network to tackle this challenge, where the model will generate a query-focused summary, a random summary. With the ground truth summary, a three-player loss is introduced to optimize the model. To the best our knowledge, we are the first to propose an approach based on self-attention mechanism in query-focused video summarization tasks. Moreover, our model is first trained on video description dataset to model the semantic relationship between

visual contents and textual information and then transfer to the summary task, which makes the model be exposed to acquire sufficient semantic information so as to produce high-quality generic summary. Then the query-aware scoring module can be useful to generate query-related video summary.

C. Video Description Task

In the task of describing videos, the most used structure is Encoder-Decoder architecture. [42]. The decoder is usually recurrent. Attention mechanism over input features is an exciting branch in the video captioning task. In [43], Xu *et al.* introduced an attention-based model that learns to focus on salient objects to generate caption words. Yu *et al.* [44] proposed a hierarchical recurrent neural networks to generate paragraph for video. There are some works [45]–[47] utilize reinforcement learning algorithms to tackle the challenge. In [48], Chen *et al.* proposed a PickNet, which select important frames then use them to generate caption words.

Different from these works, we proposed a hierarchical self-attention network to model visual object and its related textual information. Then we transfer the parameters back to the task of video summarization and generate semantically rich summaries.

III. The Proposed Method

Our method is consist of the following components: 1) the hierarchical self-attentive network(HSAN) learns the congruent relationship between video contents and its corresponded semantic information from both frame-level and segment-level, which can generate the generic video summary; 2) the reinforced attentive description generator which can enhance the relationship between textual information and visual contents by reducing the distance between the generated description and the ground truth description; 3) the query-aware scoring module(QSM) which computes the relevant score for each segment to the given query and produces the query-related video summary.

A. Problem Formulation

In order to gain sufficient semantic information from a larger video dataset, we train the parameters of our model on a video description dataset. Let $\mathbf{v}$ and $\mathbf{w}$ denote a video and its related description, respectively. We denote the video as a sequence of frames $\mathbf{v} = \{v_1, v_2, v_3, \ldots, v_T\}$, where v_i presents the i-th frame from the video and T is the length of video. The video description is a sentence, which is presented as a sequence of word $\mathbf{w} = \{w_1, w_2, w_3, \ldots, w_N\}$, where w_j is the j-th word of the description and N is the length of sentence. For the training process, we utilize the video information to generate a natural sentence that describes the video. To compute the query-relevant score for each segment, we use the concept annotations in dataset [6] to train our Query-Aware Scoring Module. When inputs are a query $\mathbf{q}$, which is composed of two concept $\{\mathbf{c_1}, \mathbf{c_2}\}$ and the visual feature of video segments $\{S_1, S_2, \ldots, S_K\}$, this module will compute the concept-relevant score for each segment then we merge these scores as the query-relevant score. Finally, we can make use of the query-relevant score to produce a diverse subset of video segments that can not only represent the origin video but related to the query.

Input: A long video $\mathbf{v}$ and a query $\mathbf{q}$.

Output: A diverse subset of video shots that remains the origin information of original video and each shot in the subset is related to the given query.

B. Hierarchical Self-Attentive Network

In this part, we will introduce the first component of our model, named hierarchical self-attentive network(HSAN), which encodes the video features from frame-level and segment-level and learns the internal relationship among visual contents. The framework of hierarchical self-attentive network is presented in figure 3.

Due to the hierarchical structure of video, a long video consists of several stories, and a story is composed of multiple segments, where each segment contains many frames. Each visual content has its corresponding semantics. We have to learn the representation of video from different levels in order to better understand the semantic information of video. For this purpose, we first extract each the visual feature of each frame using the pretrained deep convolutional network. We denote the frame-level representation for video $\mathbf{v}$ by $\mathbf{v}^{(f)} = (\mathbf{v}_1^{(f)}, \mathbf{v}_2^{(f)}, \ldots, \mathbf{v}_T^{(f)})$. Then we split the video $\mathbf{v}$ into a set of segments $\{S_1, S_2, \ldots, S_K\}$, where S_i is the i-th segment from the video. The dimension of the video representation is d_f. After the video segmentation process, we can get K video segments, the length of video segment $\mathbf{S}_i$ is defined as s_i. The $\mathbf{v}_i^{(f)} \in \mathbf{S}_k$ means the i-th frame is in the k-th segment set. We next denote the segment-level semantic representation of video $\mathbf{v}$ by $\mathbf{v}^{(s)} = (\mathbf{v}_1^{(s)}, \mathbf{v}_2^{(s)}, \ldots, \mathbf{v}_K^{(s)})$, where $\mathbf{v}_i^{(s)}$ is the segment-level semantic embedding of the i-th segment. The word-level representation of natural language description, given by the pretrained word embedding model, is denoted by $\mathbf{h}^w = (\mathbf{h}_1^w, \mathbf{h}_2^w, \ldots, \mathbf{h}_N^w)$, where N is the number of words in description $\mathbf{w}$.

First of all, we need to split the video into several segments. There are some heuristic algorithms like uniform segmentation and the method based on visual feature [23]. We construct video segments using the Kernel Temporal Segmentation [29], which preserves visual consistency in each segment. The average duration of segments is 5 seconds.

After getting the frame-level features from the previous process, we propose a hierarchical self-attentive network to learn the relationships of visual content from frame-level and segment-level. As shown in figure 3, the network first takes frame-level features as inputs, then learns the relative semantic relationship between different frames in the same segment and forms more cohesive features through the self-attention mechanism. The module will aggregate features into segment-level features. Then the global self-attention module obtains the output from the previous module and learns the relative semantic relationship between all segments in the video, the same as the local self-attention module.

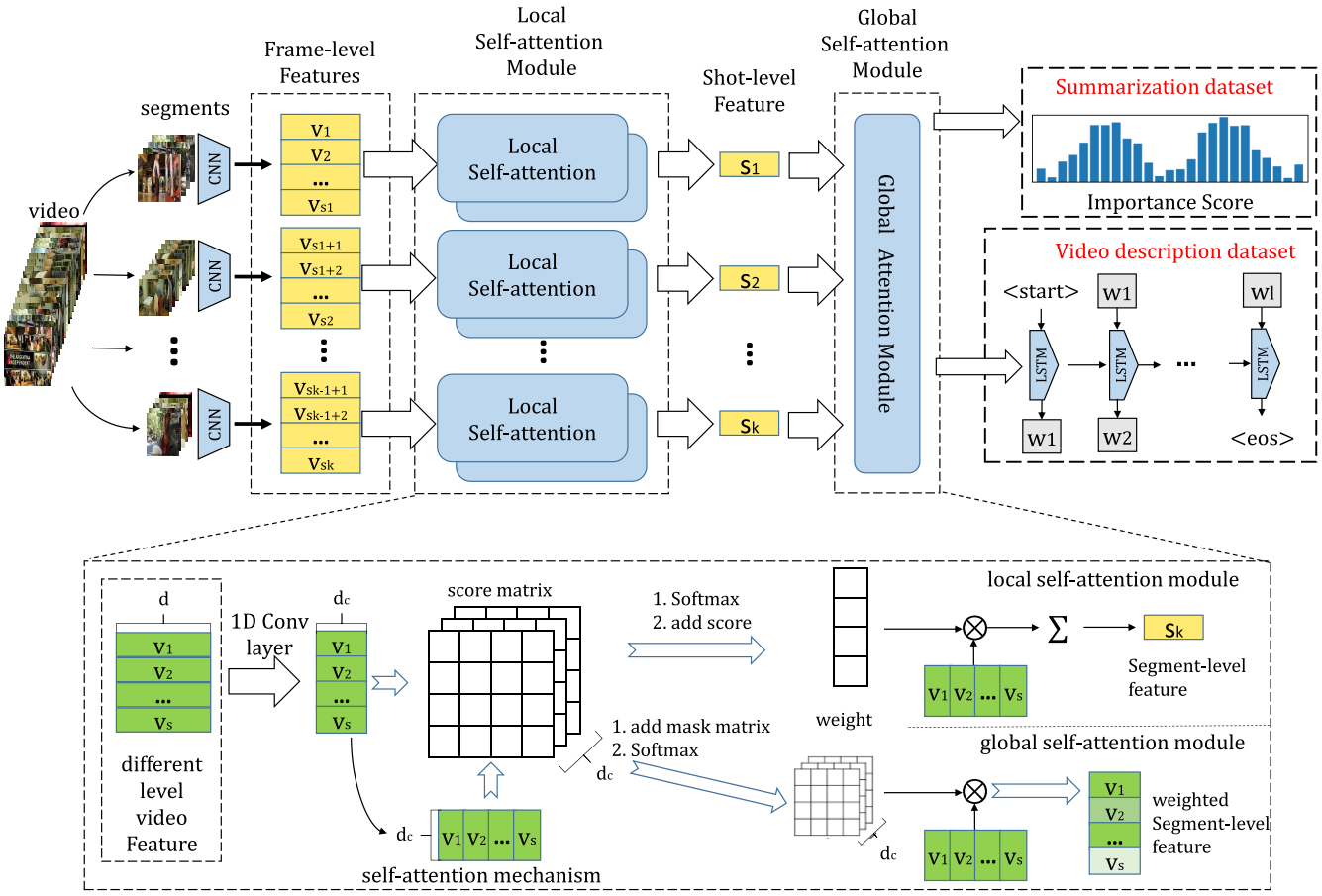


Fig. 3. The Framework of hierarchical self-attentive network. First, we split the video into segments and use pretrained CNN to extract video features. Then we propose a local self-attention module and a global self-attention module to capture the semantic relationship from both frame-level and segment-level. The input of local self-attention module is frame-level feature and the output of local self-attention module is segment-level feature. Then outputs of the global self-attention module are sent to a reinforced caption generator to learn the relation of caption and its related visual content. For video summarization, the model can generate importance score for each frame or each segment.

The local self-attention module is designed to capture the semantic relationship between all frames in the same segment. The detailed structure is presented at the bottom of figure 3. Given the sequence of frame-level representation $(\mathbf{v}_1^{(f)}, \mathbf{v}_2^{(f)}, \ldots, \mathbf{v}_{s_k}^{(f)})$ from the segment $\mathbf{S}_k$, in order to reduce the dimension of input feature, which can decrease the running memory of the model, we establish a 1D convolutional layer on the temporal dimension of the feature. After the convolution operation, the dimensions of visual information in the feature sequence will be reduced to d_c. The output vectors contain visual information from the original features and the extra information from other features in the sliding window. Then we can compute the alignment score matrix. Given the output vector of i-th frame and j-th frame from convolutional layer, which we note as $\mathbf{o}_i$ and $\mathbf{o}_j$, the alignment score vector is given by

$$f(\mathbf{o}_i, \mathbf{o}_j) = \mathbf{P}^{(f)} \tanh(\mathbf{W}_1 \mathbf{o}_i + \mathbf{W}_2 \mathbf{o}_j + \mathbf{b}), \tag{1}$$

where $\mathbf{P}^{(f)}, \mathbf{W}_1, \mathbf{W}_2 \in \mathbb{R}^{d_c \times d_c}$ are the trainable parameters, $\mathbf{b} \in \mathbb{R}^{d_c}$ is bias vector. And the shape of output alignment score matrix is $s_k \times s_k \times d_c$, which means the module can focus on the frames from the viewpoint of each feature. We propose the module to learn the relative semantic relationship of different frames in the same segments, and for different segments, the relation structure should be similar. Therefore, for different segments, our local self-attention modules share all the trainable parameters, which also reduces the amount of parameters in our model.

TABLE I
SUMMARY TABLE OF PARAMETERS

Parameters	Notation
Frame-level visual features	$\mathbf{v}^{(f)}$
Frame-level score (intra-segment)	$s^{(f)}$
Segment-level visual feature	$\mathbf{v}^{(s)}$
Integrated segment-level visual feature	$\hat{\mathbf{v}}^{(s)}$
Segment-level score (inter-segment)	$s^{(s)}$
Weighted segment-level visual feature	$\mathbf{v}^{(g)}$
Generated caption words from step $t+1$ to N	$\hat{\mathbf{w}}_{t+1:N}$

For each frame $\mathbf{v}_t^{(f)} \in \mathbf{S}_k$, we then normalize the alignment matrix with the softmax function among the temporal axis. The attention score vector is given by

$$\gamma_{ij} = \frac{\exp(f(\mathbf{o}_i, \mathbf{o}_j))}{\sum_{t \in s_k} \exp(f(\mathbf{o}_i, \mathbf{o}_t))}, \tag{2}$$

The elements in attention score vector $\gamma_{ij} \in \mathbb{R}^{d_c}$ mean the attention score of i-th frame to j-th frame on each feature and $\gamma \in \mathbb{R}^{s_k \times s_k \times d_c}$. Therefore, we compute the sum of the score matrix among the temporal axis and visual feature axis, which can represent the importance score for frame j, and we use softmax function to normalize it. Therefore, the importance score $s_j^{(f)}$ for frame j is computed by

$$\gamma_j = \sum_{t \in s_k} \sum_{d_e} \gamma_{tj}, \tag{3}$$

$$s_j^{(f)} = \frac{\exp(\gamma_j)}{\sum_{t \in s_k} \exp(\gamma_t)}, \tag{4}$$

And we can compute the segment-level feature $\mathbf{v}_k^{(s)}$ for segment $\mathbf{S}_k$ as

$$\mathbf{v}_k^{(s)} = \sum_{\mathbf{v}_t^{(f)} \in \mathbf{S}_k} s_t^{(f)} \mathbf{v}_t^{(f)} \tag{5}$$

After getting the segment-level representation $\mathbf{v}^{(s)} = (\mathbf{v}_1^{(s)}, \mathbf{v}_2^{(s)}, \ldots, \mathbf{v}_K^{(s)})$, where K is the number of segments. We employ a global self-attention module to learn about the inter-segment relationship among each segment. We build another 1D convolutional layer to reduce the dimension of feature, where the sliding window is set to 1. Given the sequence of segment-level representation, we compute the alignment score for every pair in the sequence similar to the process in the local self-attention module. For example, the alignment score for the representation of segment $\mathbf{S}_i$ and segment $\mathbf{S}_j$ is given by

$$f(\mathbf{o}_i^{(s)}, \mathbf{o}_j^{(s)}) = \mathbf{P}^{(s)} \tanh(\mathbf{W}_1 \mathbf{o}_i^{(s)} + \mathbf{W}_2 \mathbf{o}_j^{(s)} + \mathbf{b}), \tag{6}$$

where $\mathbf{P}^{(s)}, \mathbf{W}_1, \mathbf{W}_2 \in \mathbb{R}^{d_c \times d_c}$ are the trainable parameters, $\mathbf{b} \in \mathbb{R}^{d_c}$ is bias vector.

Inspired by the work in [49], we encode the segment features with directional information, by adding the alignment score matrix with positional mask matrix M^{fw} and M^{bw}, which are defined as:

$$M_{ij}^{fw} = \begin{cases} 0, & i \le j \\ -\infty, & i > j \end{cases} \tag{7}$$

$$M_{ij}^{bw} = \begin{cases} 0, & i \ge j \\ -\infty, & i < j \end{cases} \tag{8}$$

Following the same process as the local self-attention module, we can calculate the self-aligned score matrix for each segment with softmax function. Therefore the integrated segment-level feature $\hat{\mathbf{v}}_t^{(s)}$ for segment $\mathbf{S}_t$ can be computed as,

$$\gamma_{ij} = \frac{\exp(f(\mathbf{o}_i^{(s)}, \mathbf{o}_j^{(s)}) + M_{ij})}{\sum_{t \in K} \exp(f(\mathbf{o}_i^{(s)}, \mathbf{o}_t^{(s)}) + M_{it})} \in \mathbb{R}^{d_c}, \tag{9}$$

$$\hat{\mathbf{v}}_t^{(s)} = \sum_{i \in K} \gamma_{it} \odot \mathbf{o}_i^{(s)}, \tag{10}$$

The global self-attention module with forward and backward direction do not share the parameters, so we concatenate the output vectors on the axis of feature and denoted as $\hat{\mathbf{v}}_i^{(s)}$, which is the outputs representation for i-th segment. With the bi-directional self-aligned features for the sequence of segments, we can calculate the importance score for segment $\mathbf{S}_i$, which is given by,

$$s_i^{(s)} = \sigma(\mathbf{P} \cdot [\mathbf{W}_1 \hat{\mathbf{v}}_i^{(s)} + \mathbf{b}_1] + b), \tag{11}$$

where $\mathbf{W}_1 \in \mathbb{R}^{d_c \times d_c}$, $\mathbf{P}, \mathbf{b}_1 \in \mathbb{R}^{d_f}$ and b are trainable parameters. The score s_i presents the importance score for segment $\mathbf{S}_i$. We multiply the segment-level score by the output score from the local self-attention module. For example, we can obtain the summary score $s_i = Norm(s_i^{(f)} \times s_j^{(s)})$ for the i-th frame in the j-th segment, where $Norm$ is a function to normalize the score into 0 to 1 range.

C. Reinforced Caption Generator

In this section, we present the reinforced caption generator module, which is to generate natural language descriptions for a video summary. With the reward send back to the hierarchical self-attentive network, our network will be optimized and measure the association between visual content and textual information.

We multiply the importance score with the integrated segment-level features to get weighted segment features. As shown in figure 3, unlike the video summarization task, the output of global self-attention module with video description datasets is a sequence of weighted segment features, which contain both the visual information of video segment and the summary information. We denote them as $\mathbf{v}^{(g)} = (\mathbf{v}_1^{(g)}, \mathbf{v}_2^{(g)}, \ldots, \mathbf{v}_K^{(g)})$, where $\mathbf{v}_i^{(g)} = s_i^{(s)} \cdot \hat{\mathbf{v}}_i^{(s)}$ and K is the number of segments.

Now we introduce the structure of our generator. The generator is LSTM-based structure. Given the weighted segment-level features, the recurrent LSTM generates the next word in the dictionary by sampling $w_t \sim p_\theta(w_t | [\mathbf{w}_{1:t-1}, \mathbf{h}_t^{(w)}, \mathbf{c}_t])$, where $\mathbf{h}_t^{(w)}$ is the LSTM state and $\mathbf{c}_t$ is the fused context vector at t step. The context vector $\mathbf{c}_t$ is a fused vector based on the LSTM state $\mathbf{h}_t^{(w)}$ and the weighted features$\mathbf{v}^{(g)}$, which represents the description-aware visual feature. The attention score for the segment $\mathbf{S_k}$ is given by

$$\beta_{t,j}^{(w)} = \mathbf{P}^{(w)} tanh(\mathbf{W}_g \mathbf{v}_j^{(g)} + \mathbf{W}_h \mathbf{h}_t^{(w)} + \mathbf{b}^{(w)}), \tag{12}$$

where $\mathbf{P}^{(w)}$ is the parameter vector for computing attention score for the generator, $\mathbf{W}_g$, $\mathbf{W}_h$, $\mathbf{b}^{(w)}$ are parameter matrices and bias vector. At each time step t, given the decoder state $\mathbf{h}_t^{(w)}$, we normalize its attention scores using softmax function over segments and then compute the context vector. The attention score and the context vector at the step t are given by

$$\gamma_{t,j}^{(w)} = \frac{\exp(\beta_{t,j}^{(w)})}{\sum_{j \in K} \exp(\beta_{t,j}^{(w)})}, \tag{13}$$

$$\mathbf{c}_t = \sum_{j \in K} \gamma_{t,j}^{(w)} \mathbf{v}_j^{(g)}, \tag{14}$$

We train the caption generator with reinforcement learning [50]. We choose the reward function based on BLEU [51], to measure the semantic similarity between the generated description $\hat{\mathbf{w}}$ and the ground-truth video caption $\mathbf{w}$,

and denote it by $R(\hat{\mathbf{w}}) = \text{BLEU}(\hat{\mathbf{w}}, \mathbf{w})$. The expected reward at t-th step is given by $Q(\hat{w}_t|\hat{\mathbf{w}}_{1:t-1}, \mathbf{v}^{(g)}) = E_{p_\theta(\hat{\mathbf{w}}_{t+1:M}|\hat{\mathbf{w}}_{1:t})} R(\hat{\mathbf{w}})$. Since the expectation is hard to calculate with an exponential search process, we estimate it by aggregating the Monte-Carlo simulation at each step, given by

$$Q(\hat{w}_t|\hat{\mathbf{w}}_{1:t-1}, \mathbf{v}^{(g)})) \approx \begin{cases} \frac{1}{J}\sum_{n=1}^{J} R([\hat{\mathbf{w}}_{1:t}, \hat{\mathbf{w}}_{t+1:M}^{(n)}]), & t < l \\ R([\hat{\mathbf{w}}_{1:t-1}, \hat{\mathbf{w}}_t]). & t = N \end{cases} \tag{15}$$

The $\{\hat{\mathbf{w}}_{t+1:N}^{(1)}, \hat{\mathbf{w}}_{t+1:N}^{(2)}, \ldots, \hat{\mathbf{w}}_{t+1:N}^{(J)}\}$ is the set of generated words with J times sampling, which are randomly sampled starting from the $t+1$-th step. And then the gradient can be given as,

$$\nabla_\theta L(\theta) \approx \sum_{t=1}^{T} Q(\hat{w}_t|\hat{\mathbf{w}}_{1:t-1}, \mathbf{v}^{(g)}))\nabla_\theta log(p_\theta(w_t)) \tag{16}$$

When training the HSAN with policy gradient, in order to prevent high variance of gradient, a baseline reward is usually considered. As a result, the gradient is given as,

$$\nabla_\theta L(\theta) \approx \sum_{t=1}^{T} (Q_t - b)\nabla_\theta log(p_\theta(w_t)) \tag{17}$$

where Q_t is the estimated reward at time step t and b is obtained by the inference process at test time.

D. Query-Aware Scoring Module

In this part, we will introduce our query-aware scoring module, which is proposed for the task of query-focused video summarization. In this specific task, the video is represented as a sequence of video shots, where video shot is a small clip of video. Therefore, we design the query-aware scoring module, which can compute concept-relevant score based on the relevance between the video shot and user's query.

Our goal is to learn a model that can select the most related shots given a concept by returning a concept-relevant score. The module takes the concatenating result of shot-level features and segment-level features, which is denoted as $\mathbf{v}_i^{(q)} = [\mathbf{v}_i^{(f)}, \hat{\mathbf{v}}_j^{(s)}]$, where the i-th shot is belong to the j-th segment, and their related concepts as inputs. Moreover, each shot is related to one or multiple concepts. Given a specific concept c, we first obtain its embedding feature f_c using pretrained language model. Given a concept feature f_c and the representation of i-th video shot, we first calculate their distance-based similarity by

$$d_i = \mathbf{W}_s \mathbf{v}_i^{(q)} \odot \mathbf{W}_c f_c \tag{18}$$

where $\mathbf{W}_s$ and $\mathbf{W}_c$ are the parameter matrices which project the visual features and textual features into the same vector space. Then we let the output pass a MLP and get the concept-relevant score between i-th video shot and concept c. In the dataset [6], each query q contains two concepts. Therefore we take the average of two concept-relevant score as the query-relevant score $s^q = \{s_1^q, s_2^q, \ldots, s_T^q\}$.

For the training process, given the concept-relevant score for each shot, which is denoted as $s^c = \{s_1^c, s_2^c, \ldots, s_T^c\} \in [0, 1]$ and the ground truth annotations $\hat{s} = \{\hat{s}_1, \hat{s}_2, \ldots, \hat{s}_T\} \in \{0, 1\}$ which represents whether the video shot is related to the concept, the loss can be formulated as a cross-entropy loss:

$$L_{qs}(\theta) = \frac{1}{T}\sum_{t=1}^{T} \hat{s}_t log s_t^c + (1 - \hat{s}_t) log(1 - s_t^c) \tag{19}$$

By minimizing the loss, the query-aware scoring module can focus on the most concept-related video shots. The query is composed of several concepts. We take the average of concept-relevant score of contained concepts as the query-relevant score.

Algorithm 1

Require: Training pairs(Video, Description) from video description dataset, training pairs(Video, Query, Summary annotation) from query-focused video summarization dataset, Hierarchical Self-attentive Network (HSAN), Caption Generator g, Query-Aware Scoring Module (QSM)

Initialize the model parameters in HSAN and QSM with random weights
Load pre-trained CNN network, training pairs, splits videos into several segments and extract frame-level visual features
for each iteration $= 1$, M_1 **do**
  Randomly sample a minibatch from video description dataset
  Run a forward with HSAN to obtain the importance score for each frame
  Using weighted video features to generate video captions
  for t in 1:T **do**
    Compute reward by Eq. 15
    Calculate the loss between generated sentences and ground truth
    Update the caption generator g and HSAN via policy gradient
  end for
end for
Transfer the parameters in HSAN for query-focused video summarization
for each iteration $= 1$, M_2 **do**
  Randomly sample a minibatch from query-focused video summarization dataset
  Computing query-relevant score for each frame
  Calculate the loss by Eq. 19 and optimize the parameters in QSM
end for
Generate video summary based on the query-relevant score

IV. Experiment

In this section, we first introduce the datasets and preprocessing methods, then we conduct several experiments on these datasets.

A. Datasets and Experiment Settings

1) Datasets: We conduct the experiments on the query-focused video summarization dataset proposed in [6], which

is built upon the UT Egocentric(UTE) dataset [9]. The UTE dataset is composed by videos taken from the first person perspective. The total number of videos in the dataset is 4 and the video duration ranges from 3 hours to 5 hours, which contains different daily life scenarios. Sharghi *et al.* [6] provide 48 concepts as user's queries for the dataset. The concepts are concise and diverse, which are related to some common objects in our daily life. Each query contains two concepts and there are 46 queries in the dataset. As for queries, there are four different scenarios in this task [6], that is, 1) all concepts in the query appear in the same video shot, 2) all concepts in the query appear in the video but not in the same shot, 3) some of the concepts in the query appear in the video, 4) none of the concepts in the query appear in the video. It is worth mentioning that the fourth scenario is to some extent the same as general form video summarization. The dataset provides per-shot annotation, from which each shot is labeled with several concepts. We first train our model on the ActivityNet dataset [52], [53]. The ActivityNet data contains 20,000 videos amounting to 849 hours and 100K descriptions. The average time duration of ActivityNet videos is around 180 seconds and the longest video runs for over 10 minutes. In a certain video, there are several segments which have corresponding descriptions.

2) Experiment Settings: We preprocess the videos in query-focused video summarization dataset as follow. We first sample the video to 1 fps and then reshape each frame into size of 224×224. Next, we utilize the pretrained ResNet [54], which is pretrained on ImageNet [55], to extract the visual representation of each frame and take the 2048-dimensional vector for each frame as the frame-level features. Following the experimental setting in [6], we split the video into a set of video segments with a duration of 5 seconds.

Similar to the preprocessing of query-focused video summarization datasets, we sample the videos in the ActivityNet dataset to 5 fps and get the frame representation using the ResNet [54]. Then we pick out the video clips with captions and remove those clip with duration less than 20 seconds. We also need to clean the caption information by filtering out those clips with video description less than 5 words and more than 20 words. Next, we replaced the words with less than five occurrences with <UNK> and add the token <EOS> for each sentence. In order to extract the semantic representation of caption words, we employ a pretrained word2vec model. Specifically, the size of the vocabulary set is 3955, and the dimension of word vector is 300.

3) Evaluation Protocols: We follow the experiment setting in [6] for fair comparison. Considering that the video summary should be a relatively subjective result rather than a unique result, the author provides a dataset with dense per-shot annotations. They measure the performance of the model by calculating semantic similarity between different shots, rather than just measuring temporal overlap or relying on low-level visual features. In [6], they first calculate the conceptual similarity between each two video shots based on the intersection-over-union(IOU) of their related concepts. Then they use the conceptual similarity as the edge weights in the bipartite graph between two summaries instead of low-level visual features that do not contain the semantic information to find the maximum weight matching of the graph. Finally, precision, recall and F1 score can be computed based on the number of matching pairs.

4) Baselines: In our evaluation, we compare our proposed method with other query-focused video summarization approaches as follows:

- **SeqDPP** [8] which formulates video summarization as a subset selection problem and use submodular maximization to found a good summary. The original approach does not consider user queries.
- **SH-DPP** [7] is the extension of SeqDPP method, where the author adds an extra layer in the process of SeqDPP to judge whether a video shot is related to a given query.
- **QC-DPP** [6] is the extension of SeqDPP method. In this approach, the author introduces a memory network to parameterize the kernel matrix.
- **TPAN** [41] is three-player adversarial network. The author uses generative adversarial network to tackle the task and introduce a random summary as an extra adversarial sample.

B. Implementation Details

In order to obtain rich semantic information, we first train our hierarchical self-attentive network(HSAN) on the video description dataset. We compress the dimensions of the visual features into 512 by the 1D convolutional layer and the dimension of the hidden state of the LSTM in the generator is set to 1024. Then we transfer the parameters in HSAN and compute the query-relevant score between segment-level features and query. The dimension of the fused vector space is set to 512. Following the setting in [6], we randomly select two videos for training, one for testing and the remaining one for testing. In the training process, we use Adam optimizer [56] to minimize the loss, with its initial learning rate 0.1 and decay rate of 0.8. When training the query-aware scoring module, the learning rate of summary annotation part is set to 0.001. The mini-batch strategy is also used and the batch size is set to 10.

C. Quantitative Results

1) Comparison Analysis: Table II show the comparison results for query-focused video summarization in terms of precision, recall and f1-score. We compare our method with other approaches that have been used in this task. It is shown that our method outperforms the state-of-the-art approach (TPAN [41]) by 1.2%. More specifically, for video 2 and video 4, we obtain a better performance than TPAN [41], by 2.87% and 1.81% respectively. The improvements in performance identify the effectiveness of our approaches to learn the relevance between the video shots and users' query. The other methods are based on DPP algorithm or using the structure of generative adversarial network, and they are trained on the two randomly selected videos in the dataset. However, we employ a method based on self attention mechanism and we leverage it to model the internal relationship between visual content from both segment-level and frame-level. Furthermore, our model is first

TABLE II

COMPARISON RESULTS ON THE QUERY-FOCUSED VIDEO SUMMARIZATION DATASET [6] IN TERMS OF PRECISION, RECALL AND F1-SCORE

	SeqDPP [8]			SH-DPP [7]			QC-DPP [6]			TPAN [42]			QSAN		
	Pre	Rec	F1	Pre	Rec	F1	Pre	Rec	F1	Pre	Rec	F1	Pre	Rec	F1
Vid1	53.43	29.81	36.59	50.56	29.64	35.67	49.86	53.38	48.68	49.66	50.91	**48.74**	48.41	52.34	48.52
Vid2	44.05	46.65	43.67	42.13	46.81	42.72	33.71	62.09	41.66	43.02	48.73	45.30	46.51	51.36	**46.64**
Vid3	49.25	17.44	25.26	51.92	29.24	36.51	55.16	62.40	56.47	58.73	56.49	56.51	56.78	61.14	**56.93**
Vid4	11.14	63.49	18.15	11.51	62.88	18.62	21.39	63.12	29.96	36.70	35.96	33.64	30.54	46.90	**34.25**
Avg.	39.47	39.35	30.92	39.03	42.14	33.38	40.03	60.25	44.19	47.03	48.02	46.05	45.56	52.94	**46.59**

TABLE III

ABLATION EXPERIMENTS ON QUERY-BIASED SELF-ATTENTION NETWORK. LSA AND GSA DENOTE LOCAL SELF-ATTENTION AND GLOBAL SELF-ATTENTION ON VIDEO RESPECTIVELY

Method	Pre	Rec	F1
QSAN(w/o LSA)	42.13	49.08	43.13
QSAN(w/o GSA)	37.10	43.21	37.98
QSAN	**45.56**	**52.94**	**46.59**

TABLE IV

ABLATION EXPERIMENTS ON QUERY-BIASED SELF-ATTENTION NETWORK. SEG DENOTES SEGMENTATION PROCESS. UNIFORM DENOTES THAT WE SEGMENT THE VIDEO EVENLY

Method	Pre	Rec	F1
QSAN(w/o Seg)	41.10	43.67	41.33
QSAN(Uniform 100)	42.71	45.36	43.85
QSAN(Uniform 200)	43.99	46.63	45.27
QSAN(KTS)	**45.56**	**52.94**	**46.59**

TABLE V

PERFORMANCE COMPARISON AMONG UNSUPERVISED METHODS ON TWO GENERIC VIDEO SUMMARIZATION DATASETS IN TERMS OF F1-SCORE

Method	SumMe	TVSum
Uniform sampling	0.293	0.155
K-medoids	0.334	0.288
Dictionary selection	0.378	0.42
SUM-GAN$_{dpp}$ [12]	0.391	0.517
SASUM$_{length}$ [40]	0.410	0.546
DR-DSN [15]	0.421	**0.581**
Ours	**0.435**	0.573

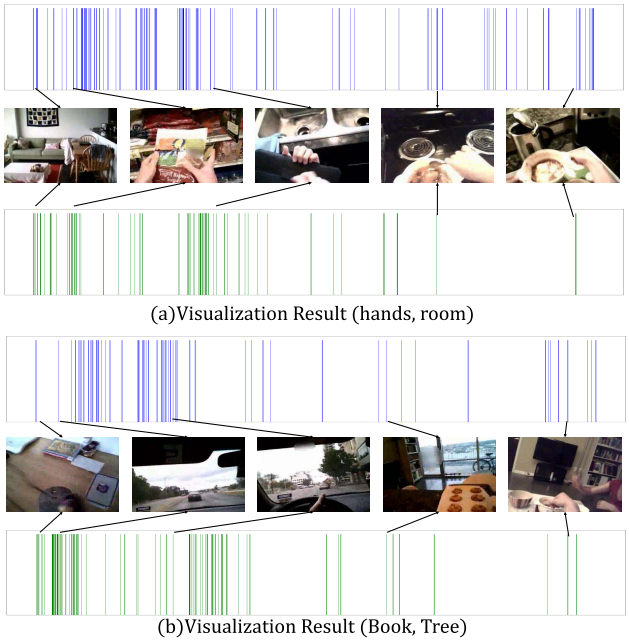


Fig. 4. The visualization results of our approach. The x-axis denotes the video shot number. Green lines represent the ground truth annotations and blue lines represent the predicted key shots from our method. (a) The results for the query "Hands Room". (b) The results for the query "Book Tree".

trained on a larger video dataset, which makes it be exposed to abundant semantic information.

2) Ablation Analysis: We conduct the ablation experiments on our proposed method QSAN including its components LSA and GSA module. The details of the experimental results are listed in Table III. LSA and GSA denote local self-attention module and global self-attention module. Without local self-attention, we use the average of the visual features among a segment as its responded segment-level visual feature. Without global self-attention, we use the output of the local self-attention module to generate description or video summary. We can find that the models without local or global self-attention perform worse than the full model on all tasks.

We also design an experiment to further investigate the effectiveness of the segmentation process in our method. We replace the KTS segmentation process with uniform segmentation and set different segment length. Compared with no segmentation process or uniform segmentation process, the full model QSAN performs better, which suggests the segmentation with semantic information is more suitable for the task.

3) Ablation Study on Transferring Parameters: To exploit the ability of QSAN for learning the visual information from other video datasets, we propose an experiment for the task of general video summarization. As is shown in table V, we propose experiments on SumMe [28] and TVSum [30] datasets. These two datasets are generic video summarization dataset, in which the query annotations are absent. SumMe [28] contains 25 user videos with an average duration of 160 seconds and internal annotations are frame-level important scores. The length of the video ranges from 1 minute to 6.5 minutes. These videos come from different categories, such as sports, cooking and daily life, which include shots taken from the first person perspective, the third person perspective and action shots, and the video content is sparse. TvSum [30] contains

50 videos from Youtube. The length of the video ranges from 2 minutes to 10 minutes. These videos cover 10 categories in the TRECVid Multimedia Event Detection (MED), with an average of 5 videos per category. Specifically, $SASUM_{length}$ method is SASUM [39] with length sparsity constraint.

Firstly, we use KTS algorithm to segment the videos and use pretrained convolutional network to extract the visual features. Then we compute the importance score for each segments using our method. We follow the steps in [4] to convert frame-level relevance scores into key frames and key shot summaries for two video summarization datasets. It can be observed that the performance of our method on the SumMe dataset has achieved the best results in all the methods. Our method on the TVSum dataset gets the second highest result, only weaker than the DS-DSN. The results show that our method can obtain sufficient semantic information from the video description and the hierarchical self-attentive network is able to learn to semantic relationship between visual content and its referred textual information and generate generic video summary.

D. Qualitative Results

We present two visualization result in Figure 4. We take two user queries, that is "Hands Room" and "Book Tree", as the inputs of our model. The x-axis in the figure represents the temporal order of video shots. The green lines represent the ground truth annotations of key shots which are related to the query. The blue lines denote the results of predicted key shots. It can be observed that the selected summaries are related to one or both concepts in the given query, which demonstrates that our proposed method is able to find diverse, representative and query-related summaries.

V. Conclusion

In this paper, we study the task of query-focused video summarization. We formulate the problem of query-focused video summarization as a task of scoring for each segment and we propose a query-biased self-attentive network to tackle this task. In the training of hierarchical self-attentive network and caption generator, the model learns the internal relationship among visual contents from frame-level and segment-level and also find out the mapping relation between visual information and textual information. Then we introduce a query-aware scoring module to compute the relevant score between video segments and user query. Finally, we can generate not only generic video summary but also query-related summary. Extensive experiments show the efficacy and efficiency of our approach.

References

[1] Z. Lu and K. Grauman, "Story-driven summarization for egocentric video," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2013, pp. 2714–2721.

[2] B. Zhao, X. Li, and X. Lu, "HSA-RNN: Hierarchical structure-adaptive RNN for video summarization," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 7405–7414.

[3] A. Khosla, R. Hamid, C.-J. Lin, and N. Sundaresan, "Large-scale video summarization using Web-image priors," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2013, pp. 2698–2705.

[4] K. Zhang, W.-L. Chao, F. Sha, and K. Grauman, "Video summarization with long short-term memory," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2016, pp. 766–782.

[5] J. Meng, H. Wang, J. Yuan, and Y.-P. Tan, "From keyframes to key objects: Video summarization by representative object proposal selection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 1039–1048.

[6] A. Sharghi, J. S. Laurel, and B. Gong, "Query-focused video summarization: Dataset, evaluation, and a memory network based approach," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 2127–2136.

[7] A. Sharghi, B. Gong, and M. Shah, "Query-focused extractive video summarization," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2016, pp. 3–19.

[8] B. Gong, W.-L. Chao, K. Grauman, and F. Sha, "Diverse sequential subset selection for supervised video summarization," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 2069–2077.

[9] Y. Jae Lee, J. Ghosh, and K. Grauman, "Discovering important people and objects for egocentric video summarization," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2012, pp. 1346–1353.

[10] R. Hong, J. Tang, H.-K. Tan, S. Yan, C. Ngo, and T.-S. Chua, "Event driven summarization for Web videos," in *Proc. 1st SIGMM Workshop Social Media WSM*, 2009, pp. 43–48.

[11] G. Kim and E. P. Xing, "Reconstructing storyline graphs for image recommendation from Web community photos," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 3882–3889.

[12] B. Mahasseni, M. Lam, and S. Todorovic, "Unsupervised video summarization with adversarial LSTM networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 202–211.

[13] C.-W. Ngo, Y.-F. Ma, and H.-J. Zhang, "Automatic video summarization by graph modeling," in *Proc. 9th IEEE Int. Conf. Comput. Vis.*, Oct. 2003, pp. 104–109.

[14] R. Panda and A. K. Roy-Chowdhury, "Collaborative summarization of topic-related videos," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 7083–7092.

[15] K. Zhou, Y. Qiao, and T. Xiang, "Deep reinforcement learning for unsupervised video summarization with diversity-representativeness reward," in *Proc. 22nd AAAI Conf. Artif. Intell.*, Apr. 2018, pp. 7582–7589.

[16] J. Wang, X. Zhu, and S. Gong, "Video semantic clustering with sparse and incomplete tags," in *Proc. 13th AAAI Conf. Artif. Intell.*, 2016, pp. 3618–3624.

[17] B. A. Plummer, M. Brown, and S. Lazebnik, "Enhancing video summarization via vision-language embedding," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 5781–5789.

[18] X. Zhu, C. C. Loy, and S. Gong, "Video synopsis by heterogeneous multi-source correlation," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2013, pp. 81–88.

[19] S. Zhang, Y. Zhu, and A. K. Roy-Chowdhury, "Context-aware surveillance video summarization," *IEEE Trans. Image Process.*, vol. 25, no. 11, pp. 5469–5478, Nov. 2016.

[20] C. T. Dang and H. Radha, "Heterogeneity image patch index and its application to consumer video summarization," *IEEE Trans. Image Process.*, vol. 23, no. 6, pp. 2704–2718, Jun. 2014.

[21] B. Zhao and E. P. Xing, "Quasi real-time summarization for consumer videos," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 2513–2520.

[22] M. Otani, Y. Nakashima, E. Rahtu, J. Heikkilä, and N. Yokoya, "Video summarization using deep semantic features," in *Proc. Asian Conf. Comput. Vis.* Cham, Switzerland: Springer, 2016, pp. 361–377.

[23] W.-S. Chu, Y. Song, and A. Jaimes, "Video co-summarization: Video summarization by visual co-occurrence," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 3584–3592.

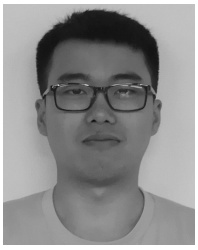
[24] G. Kim, L. Sigal, and E. P. Xing, "Joint summarization of large-scale collections of Web images and videos for storyline reconstruction," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 4225–4232.

[25] R. Panda, A. Das, Z. Wu, J. Ernst, and A. K. Roy-Chowdhury, "Weakly supervised summarization of Web videos," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 3657–3666.

[26] B. Xiong and K. Grauman, "Detecting snap points in egocentric video with a Web photo prior," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2014, pp. 282–298.

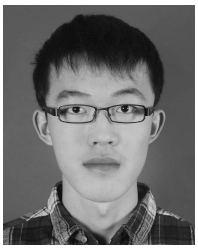
[27] A. Kanehira, L. Van Gool, Y. Ushiku, and T. Harada, "Viewpoint-aware video summarization," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 18–22.

[28] M. Gygli, H. Grabner, H. Riemenschneider, and L. Van Gool, "Creating summaries from user videos," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2014, pp. 505–520.
[29] D. Potapov, M. Douze, Z. Harchaoui, and C. Schmid, "Category-specific video summarization," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2014, pp. 540–555.
[30] Y. Song, J. Vallmitjana, A. Stent, and A. Jaimes, "TVSum: Summarizing Web videos using titles," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 5179–5187.
[31] M. Gygli, H. Grabner, and L. Van Gool, "Video summarization by learning submodular mixtures of objectives," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 3090–3098.
[32] K. Zhang, W.-L. Chao, F. Sha, and K. Grauman, "Summary transfer: Exemplar-based subset selection for video summarization," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 1059–1067.
[33] T. Yao, T. Mei, and Y. Rui, "Highlight detection with pairwise deep ranking for first-person video summarization," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 982–990.
[34] M. Rochan, L. Ye, and Y. Wang, "Video summarization using fully convolutional sequence networks," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 347–363.
[35] A. Sharghi, A. Borji, C. Li, T. Yang, and B. Gong, "Improving sequential determinantal point processes for supervised video summarization," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Sep. 2018, pp. 517–533.
[36] K. Zhang, K. Grauman, and F. Sha, "Retrospective encoders for video summarization," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Sep. 2018, pp. 383–399.
[37] S. Cai, W. Zuo, L. S. Davis, and L. Zhang, "Weakly-supervised video summarization using variational encoder-decoder and Web prior," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Sep. 2018, pp. 184–200.
[38] S. Yeung, A. Fathi, and L. Fei-Fei, "VideoSET: Video summary evaluation through text," 2014, *arXiv:1406.5824*. [Online]. Available: http://arxiv.org/abs/1406.5824
[39] H. Wei, B. Ni, Y. Yan, H. Yu, X. Yang, and C. Yao, "Video summarization via semantic attended networks," in *Proc. 22nd AAAI Conf. Artif. Intell.*, 2018, pp. 216–223.
[40] A. B. Vasudevan, M. Gygli, A. Volokitin, and L. Van Gool, "query-adaptive video summarization via quality-aware relevance estimation," in *Proc. ACM Multimedia Conf. MM*, 2017, pp. 582–590.
[41] Y. Zhang, M. Kampffmeyer, X. Liang, M. Tan, and E. P. Xing, "Query-conditioned three-player adversarial network for video summarization," 2018, *arXiv:1807.06677*. [Online]. Available: http://arxiv.org/abs/1807.06677
[42] S. Venugopalan, M. Rohrbach, J. Donahue, R. Mooney, T. Darrell, and K. Saenko, "Sequence to sequence—Video to text," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 4534–4542.
[43] K. Xu *et al.*, "Show, attend and tell: Neural image caption generation with visual attention," in *Proc. Int. Conf. Mach. Learn.*, Jun. 2015, pp. 2048–2057.
[44] H. Yu, J. Wang, Z. Huang, Y. Yang, and W. Xu, "Video paragraph captioning using hierarchical recurrent neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 4584–4593.
[45] M. Ranzato, S. Chopra, M. Auli, and W. Zaremba, "Sequence level training with recurrent neural networks," 2015, *arXiv:1511.06732*. [Online]. Available: http://arxiv.org/abs/1511.06732
[46] S. J. Rennie, E. Marcheret, Y. Mroueh, J. Ross, and V. Goel, "Self-critical sequence training for image captioning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 7008–7024.
[47] X. Wang, W. Chen, J. Wu, Y.-F. Wang, and W. Y. Wang, "Video captioning via hierarchical reinforcement learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 4213–4222.
[48] Y. Chen, S. Wang, W. Zhang, and Q. Huang, "Less is more: Picking informative frames for video captioning," in *Proc. Eur. Conf. Comput. Vis.*, Sep. 2018, pp. 358–373.
[49] T. Shen, T. Zhou, G. Long, J. Jiang, S. Pan, and C. Zhang, "Disan: Directional self-attention network for rnn/cnn-free language understanding," in *Proc. 22nd AAAI Conf. Artif. Intell.*, 2018, pp. 5446–5455.
[50] R. J. Williams, "Simple statistical gradient-following algorithms for connectionist reinforcement learning," *Mach. Learn.*, vol. 8, nos. 3–4, pp. 229–256, 1992.
[51] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "BLEU: A method for automatic evaluation of machine translation," in *Proc. 40th Annu. Meeting Assoc. Comput. Linguistics ACL*, 2001, pp. 311–318.
[52] F. C. Heilbron, V. Escorcia, B. Ghanem, and J. C. Niebles, "ActivityNet: A large-scale video benchmark for human activity understanding," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 961–970.
[53] R. Krishna, K. Hata, F. Ren, L. Fei-Fei, and J. C. Niebles, "Dense-captioning events in videos," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 706–715.
[54] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. A. Alemi, "Inception-v4, inception-resnet and the impact of residual connections on learning," in *Proc. 21st AAAI Conf. Artif. Intell.*, 2017, pp. 4278–4284.
[55] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 248–255.
[56] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*. [Online]. Available: http://arxiv.org/abs/1412.6980

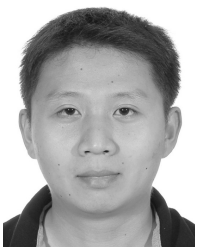


Shuwen Xiao received the B.E. degree in computer science and technology from Tongji University, China, in 2017. He is currently pursuing the M.S. degree in computer science with Zhejiang University. His research interests include machine learning and computer vision.

Zhou Zhao received the B.S. and Ph.D. degrees in computer science from The Hong Kong University of Science and Technology in 2010 and 2015, respectively. He is currently an Associate Professor with the College of Computer Science, Zhejiang University. His research interests include machine learning and data mining.



Zijian Zhang received the bachelor's degree in computer science and technology from Zhejiang University, China, in 2019, where he is currently pursuing the Ph.D. degree in computer science. His research interests include machine learning, data mining, computer vision, and natural language processing.



Ziyu Guan received the B.S. and Ph.D. degrees in computer science from Zhejiang University, China, in 2004 and 2010, respectively. He had worked as a Research Scientist at the University of California at Santa Barbara from 2010 to 2012. He is currently a Full Professor with the School of Information and Technology, Northwest University, China. His research interests include attributed graph mining and search, machine learning, expertise modeling and retrieval, and recommender systems.

Deng Cai received the Ph.D. degree in computer science from the University of Illinois at Urbana–Champaign in 2009. He is currently a Professor with the State Key Laboratory of CAD&CG, College of Computer Science, Zhejiang University, China. His research interests include machine learning, data mining, and information retrieval.