

Paper review

CLIP-It!: Language-Guided Video Summarization **(NeurIPS 2021)**

Presentation: **Jeiyoon Park**
6th Generation, TAVE

Outline

1. Contribution
2. Method
3. Experiments
4. Conclusion

Outline

1. Contribution
2. Method
3. Experiments
4. Conclusion

Detour: Video Summarization

1. Video Summarization



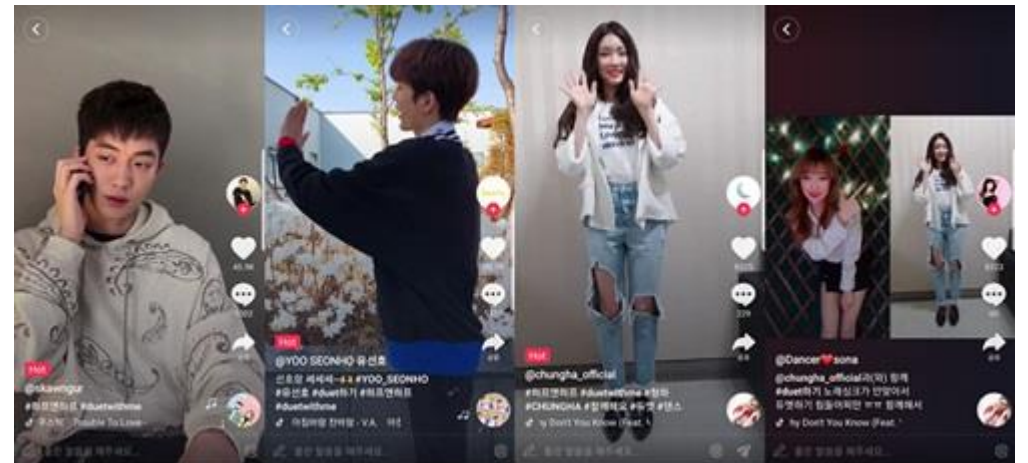
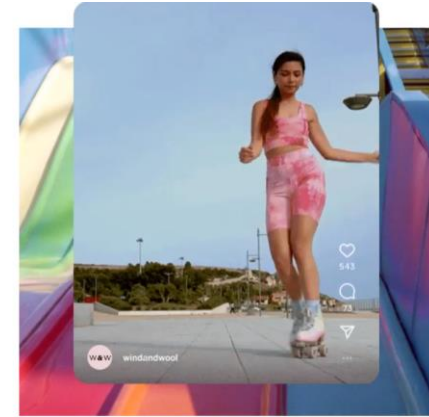
Detour: Video Summarization

2. Increased Preference for Short-Form Videos



릴스

Instagram의 모든 곳에서 쉽게 발견할 수 있는 짧고 재미있는 동영상으로 사람들의 관심을 사로잡으세요.



Contribution: Why This paper

1. Language-Guided Video Summarization

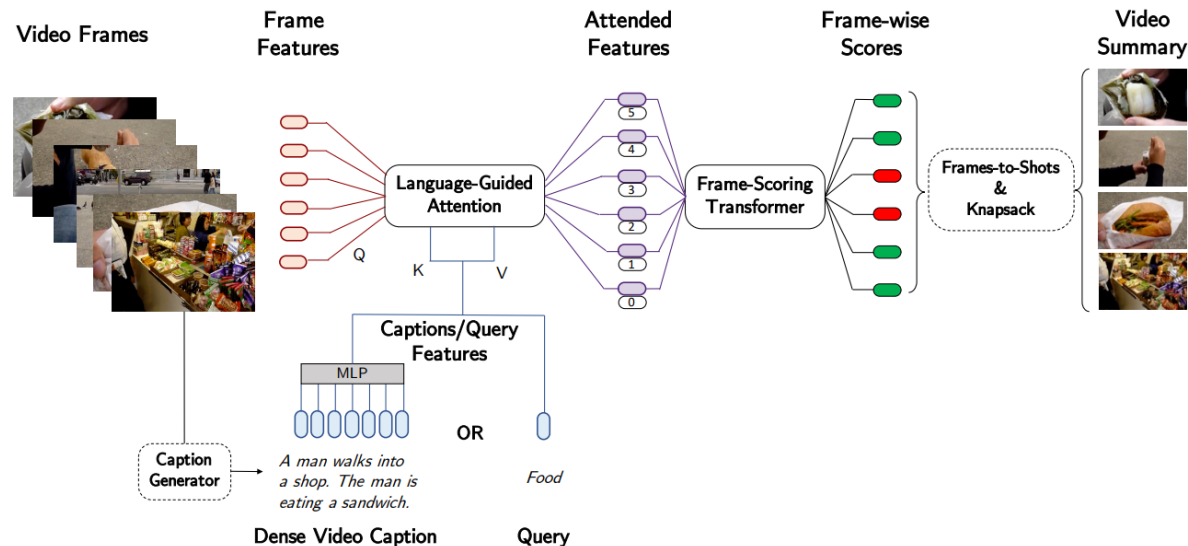
- The importance of scenes in a video is subjective
- Users should have the option of customizing the summary to specify what is important to them



Contribution: Why This paper

2. CLIP-It

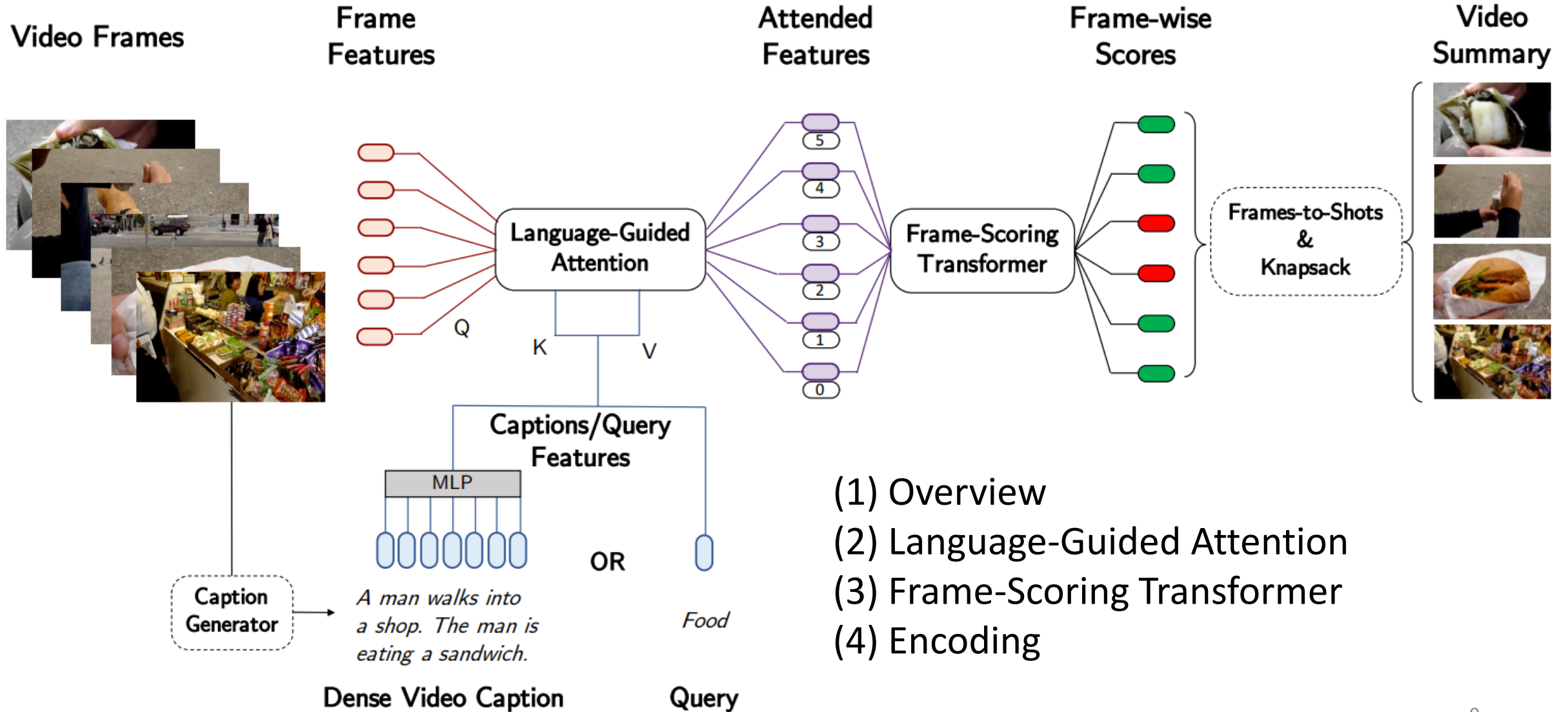
- A single framework for Addressing both (1) generic, and (2) query-focused video summarization
- With an automatically generated dense video caption: (1) generic video summarization
- With a user-defined query: (2) query-focused summarization
- Using a language-guided multimodal transformer



Outline

1. Contribution
- 2. Method**
3. Experiments
4. Conclusion

Language-Guided Video Summarization



Language-Guided Video Summarization

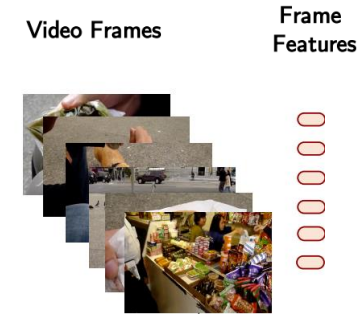
- (1) Overview
- (2) Language-Guided Attention
- (3) Frame-Scoring Transformer
- (4) Encoding

1. Overview

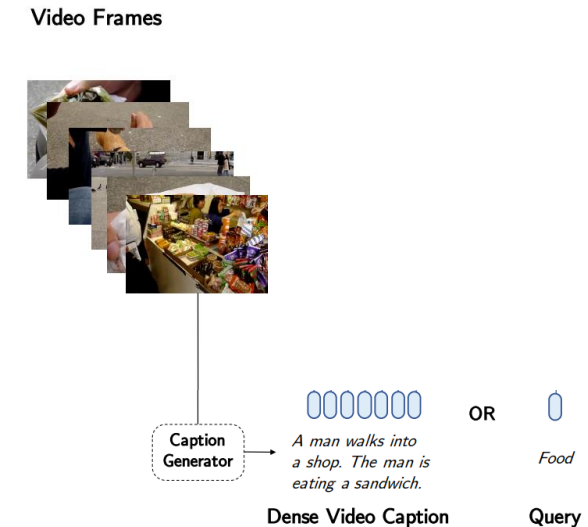
(1) F_i is frames, $i \in [1, \dots, N]$



(2) A pretrained network f_{img} embeds the frames



(3) A pretrained network f_{txt} embeds the query or dense video caption $C_j, j \in [1, \dots, M]$, where M is a sentence



Language-Guided Video Summarization

(1) Overview

(2) Language-Guided Attention

(3) Frame-Scoring Transformer

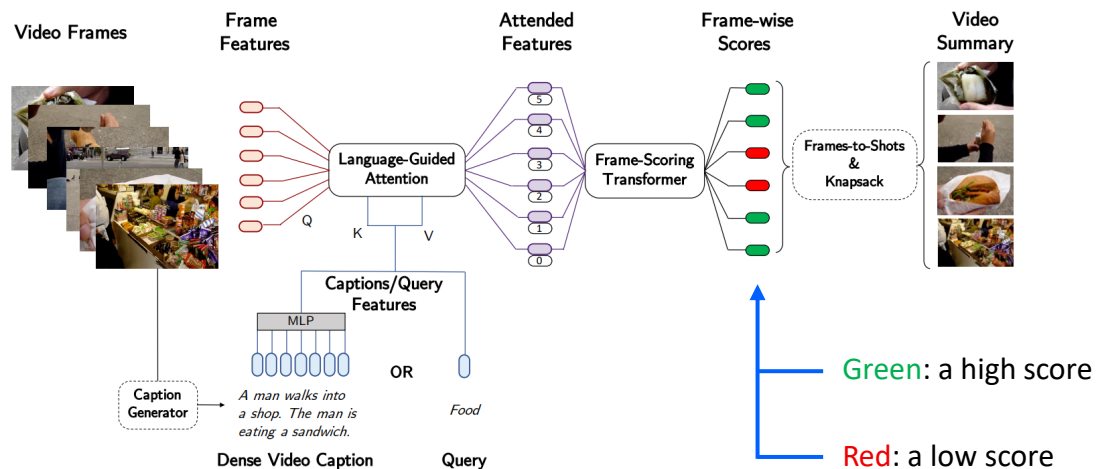
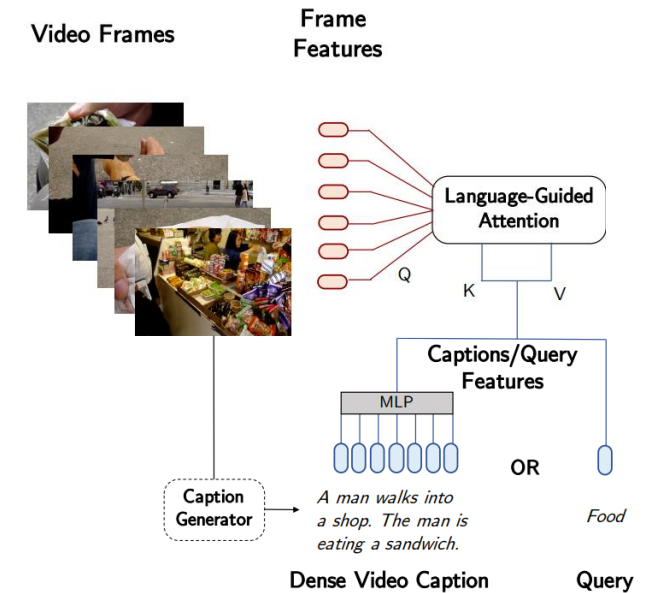
(4) Encoding

1. Overview

(4) We compute language attended image embeddings using learned Language-Guided Multi-head Attention:

$$f_{img_txt}^*$$

(5) Finally, we train a **Frame-Scoring Transformer** which assigns scores to each frame in the video



Language-Guided Video Summarization

- (1) Overview
- (2) Language-Guided Attention
- (3) Frame-Scoring Transformer
- (4) Encoding

2. Language-Guided Attention

- Using a single attention head does not suffice as the goal is to allow all captions to attend to all frames in the video
- We set Query Q , Key K , and Value V as follows:

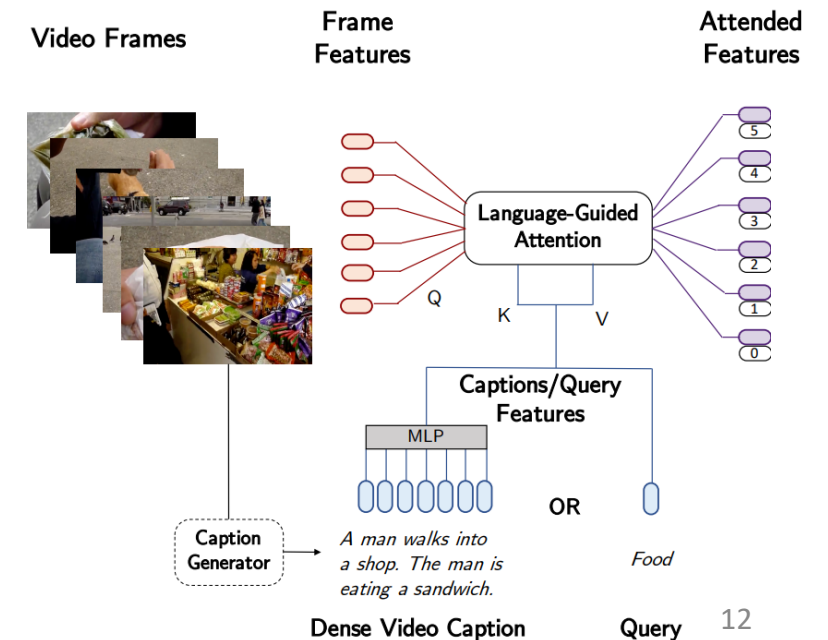
$$Q = f_{img}(F_i), \text{ where } i \in [1, \dots, N],$$

$$K, V = f_{txt}(C_j), \text{ where } j \in [1, \dots, M],$$

$$\text{Language - Guided Attn.}(Q, K, V) = \text{Concat}(\text{head}_1, \dots, \text{head}_h)W^O,$$

$$\text{where head}_i = \text{Attention}(QW_i^Q, KW_i^K, VW_i^V)$$

$$\text{and Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$

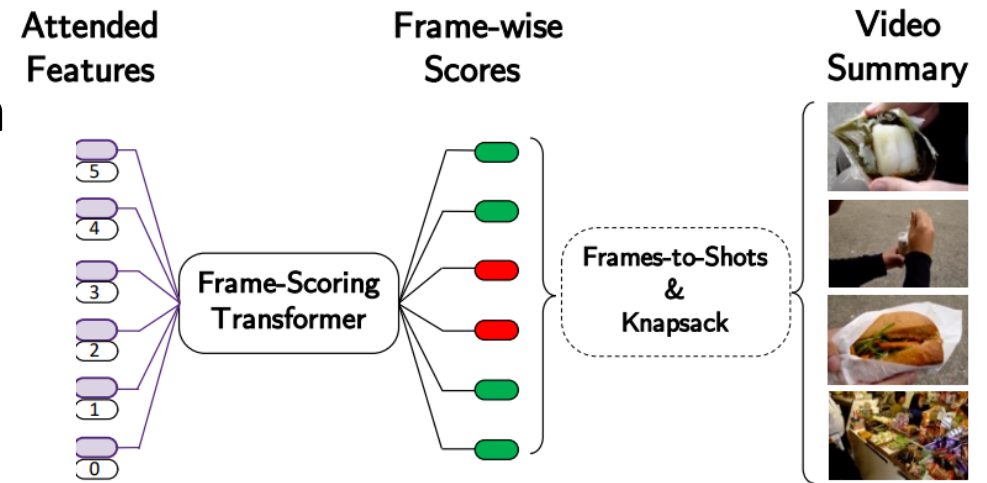


Language-Guided Video Summarization

- (1) Overview
- (2) Language-Guided Attention
- (3) Frame-Scoring Transformer
- (4) Encoding

3. Frame-Scoring Transformer

- It doesn't include redundant information, e.g., several key shots from the same event
- **Frame-Scoring Transformer** takes image-text representation as input and outputs **one score per frame**
- It uses positional encoding to insert information about the relative positions of the tokens in the sequence



Language-Guided Video Summarization

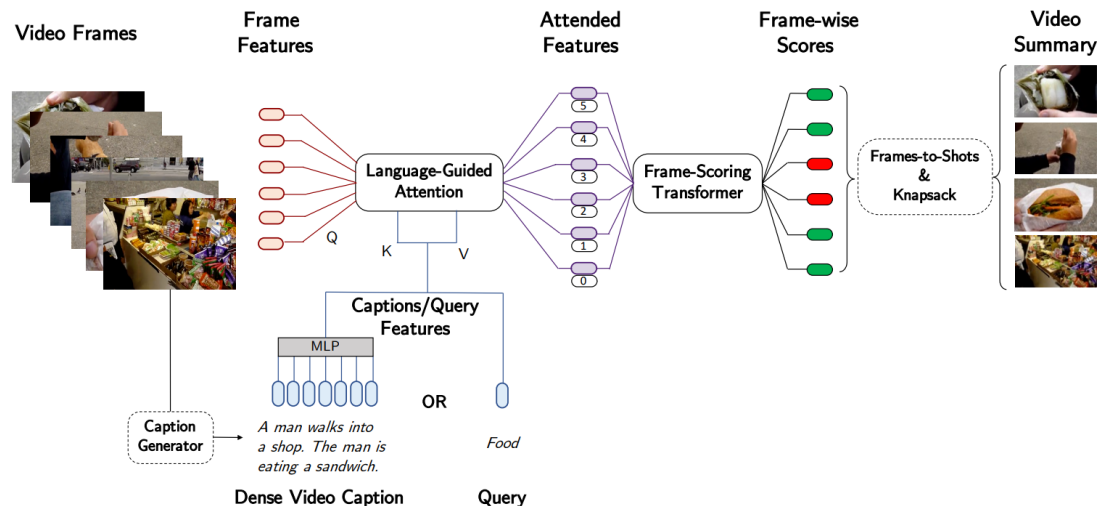
- (1) Overview
- (2) Language-Guided Attention
- (3) Frame-Scoring Transformer
- (4) Encoding

4. Encoding

(1) Image Encoding (f_{img}): *GoogleNet, ResNet, and CLIP model*

(2) Text Encoding (f_{txt}): CLIP (ViT and RN101) model

- First, embeds each sentence of the caption using the text encoder f_{txt}
- And then, concatenates and fuses using a multi-layer perceptron (MLP)



Learning

1. Supervised setting

- Classification loss
- Reconstruction loss
- Diversity loss

2. Unsupervised setting

- Reconstruction loss
- Diversity loss

Learning

1. Classification Loss

- Weighted binary cross entropy loss (\mathcal{L}_c) for classifying each frame:

$$\mathcal{L}_c = -\frac{1}{N} \sum_{i=1}^N w^* [x_i^* \log(x_i)] + (1 - w^*) [(1 - x_i^*) \log(1 - x_i)],$$

where x_i^* is the ground-truth label of the i -th frame

N is the total number of frames in the video

w^* is the weight assigned to the class x_i^* , which is set to $\frac{\#keyframes}{N}$ if x_i^* is a keyframe and

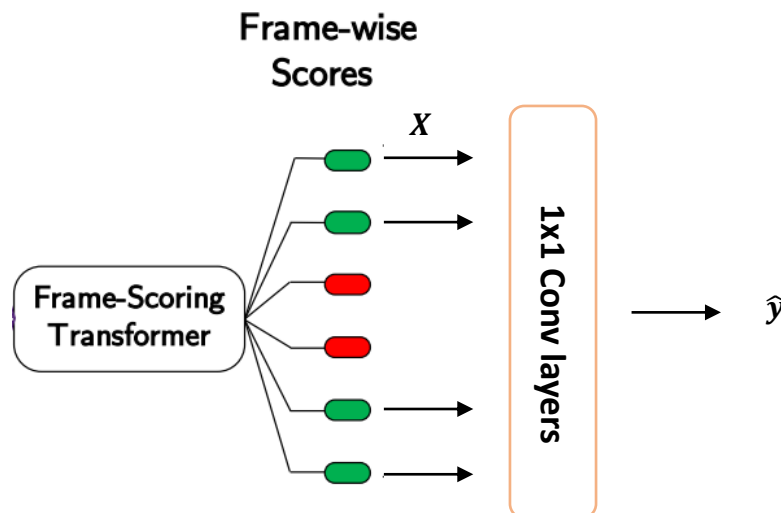
$1 - \frac{\#keyframes}{N}$ if x_i^* is a background frame

Learning

2. Reconstruction Loss

- \mathcal{L}_r is defined as the **mean squared error** between the reconstructed features and the original features corresponding to the selected keyframes, such that:

$$\mathcal{L}_r = \frac{1}{X} \sum_{i \in X} \|\mathbf{x}_i - \hat{\mathbf{y}}_i\|_2, \text{ where } \hat{\mathbf{y}} \text{ denotes the reconstructed features.}$$



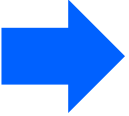
Learning

3. Diversity Loss

- To enforce diversity among selected keyframes.

$$\mathcal{L}_d = \frac{1}{X(X-1)} \sum_{i \in X} \sum_{j \in X, j \neq i} \frac{\hat{\mathbf{y}}_i \cdot \hat{\mathbf{y}}_j}{\|\hat{\mathbf{y}}_i\|_2 \cdot \|\hat{\mathbf{y}}_j\|_2},$$

where $\hat{\mathbf{y}}_i$ and $\hat{\mathbf{y}}_j$ denote the reconstructed feature vectors of the i -th and j -th node.


$$\begin{aligned} \mathcal{L}_{sup} &= \alpha \cdot \mathcal{L}_c + \beta \cdot \mathcal{L}_d + \lambda \cdot \mathcal{L}_r, \\ \mathcal{L}_{unsup} &= \beta \cdot \mathcal{L}_d + \lambda \cdot \mathcal{L}_r \end{aligned}$$

Outline

1. Contribution
2. Method
- 3. Experiments**
4. Conclusion

Experiments: Generic Video Summarization

1. Datasets

- [TVSum](#):
 - 50 videos,
 - 10 categories with 5 videos from each category (news documentary etc.),
 - 1-5 min in length
- [SumMe](#):
 - 25 videos,
 - capturing multiple events such as cooking and sports,
 - 1-6 min in length
- [Youtube dataset](#): 39 videos
- [Open Video Project \(OVP\) dataset](#): 50 videos

Experiments: Generic Video Summarization

2. Results

Table 1: **Supervised.** Comparing F1 Scores of our methods with supervised baselines on the SumMe [7] and TVSum [36] datasets using Standard, Augment, and Transfer data configurations.

Method	SumMe			TVSum		
	Standard	Augment	Transfer	Standard	Augment	Transfer
Zhang <i>et al.</i> (SumTransfer) [42]	40.9	41.3	38.5	-	-	-
Zhang <i>et al.</i> (LSTM) [43]	38.6	42.9	41.8	54.7	59.6	58.7
Mahasseni <i>et al.</i> (SUM-GAN _{sup}) [22]	41.7	43.6	-	56.3	61.2	-
Rochan <i>et al.</i> (SUM-FCN) [31]	47.5	51.1	44.1	56.8	59.2	58.2
Rochan <i>et al.</i> (SUM-DeepLab) [31]	48.8	50.2	45.0	58.4	59.1	57.4
Zhou <i>et al.</i> [47]	42.1	43.9	42.6	58.1	59.8	58.9
Zhang <i>et al.</i> [44]	-	44.9	-	-	63.9	-
Fajtl <i>et al.</i> [3]	49.7	51.1	-	61.4	62.4	-
Rochan <i>et al.</i> [30]	-	48.0	41.6	-	56.1	55.7
Chen <i>et al.</i> (V2TS) [1]	-	-	-	62.1	-	-
He <i>et al.</i> [9]	47.2	-	-	59.4	-	-
Park <i>et al.</i> (SumGraph) [26]	51.4	52.9	48.7	63.9	65.8	60.5
GoogleNet+bi-LSTM	38.5	42.4	40.7	53.9	59.6	58.6
ResNet+bi-LSTM	39.4	44.0	42.6	55.0	61.0	59.9
CLIP-Image+bi-LSTM	41.1	45.9	44.9	56.8	63.7	61.6
CLIP-Image+Video Caption+bi-LSTM	41.2	46.1	45.5	57.1	64.3	62.4
GoogleNet+Transformer	51.6	53.5	49.4	64.2	66.3	61.3
ResNet+Transformer	52.8	54.9	50.3	65.0	67.5	62.8
CLIP-Image+Transformer	53.5	55.3	51.0	65.5	68.1	63.4
CLIP-It: CLIP-Image+Video Caption+Transformer	54.2	56.4	51.9	66.3	69.0	65.5

- **Standard:** training and test splits are from the same dataset

- **Augment:** Training set from one dataset is combined with all the data from the remaining three datasets

- **Transfer:** It involves training a model on three datasets and evaluating on the fourth unseen dataset

Experiments: Generic Video Summarization

2. Results

Table 2: **Unsupervised.** Comparing F1 Scores of our methods with unsupervised baselines on the SumMe [7] and TVSum [36] datasets using Standard, Augment, and Transfer data configurations.

Method	SumMe			TVSum		
	Standard	Augment	Transfer	Standard	Augment	Transfer
Mahasseni <i>et al.</i> [22]	39.1	43.4	-	51.7	59.5	-
Yuan <i>et al.</i> [41]	41.9	-	-	57.6	-	-
Rochan <i>et al.</i> (SUM-FCN _{unsup}) [31]	41.5	-	39.5	52.7	-	-
Rochan <i>et al.</i> [30]	47.5	-	41.6	55.6	-	55.7
He <i>et al.</i> [9]	46.0	47.0	44.5	58.5	58.9	57.8
Park <i>et al.</i> (SumGraph) [26]	49.8	52.1	47.0	59.3	61.2	57.6
GoogleNet+bi-LSTM	33.1	38.0	36.5	47.7	54.9	52.3
ResNet+bi-LSTM	34.5	40.1	39.6	51.0	56.2	53.8
CLIP-Image+bi-LSTM	35.7	41.0	41.4	52.8	58.7	56.0
CLIP-Image+Video Caption+bi-LSTM	36.9	42.4	42.5	53.5	59.4	57.6
GoogleNet+Transformer	50.0	52.7	47.6	59.9	62.1	58.4
ResNet+Transformer	50.8	53.9	49.3	61.1	63.0	59.9
CLIP-Image+Transformer	51.2	53.6	49.2	61.9	64.0	60.6
CLIP-It: CLIP-Image+Video Caption+Transformer	52.5	54.7	50.0	63.0	65.7	62.8

- **Standard:** training and test splits are from the same dataset

- **Augment:** Training set from one dataset is combined with all the data from the remaining three datasets

- **Transfer:** It involves training a model on three datasets and evaluating on the fourth unseen dataset

Experiments: Generic Video Summarization

2. Results



Figure 3: Comparison of ground-truth summary to results from CLIP-Image+Transformer and the full CLIP-It model (CLIP-Image+Video Caption+Transformer). The input is a recipe video. Without captions, the model assigns high scores to certain irrelevant frames such as scenes of the woman talking or eating which hurts the precision. With captions, the cross-attention mechanism ensures that frames with important actions and objects are assigned high scores.

Experiments: Generic Video Summarization

2. Results



Figure 4: Qualitative result comparing the generic summary from CLIP-It with the ground-truth summary. The plots showing predicted and ground-truth frame-level scores are similar, indicating that frames that were given a high score in ground-truth were also assigned high scores by our model.

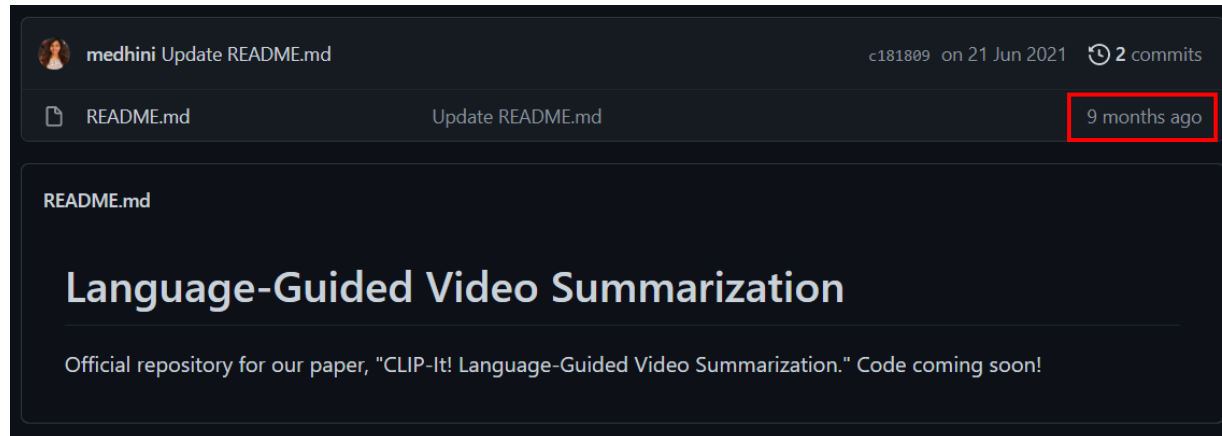
Outline

1. Contribution
2. Method
3. Experiments
- 4. Conclusion**

Conclusion

1. Any drawbacks?

(1) No code is available



(2) Language-guided summarization?



Thank you

<https://jeiyoong.github.io/>