Deep Reinforcement Learning for Unsupervised Video Summarization with Diversity-Representativeness Reward

Kaiyang Zhou, Yu Qiao, Tao Xiang

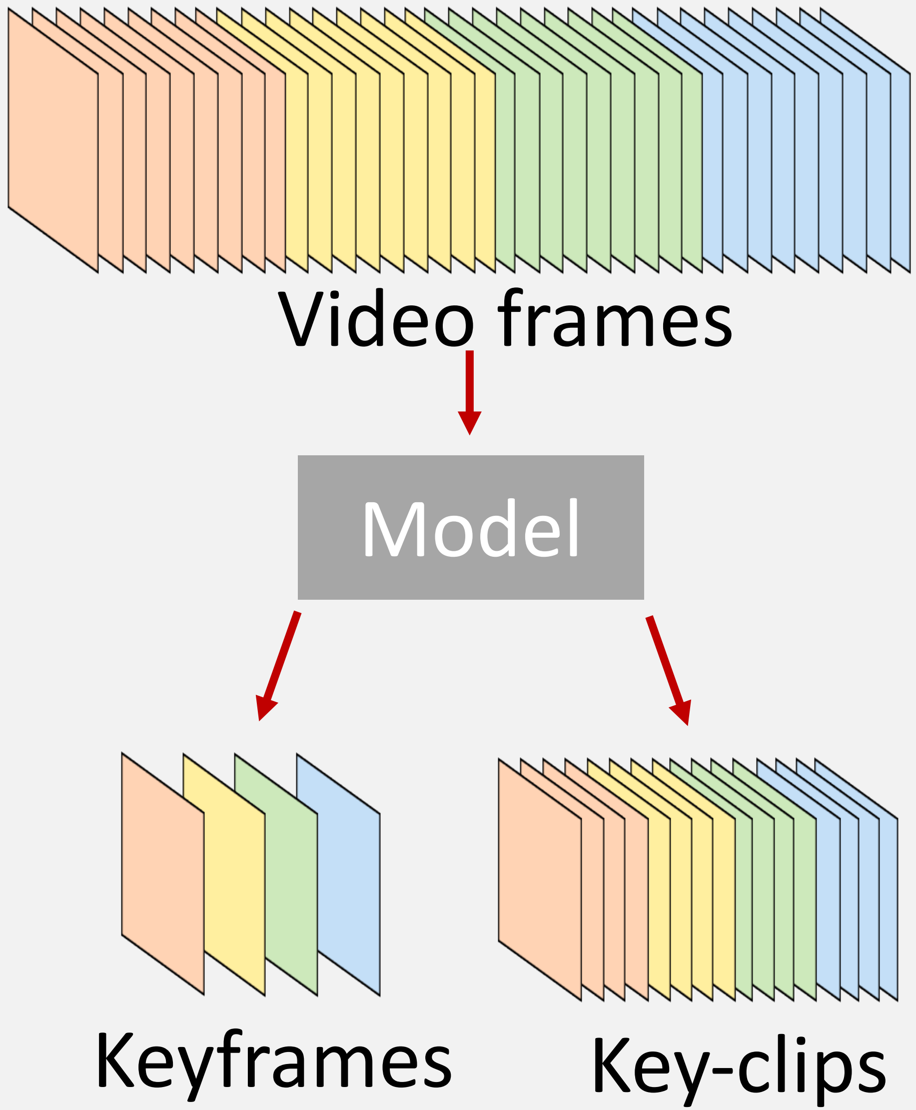
中国科学院深圳先进技术研究院
SHENZHEN INSTITUTES OF ADVANCED TECHNOLOGY
CHINESE ACADEMY OF SCIENCES

Queen Mary
University of London

AAAI 2018

What is video summarization?

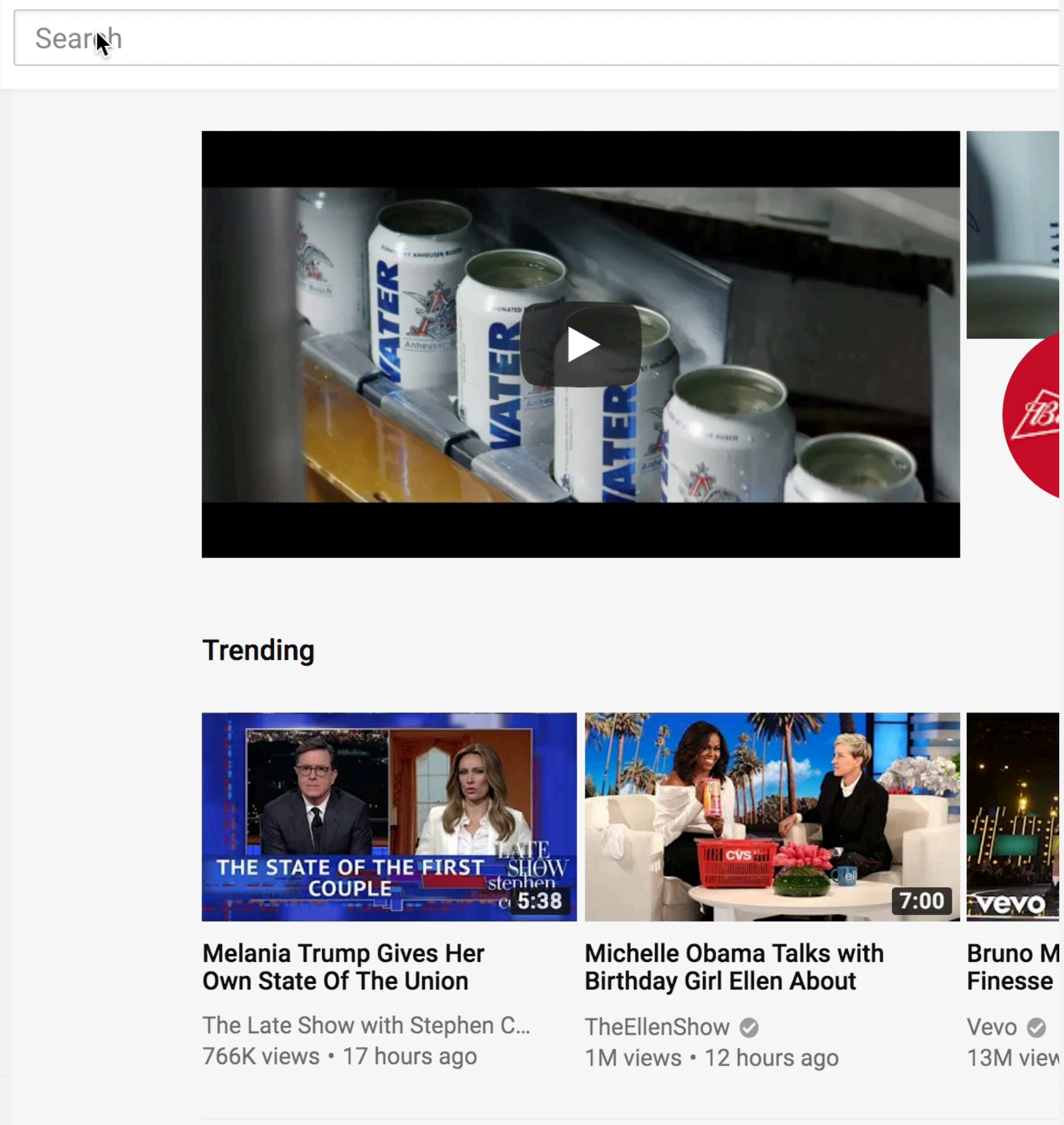
Goal: to automatically summarize videos into keyframes or key-clips.



We want
- Diverse
- Representative

Application of video summarization

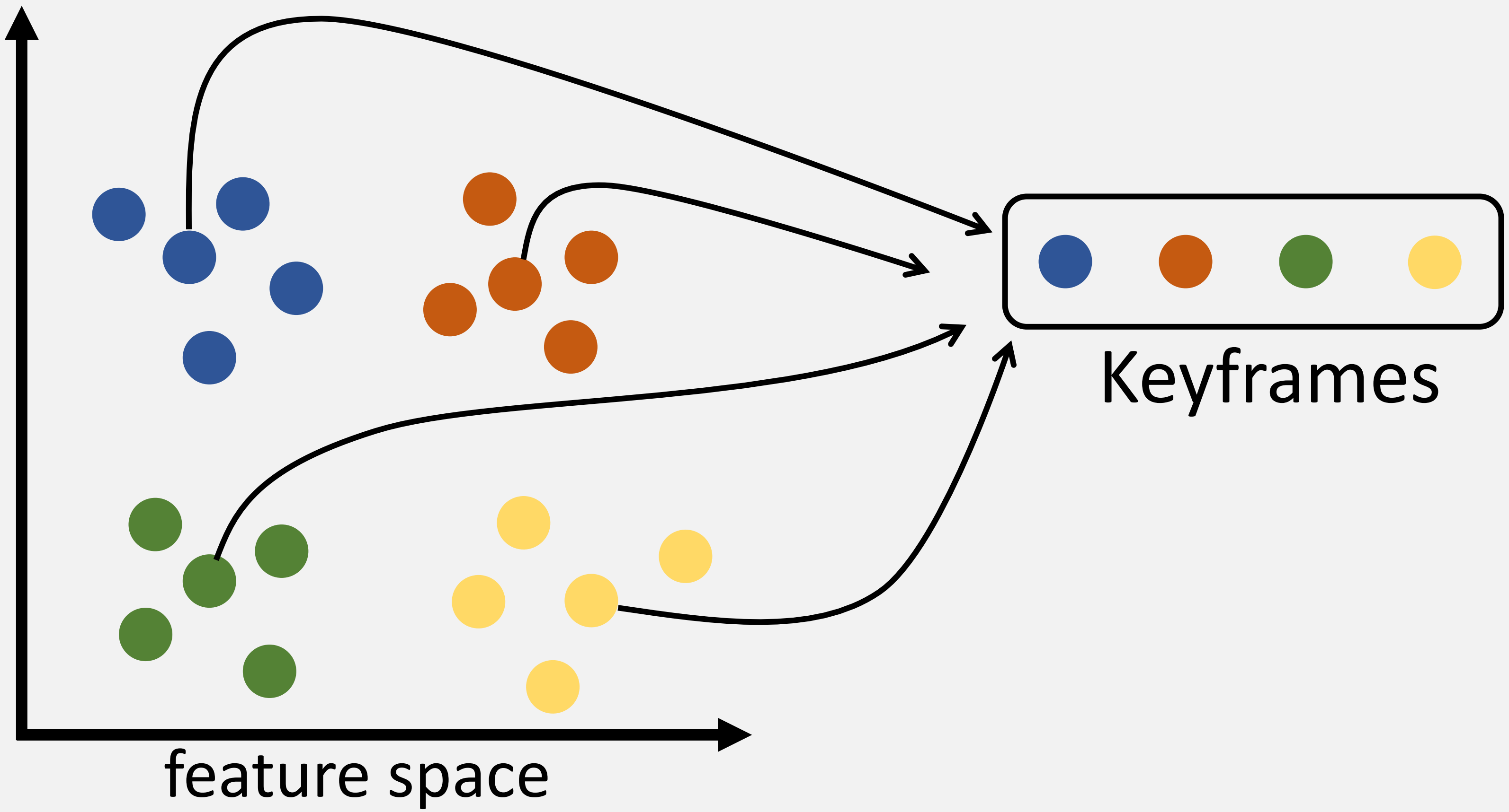
e.g. YouTube video preview



Unsupervised video summarization

Idea: to analyze correlations between frames in feature space

1. Feature extraction 2. Clustering 3. Keyframes extraction

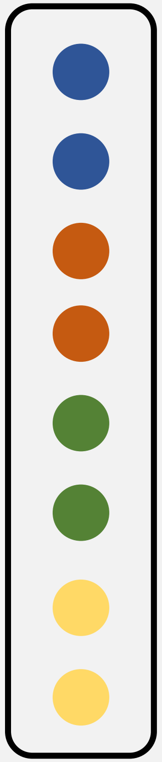


De Avila et al. *Pattern Recognition Letters* 2011.

Supervised video summarization

Idea: to exploit human labels

scores: $y = \{0.1, 0.8, 1.0, 0.2, ...\}$ keyframes: $y = \{0, 1, 1, 0, ...\}$



feature vectors

$\xrightarrow{\text{Training}}$ $\text{loss} = (y - w^T X)^2$

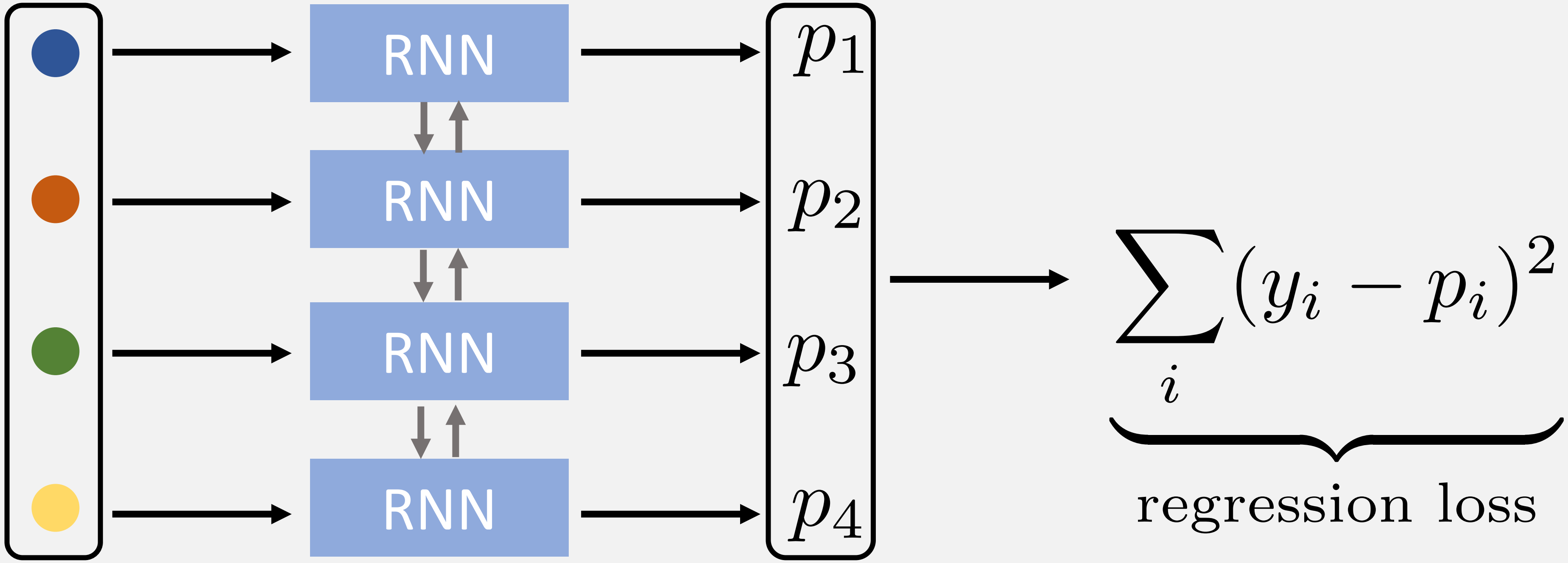
$\xrightarrow{\text{Inference}}$ $p = w^T X'$

- Temporal relations are hard to capture by linear models.

Gygli et al. *ECCV* 2014.

Recurrent neural network with supervised learning

Idea: use RNN to capture temporal relations

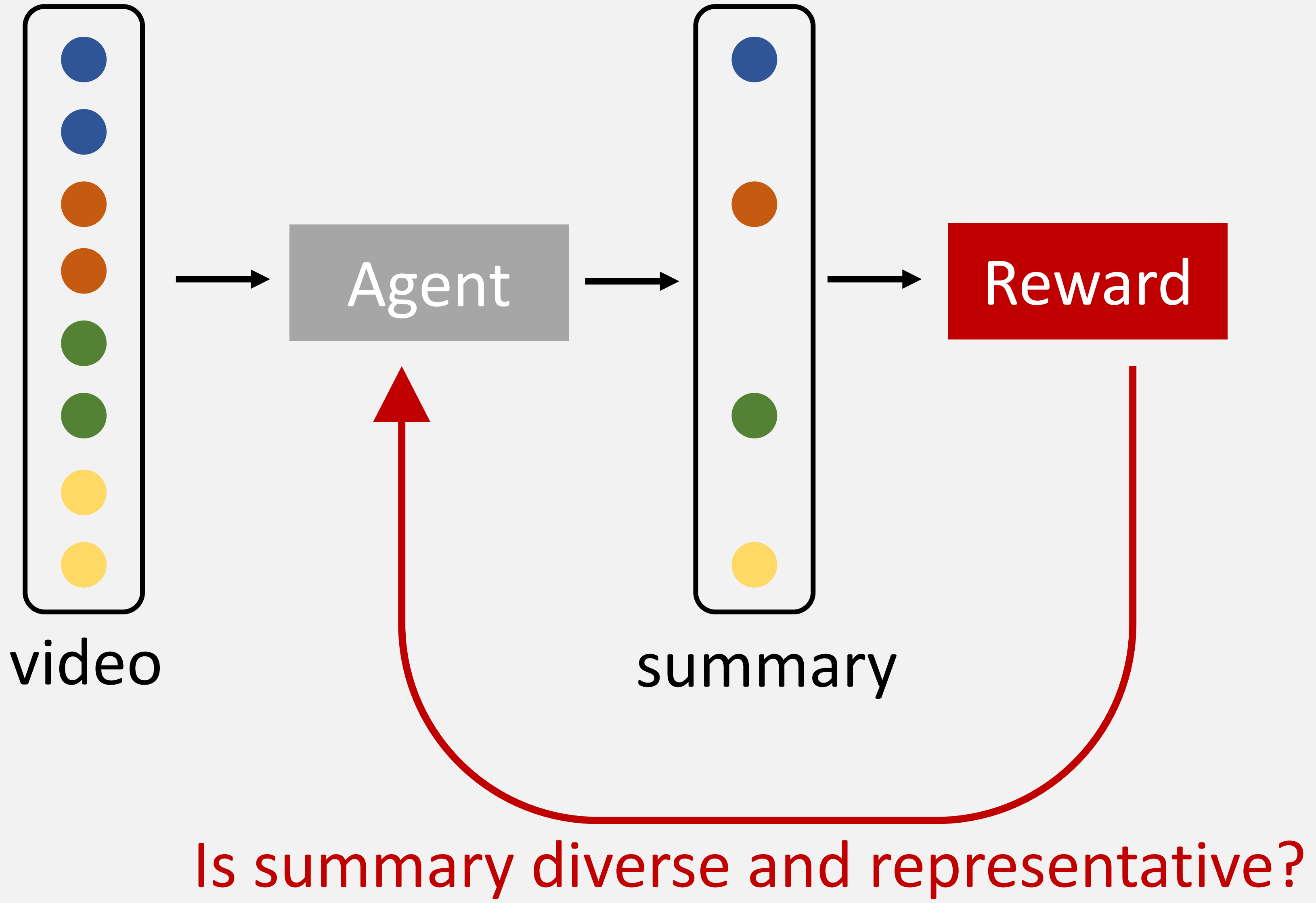


- Collecting labels here is much more expensive than that of other tasks.
- Labels may not provide good supervision signals. (b/c labels are subjective)

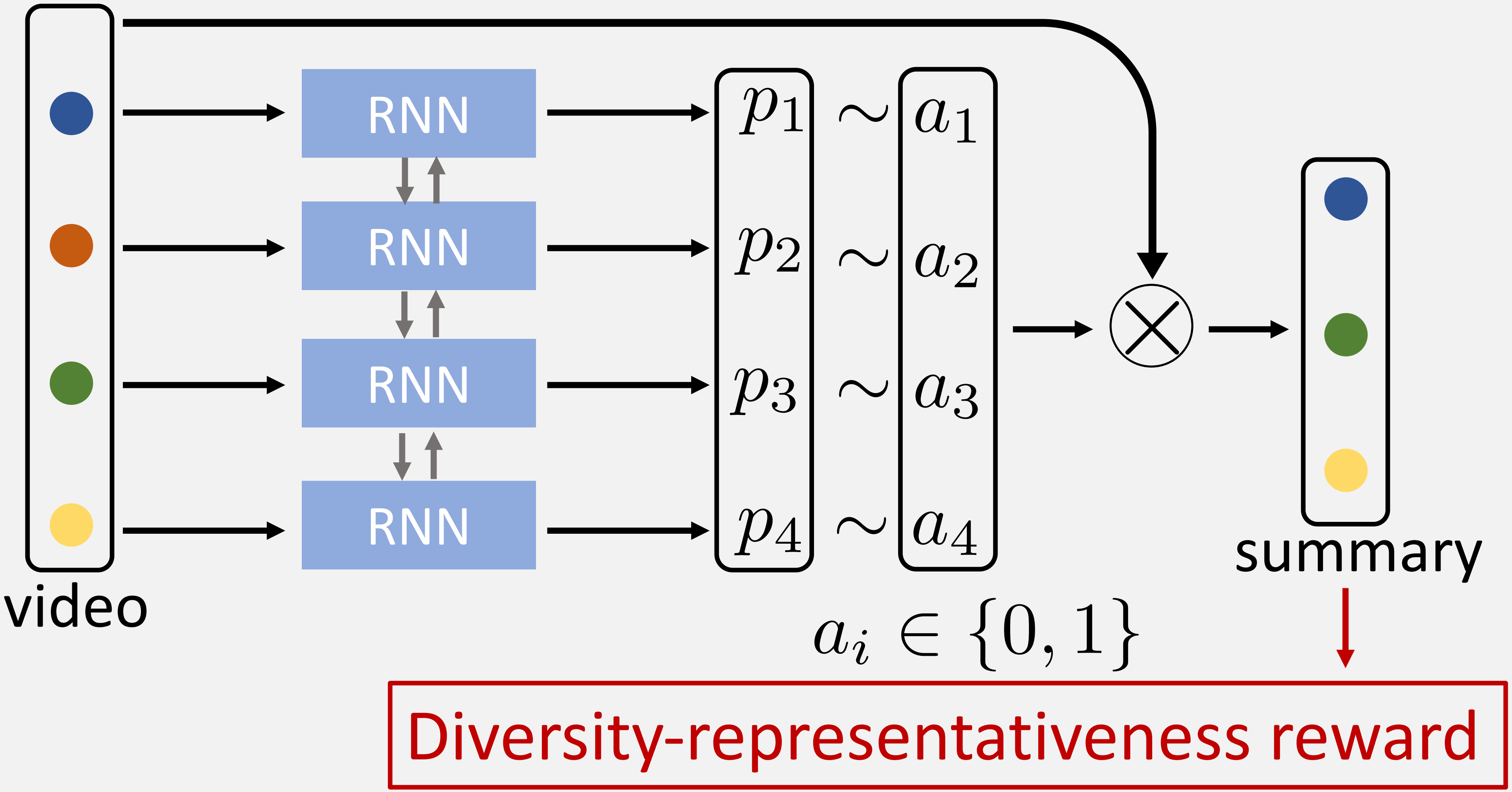
Zhang et al. *ECCV* 2016.

Main idea

To mimic how humans summarize videos



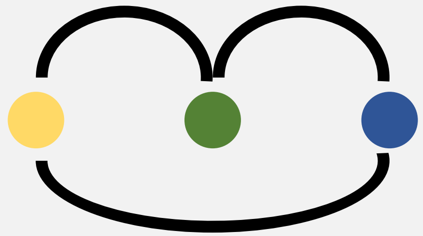
Model


RNN
RNN
RNN
RNN
$p_1 \sim a_1$
$p_2 \sim a_2$
$p_3 \sim a_3$
$p_4 \sim a_4$
video
summary
$a_i \in \{0, 1\}$
Diversity-representativeness reward

Diversity reward

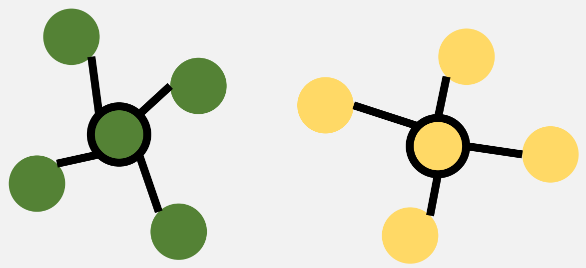
$$R_{\text{div}} = \frac{1}{|\mathcal{Y}|(|\mathcal{Y}|-1)} \sum_{t \in \mathcal{Y}} \sum_{\substack{t' \in \mathcal{Y} \\ t' \neq t}} d(x_t, x_{t'})$$

Set of selected frames



Representativeness reward

$$R_{\text{rep}} = \exp\left(-\frac{1}{T} \sum_{t=1}^{T} \min_{t' \in \mathcal{Y}} ||x_t - x_{t'}||_2\right)$$



Optimization

Reward:

$$R = R_{\text{div}} + R_{\text{rep}}$$

Objective function:

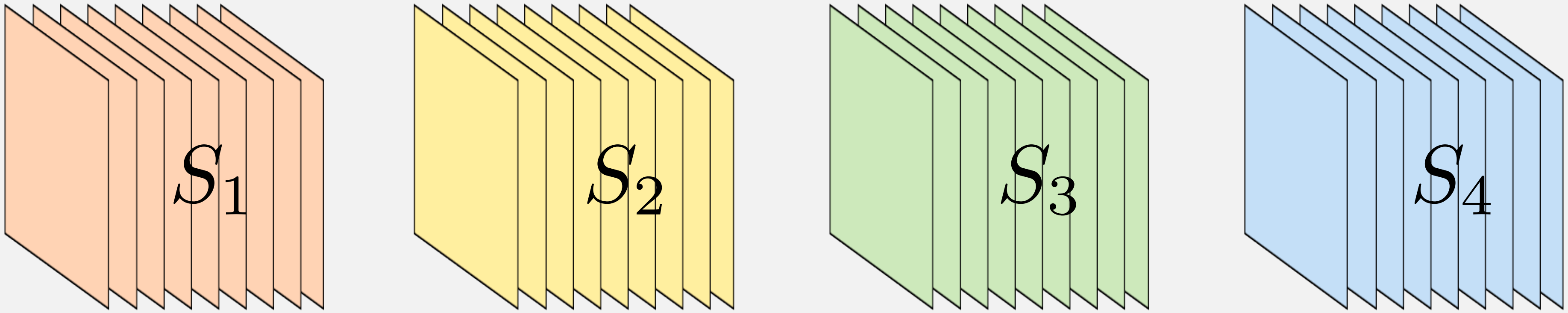
$$J(\theta) = \mathbb{E}[R]$$

Approximate gradients via REINFORCE:

$$\nabla_\theta J(\theta) \approx \frac{1}{N} \sum_{n=1}^{N} \sum_{t=1}^{T} (R_n - b) \nabla_\theta \log \pi_\theta(a_t | h_t)$$

Williams. *Machine Learning* 1992.

Inference



Score prediction:

$$\{p_i\}_{i=1}^{T} = \mathrm{RNN}(\{x_i\}_{i=1}^{T})$$

Compute clip-level scores:

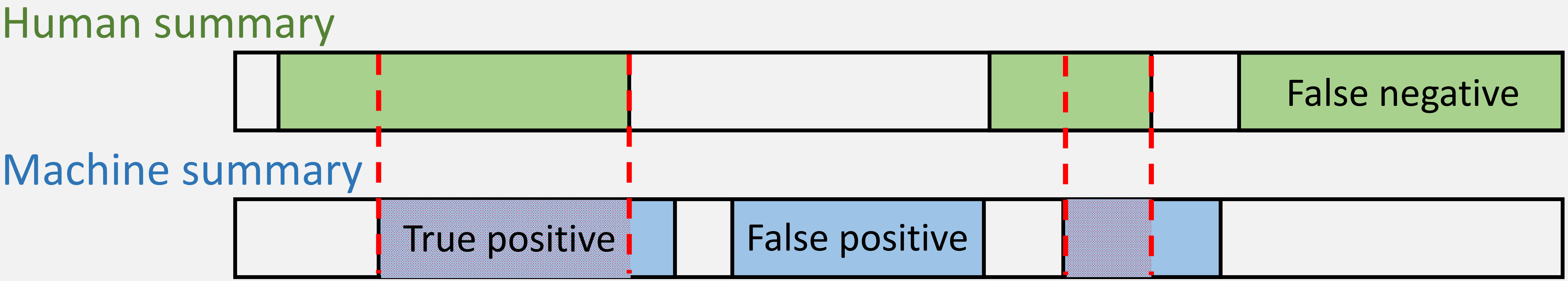
$$I(S_k) = \frac{1}{|S_k|} \sum_{i \in S_k} p_i$$

Select clips (0/1 Knapsack):

$$\arg\max_{\mu} \sum_k \mu_k I(S_k), \quad \sum_k \mu_k |S_k| \leq \gamma, \quad \mu_k \in \{0, 1\}$$

Song et al. *CVPR* 2015.

Evaluation



Metric: *F-score* = (2 x precision x recall) / (precision + recall)

Dataset	# videos	Length (mins)	Description	# annotators per video
SumMe	25	1-6	User videos	15-18
TVSum	50	2-10	YouTube videos	20

Gygli et al. *ECCV* 2014, Song et al. *CVPR* 2015.

Quantitative Results

Table: Comparison with other unsupervised approaches.

Method	SumMe (%)	TVSum (%)
Video-MMR	26.6	-
Uniform sampling	29.3	15.5
K-medoids	33.4	28.8
Vsumm	33.7	-
Web image	-	36.0
Dictionary selection	37.8	42.0
Online sparse coding	-	46.0
Co-archetypal	-	50.0
GAN_{dpp}	39.1	51.7
Ours	**41.4**	**57.6**

$\uparrow 6\%$ $\uparrow 11\%$

Quantitative Results

Table: Comparison with other supervised approaches.

Method	SumMe (%)	TVSum (%)
Interestingness	39.4	-
Submodularity	39.7	-
Summary transfer	40.9	-
Bi-LSTM	37.6	54.2
DPP-LSTM	38.6	54.7
GAN_{sup}	**41.7**	56.3
Ours	41.4	**57.6**

$\uparrow 2\%$

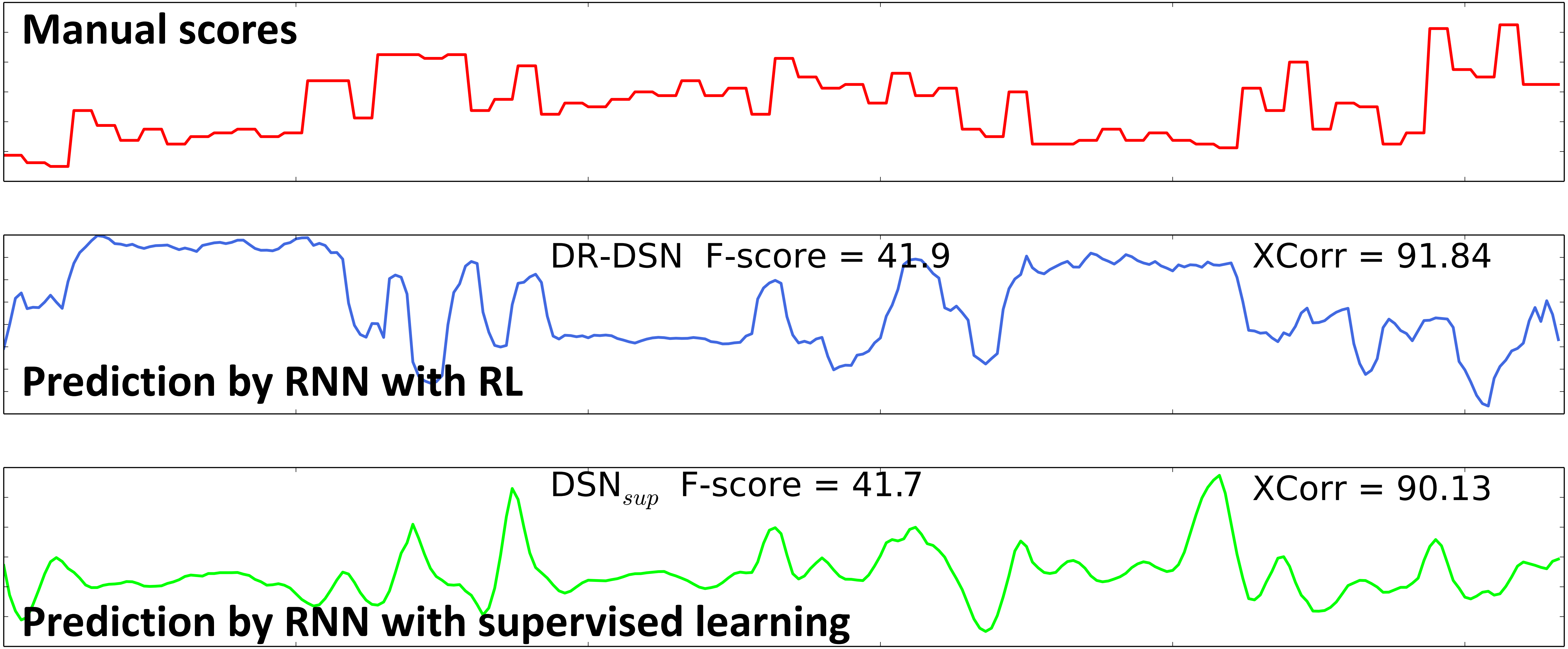
Quantitative Results

Table: Comparison with other supervised approaches.

Method	SumMe (%)	TVSum (%)
Interestingness	39.4	-
Submodularity	39.7	-
Summary transfer	40.9	-
Bi-LSTM	37.6	54.2
DPP-LSTM	38.6	54.7
GAN_{sup}	41.7	56.3
Ours	41.4	57.6
Ours (supervised)	**42.1**	**58.1**

For more experiments and details, please see our paper.

Qualitative Results



Video #10 in TVSUM

Summary

1. Proposed a label-free reward.
2. Outperformed/competitive to other unsupervised/supervised ones.
3. Extended the unsupervised method to the supervised version.

Improvements:

1. Incorporate video segmentation into the end-to-end pipeline.
2. Reduce variance during training. (actor-critic?)
3. Improve the model to deal with long videos. (memory network?)

Code and data available at: https://github.com/KaiyangZhou/vsumm-reinforce

Thanks!

Any questions?

please feel free to contact me at: k.zhou@qmul.ac.uk