Instructional Video Summarization using Attentive Knowledge Grounding

Kyungho Kim*, Kyungjae Lee*, and Seung-won Hwang **

Yonsei University, Seoul, South Korea
{ggdg12,lkj0509,seungwonh}@yonsei.ac.kr

Abstract. This demonstration considers the scenario of summarizing an instructional video, for query such as "how to cook galbi", to efficiently obtain a skillset. Specifically, we use the query to retrieve both the relevant video and the external procedural knowledge, such as a recipe document, and show summarization is more effective with such augmentation.

Keywords: Multimodality · Video Summarization · Attentive Knowledge Grounding

1 Introduction

Instructional videos have become an effective means to learn new skills including cooking, gardening, and sports. Though more people rely on videos for searching instructional information, video search scenarios focus on returning the entire video matching user queries [3, 5].

In this paper, we study a summarization scenario of effectively learning skill from video summaries, by providing the extracted key segments from the full video. Existing video summarization [6, 7] extracts highlight clips, identifying whether each video frame is the keyframe (1) or not (0). For this purpose, most recently, multi-modality of both video clip and transcript has been used [10], for such classification.

Our key claim is that deciding keyframe should not be localized to the given clip (and its transcript), but globally matched with respect to the entire procedure. To validate, we propose to augment external procedural knowledge (e.g., cooking recipe) and leverage its relevance to the given clip. Figure 1 illustrates our scenario of summarizing an instructional video for user query Q, such as '*how to make Galbi*'. We use Q, not only to retrieve the video, but also to retrieve its recipe document, to decide a keyframe, using transcript, video clip, and recipe.

Inspired by conversational Question Answering (ConvQA) [11], retrieving a relevant document as knowledge to enrich the representation of utterances, we propose to use the recipe as a procedural knowledge to enrich that of video utterance, which is video clip and its transcript pair in our context. Specifically,

* First two authors equally contributed to this work.
** corresponding author

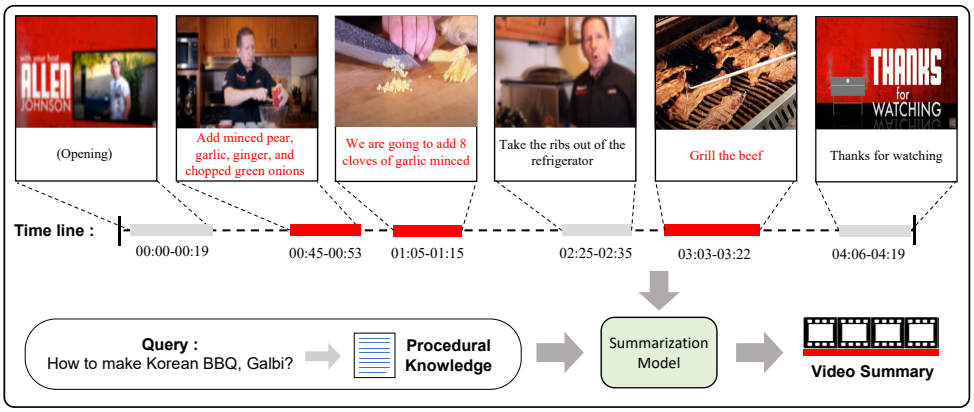

Fig. 1. An overview of our approach. Returned segments are marked in red, for users to either skip to specific instruction or watch all segments as a summary without inessential steps, such as opening and greeting.

clip is semantically matched to relevant steps in the recipe, without any relevance supervision unlike in [11], for which we propose attentive knowledge matching inspired by video QA model [1, 4]. Details will be discussed more in Section 3 & 4. We release our demo at http://dilab.yonsei.ac.kr/IVSKS.php, built on instructional cooking video set.

2 Problem Formulation

We abstract our task as predicting multiple video segments given a query. More formally, let a given video be V containing image frames I and transcript T, where l image frames ($I = \{I_1, I_2, ..., I_l\}$) are sampled from V (1 frame per 1.5 sec) and each I_k corresponds to a sentence T_k in the transcript. That is, a video V can be defined as a sequence of tuples, $V = \{(I_1, T_1), (I_2, T_2), ..., (I_l, T_l)\}$. Let a query be $\boldsymbol{Q} = \{w_1, w_2, ..., w_n\}$, where w means n words of the query. We expand the given query Q into textual descriptions answering the query using external procedural knowledge (*e.g.*, an encyclopedia of how-to queries). The knowledge often consists of multiple sentences, m sentences, corresponding to multiple steps required for Q, denoted as $\boldsymbol{K} = \{S_1, S_2, ..., S_m\}$, where S indicates each sentence in the descriptions. Our empirical finding (Section 4) suggests that K is most effective in the goal of generating an instructional summary.

With this formulation, our goal is to predict whether tuple (I_k, T_k) should be included in answer segment A or not, with respect to the given query Q_c and its finer counterpart K. For evaluation, the predicted answer segment $\boldsymbol{A}$ is compared with the ground-truth answer segment $\boldsymbol{A}^*$ with F1 score in Section 4.

3 Proposed Model

Our model consists of (a) **Video Encoding**, turning a sequence of (I, T) into multimodal representation, (b) **Knowledge-Video (KV) fusion**, aligning knowledge K with video segments. We are the first to deal with the expansion of query, with procedural knowledge grounding, and predict multiple segments as an answer.

Video Encoding: For multimodal encoding of a video, we use a pre-trained model, LXMERT [9], receiving an image and paired sentence as input. Since there is no knowledge paired with videos, thus we delay the knowledge representation to the next step of video encoding. The multimodal encoding using LXMERT with transcript and image is done as follows:

$$\boldsymbol{v_k} = LXMERT(I_k, T_k), \quad (k \in [1, l]) \tag{1}$$

where $\boldsymbol{v_k}$ indicates multimodal representation in k-th image frame and transcript. We compute the multimodal embedding for each frame at the point of LXMERT operating in image units. After encoding each frame (I_k, T_k) to $\boldsymbol{v_k}$, we contextualize the video sequence using (1) self-attention for global context, and (2) bi-directional LSTM for considering temporal information.

KV Fusion: The first step of KV fusion is to expand Q into $K = \{S_1, ..., S_m\}$, augmented by external knowledge. We then encode sentence S_k using CLS representation from BERT model [2], as $\boldsymbol{s_k} = BERT(S_k)$. However, this BERT encoding is unaware of corresponding video frame representation. The goal of KV fusion is (a) to align S_i to corresponding video representation (if exists), such that (b) importance of S_i can be estimated by whether it can be aligned with video frame, and vice versa. Specifically, we can estimate importance of S_i from frame, modeled as **Frame-to-Knowledge (F2K)** attention, or that of frame from knowledge **Knowledge-to-Frame (K2F)** attention, inspired by attention flow layer of BiDAF [8]. Formally, we describe **F2K** attention as below:

$$\alpha_i^k = \underset{i}{\text{softmax}}(\boldsymbol{v_k} W_1 \boldsymbol{s_i}), \qquad \hat{s_k} = \sum_i \alpha_i^k \cdot \boldsymbol{s_i} \tag{2}$$

where W_1 indicates a trainable matrix and α_i^k is the attention weights. That is, given k-th image, $\hat{s_k}$ represents the weighted sum of descriptions s_i, which can attend specific description corresponding to the given image.

Likewise, **K2F** attention signifies which frame has relevant and critical information, with respect to alignment with frames, as we formally describe below:

$$\beta^k = \underset{k}{\text{softmax}}(\boldsymbol{v_k} W_2 \boldsymbol{s}), \qquad \hat{v_k} = \beta^k \cdot \hat{\boldsymbol{s_k}} \tag{3}$$

where W_2 indicates a trainable matrix, β^k is the attention weights and s means the embedding of procedural knowledge. In our experiment, we decide to use s

as output of the convolution network of attentive descriptions. We use concatenation of $\hat{v_k}$ and v_k for final fusion features.

During training, we use cross-entropy loss with labels L_k where the ground-truth in from beginning to end position is equal to 1, otherwise 0. Finally, our loss function is described as below:

$$Loss = -\sum_{k}\{L_k \cdot log(FC([v_k|\hat{v_k}])) + (1 - L_k) \cdot (1 - log(FC([v_k|\hat{v_k}])))\} \quad (4)$$

where FC is fully-connected network with a sigmoid function, and | indicates concatenation. At inference time, we extract frames over threshold, and aggregate them as the final answer summary.

4 Data Collection and Demonstration

Dataset: We build demo upon YouCook2 dataset[1], containing 2000 query-video pairs. As annotators segment videos covering important steps, we can compute F1 score with respect to the chosen segments as ground-truth summary answer $\boldsymbol{A^*}$. For overall scores, we average each score over all of the instances. Our procedural knowledge reference documents were crawled from a collection of recipes[2], and we retrieve top 1 recipe by searching queries then use them as knolwedge K.

Our demo URL shows example queries and their results, and Table 1 reports our quantitative evaluation: **Random** is a naive baseline selecting a random start/end frame for each video. **Summ** is a summarization baseline without considering knowledge K, by removing $\hat{v_k}$ in Eq (4). The rest is **Ours**. Ours with query expansion, augmented from external knowledge, and KV fusion using **F2K** and **K2F** improves the F1 score ($\sim$ 70%), leading to 2% gain, compared to **Summ**.

	Random	Summ (no K)	Ours (KV fusion)
F1 Score	40.0%	67.6%	**69.6%**

Table 1. Empirical validation

Acknowledgements

Microsoft Research Asia and Artificial Intelligence Graduate School Program (2020-0-01361).

[1] http://youcook2.eecs.umich.edu/
[2] https://www.allrecipes.com/

References

1. Colas, A., Kim, S., Dernoncourt, F., Gupte, S., Wang, D.Z., Kim, D.S.: Tutorialvqa: Question answering dataset for tutorial videos. arXiv:1912.01046 (2019)
2. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv:1810.04805 (2018)
3. Dong, J., Li, X., Xu, C., Ji, S., He, Y., Yang, G., Wang, X.: Dual encoding for zero-example video retrieval. In: CVPR (2019)
4. Lee, K., Duan, N., Ji, L., Li, J., Hwang, S.w.: Segment-then-rank: Non-factoid question answering on instructional videos. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 34, pp. 8147–8154 (2020)
5. Liu, Y., Albanie, S., Nagrani, A., Zisserman, A.: Use what you have: Video retrieval using representations from collaborative experts. arXiv:1907.13487 (2019)
6. Otani, M., Nakashima, Y., Rahtu, E., Heikkila, J.: Rethinking the evaluation of video summaries. In: CVPR (2019)
7. Rochan, M., Ye, L., Wang, Y.: Video summarization using fully convolutional sequence networks. In: ECCV. pp. 347–363 (2018)
8. Seo, M., Kembhavi, A., Farhadi, A., Hajishirzi, H.: Bidirectional attention flow for machine comprehension. ArXiv **abs/1611.01603** (2017)
9. Tan, H., Bansal, M.: Lxmert: Learning cross-modality encoder representations from transformers. arXiv:1908.07490 (2019)
10. Xu, F.F., Ji, L., Shi, B., Du, J., Neubig, G., Bisk, Y., Duan, N.: A benchmark for structured procedural knowledge extraction from cooking videos. arXiv preprint arXiv:2005.00706 (2020)
11. Yang, L., Qiu, M., Qu, C., Guo, J., Zhang, Y., Croft, W.B., Huang, J., Chen, H.: Response ranking with deep matching networks and external knowledge in information-seeking conversation systems. In: SIGIR (2018)