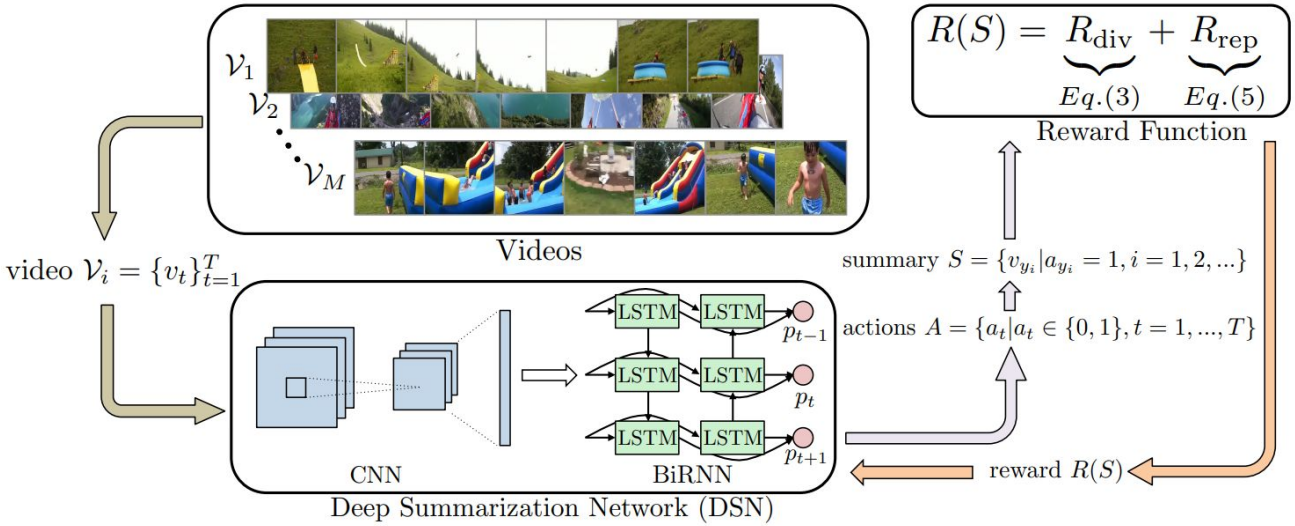
Deep Reinforcement Learning for Unsupervised Video Summarization with Diversity-Representativeness Reward

Kaiyang Zhou et al. AAAI-18

reviewer : 강동구



목차

Introduction

Video Summarization(VS)란, 원본영상을 잘 나타내는(?) 요약(⅛) 영상을 생성하는 것.

이 논문은 VS를 Sequential decision making process로 여김. → RL 적용.

방법 :

1. Deep Summarization Network(DSN) (encoder(CNN)-decoder(bi-LSTM))
2. diversity(차별성), representativeness(대표성)을 반영한 reward function

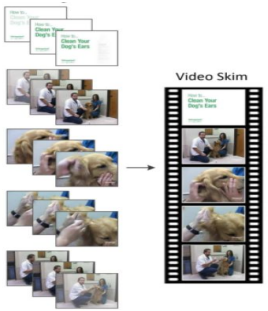
성능 : 2018년 기준, unsupervised 방법론중 최고, supervised 방법론과도 필적

(*그리고, 구글에 “python video summarization” 검색 시 최상단에 위치)

Background

기존 영상 요약 방법

1. Video storyboard : key frame의 slide show. 각 프레임들간 연결이 매끄럽지 못함.
2. Video skim : 하나의 영상을 여러개의 segement 영상으로 나눠, 후에 key segment를 이어 붙인 방식



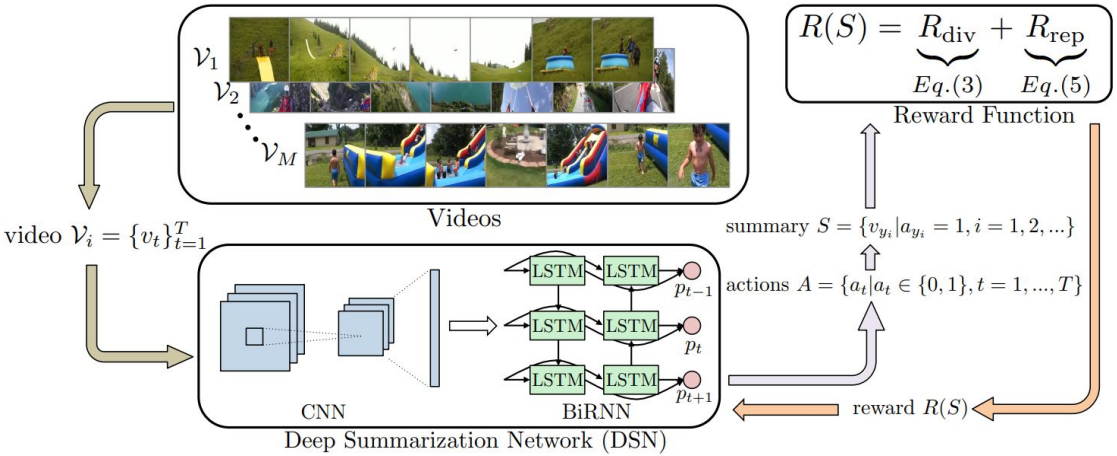
1. offline : 녹화된 영상을 요약
2. online : 실시간 영상 요약

주요 데이터셋

1. SumMe : 휴양지 스포츠 분야의 1~6분 길이의 25개 영상을 15~18명이 annotation
2. TVSum : 뉴스, 다큐먼터리의 2~10분 길이의 50개 영상을 20명이 annotation

Deep Summarization Network(DSN)

Encoder-Decoder 구조. Encoder로 CNN기반의 Googlenet 사용. Decoder는 bi-LSTM



encoder로 프레임의 특징 추출(feature extraction)
decoder로 프레임의 특징으로 확률값 p 생성

$$p_t = \sigma(Wh_t), \quad (1)$$

$$a_t \sim \text{Bernoulli}(p_t), \quad (2)$$

*Encoder는 특징 추출용도로 학습하지 않고, Decoder만 학습.

Diversity-Representativeness Reward Function (1)

Diversity reward R_{div}

: measuring dissimilarity among the selected frames

the selected frames be $\mathcal{Y} = \{y_i | a_{y_i} = 1, i = 1, ..., |\mathcal{Y}|\}$, we compute R_{div} as the mean of the pairwise dissimilarities among the selected frames:

$$R_{\text{div}} = \frac{1}{|\mathcal{Y}|(|\mathcal{Y}| - 1)} \sum_{t \in \mathcal{Y}} \sum_{\substack{t' \in \mathcal{Y} \\ t' \neq t}} d(x_t, x_{t'}), \tag{3}$$

where $d(\cdot, \cdot)$ is the dissimilarity function calculated by

$$d(x_t, x_{t'}) = 1 - \frac{x_t^T x_{t'}}{||x_t||_2 ||x_{t'}||_2}. \tag{4}$$

(3)식이 프레임의 시간적 특성은 고려하지 않으므로, 람다(=20)를 도입.

$d(x_t, x_{t'}) = 1$ if $|t - t'| > \lambda$, where λ controls the degree of temporal distance. We will validate this hypothesis in the Experiments section.

Diversity-Representativeness Reward Function (2)

Representativeness reward R_{rep}
: 해당 프레임이 주변 프레임을 얼마나 대표하는지(k-medoids problem)

$$R_{\text{rep}} = \exp(-\frac{1}{T}\sum_{t=1}^{T}\min_{t' \in \mathcal{Y}}||x_t - x_{t'}||_2). \qquad (5)$$

두 종류의 Reward 절대값이 비슷했으며, 학습도중에도 그랬다고 함.
만약 action으로 선택된 frame이 한개도 없다면, zero reward

$$R(S) = R_{\text{div}} + R_{\text{rep}}.$$

Training & Regularization

Policy gradient 계열의 REINFORCE 알고리즘 활용.

$$\nabla_\theta J(\theta) = \mathbb{E}_{p_\theta(a_{1:T})}[R(S)\sum_{t=1}^{T}\nabla_\theta \log \pi_\theta(a_t|h_t)], \qquad (8)$$

Expectation을 구하기 위해 같은 비디오를 N(=5)번 반복하여 기대값 계산

$$\nabla_\theta J(\theta) \approx \frac{1}{N}\sum_{n=1}^{N}\sum_{t=1}^{T} R_n \nabla_\theta \log \pi_\theta(a_t|h_t), \qquad (9)$$

Training & Regularization

variance를 줄이기 위해 baseline 도입

$$\nabla_\theta J(\theta) \approx \frac{1}{N}\sum_{n=1}^{N}\sum_{t=1}^{T}(R_n - b)\nabla_\theta \log \pi_\theta(a_t|h_t), \qquad (10)$$

선택을 단순히 많이 할 수록 Reward가 커지는 현상을 방지하기 위해 아래식 활용

$$L_{\text{percentage}} = ||\frac{1}{T}\sum_{t=1}^{T} p_t - \epsilon|^2, \qquad (11)$$

```
cost = args.beta * (probs.mean() - 0.5)**2 # minimize summary length penalty term [Eq.11]
```

Experiments

(c) DR-DSN

각 프레임별 회색수치가 높을수록 중요한 프레임, 연두색이 예측한 프레임(수치는 F1-score)

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
DR-DSN	**41.4**	**57.6**

un-supervised

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
$DR\text{-}DSN_{sup}$	**42.1**	**58.1**

supervised

주관적 견해(비판?)

공개된 코드가 단순 성능만 확인되는 수준 and 오래됨..

encoder-decoder 구조가 end-to-end가 아님.

즉, 일반 사용자가 custom 영상을 encoder를 거친 데이터셋 형태로 변형해야 함.

+ 영상의 Change Point를 사전에 명시해야함(이를 찾는 방법은 또 별개…)