Reconstructive Sequence-Graph Network for Video Summarization

合肥工業大學

苏伊阳

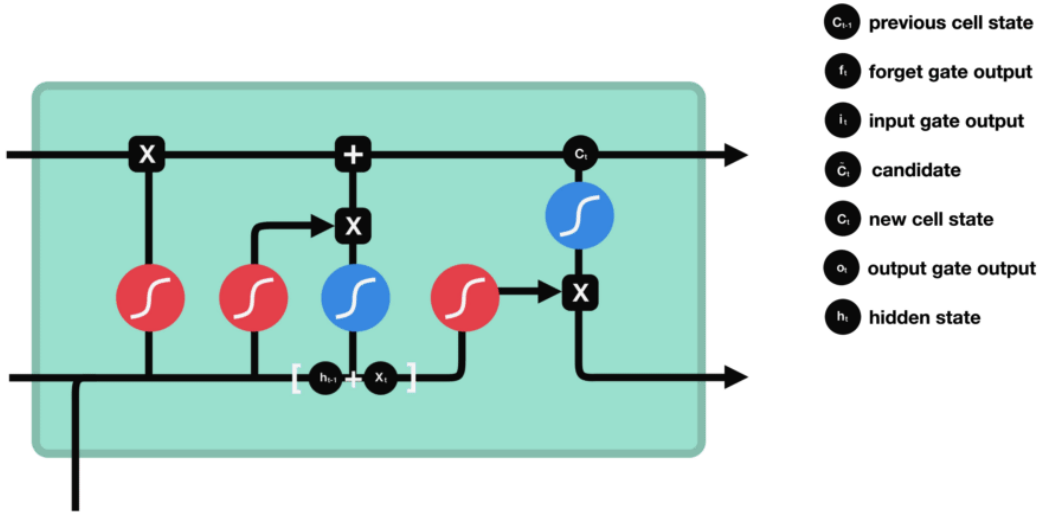
2022.12.10

[1] Zhao B, Li H, Lu X, et al. Reconstructive Sequence-Graph Network for Video Summarization[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2022, 44(5): 2793–2801.

研究背景

目前，许多视频摘要是基于RNN的方法，这种方法会将视频数据转换为帧序列，并利用序列中的时间依赖性来总结视频，但是，基于RNN的方法能够捕获局部邻域依赖，但通常无法处理全局长距离依赖，并且容易被噪声干扰。

一般来说，一个镜头中的帧记录了一个特定的活动，并随着时间的推移而平滑变化，对于剪辑之后的视频，**多跳关系**在镜头之间频繁发生。在这种情况下，局部和全局依赖关系对于理解视频内容都很重要。

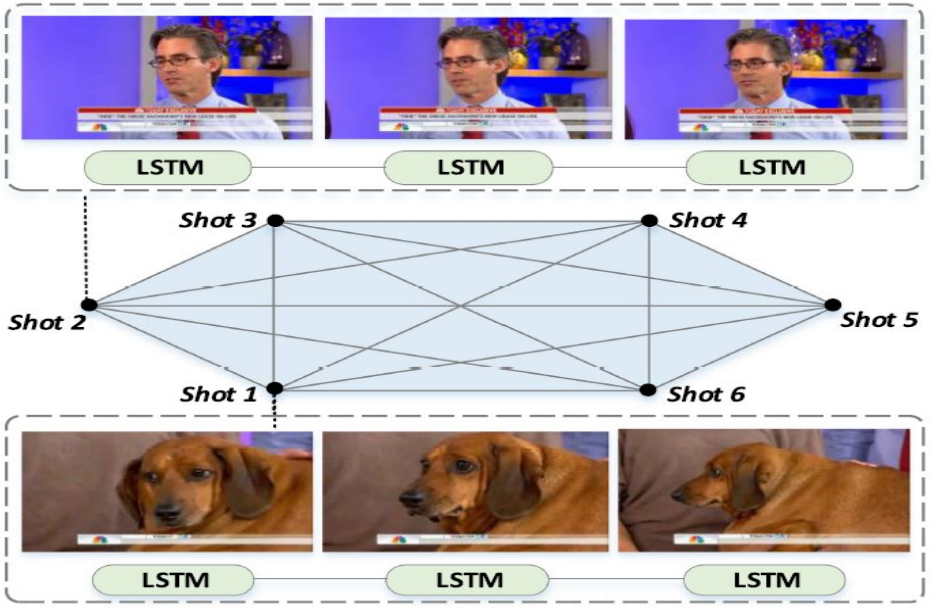


研究背景

本文提出了一种重构序列图网络(RSGN)，将帧和镜头分层编码为序列和图，其中帧级依赖（局部依赖关系）由长短期记忆(LSTM)编码，镜头级依赖（全局依赖关系）由图卷积网络(GCN)编码。

镜头是帧与视频之间的中间状态，由几个连续的帧组成。

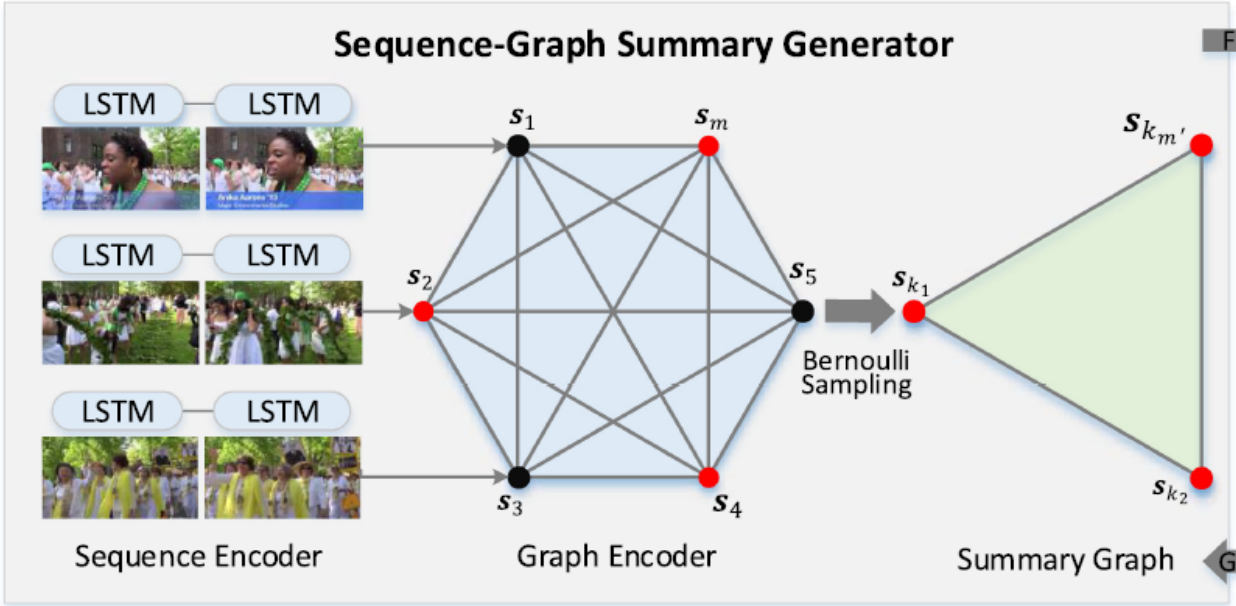
- 镜头内的帧适合用RNN建模为时间序列，因为它们很短，且随时间平稳变化。
- 不同镜头之间信息差异很大，使得相邻镜头之间的关系不像帧那样紧密，对于编辑后的视频来说，相邻镜头的内容甚至没有明显的时间依赖性，将视频镜头建模为一个完整的图更为合适。



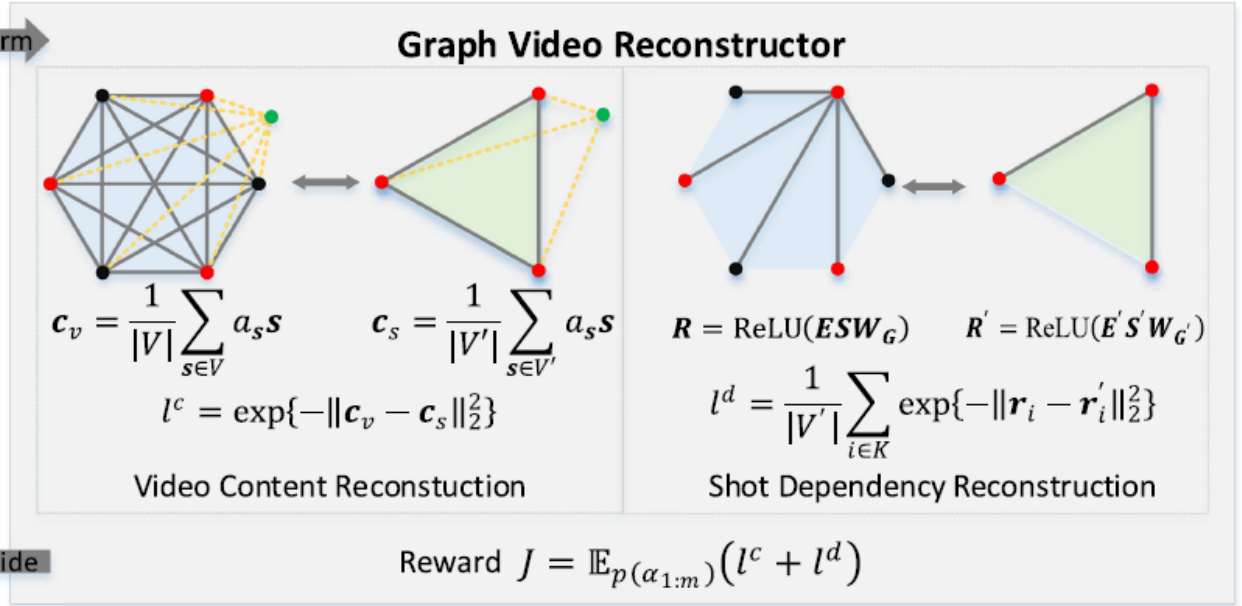
研究方法

贡献点：

- 设计了序列图模型，利用LSTM和GCN分层捕获镜头内时间依赖和镜头间依赖，有效地避免镜头位置距离造成的干扰。
- 摘要的图模型被构造为重构器，以一种无监督的方式优化生成器，并保留视频内容和镜头级依赖关系。
- 在数据集上验证，性能较好。



Form



Guide

Reward $J = \mathbb{E}_{p(\alpha_{1:m})}(l^c + l^d)$

研究方法

图卷积神经网络GCN

图：G=(V, E) 表示图结构 (有向图或无向图)，V 表示节点的集合，E 表示边的集合

$$H^{(l+1)} = \sigma\left(\tilde{D}^{-\frac{1}{2}}\tilde{A}\tilde{D}^{-\frac{1}{2}}H^{(l)}W^{(l)}\right)$$

$$\tilde{A} = A + I$$

$$\tilde{D}_{ii} = \sum_j \tilde{A}_{ij}$$

图卷积计算公式

$$H^{(l+1)} = f\left(H^{(l)}, A\right) = \sigma(AH^{(l)}W^{(l)})$$

A是邻接矩阵，H为所有节点的特征向量矩阵，W是卷积的参数

$$H^{(0)} = X \in R^{n\times d}$$

$$\begin{bmatrix}0&1&1&0\\1&0&0&0\\1&0&0&1\\0&0&1&0\end{bmatrix}\begin{bmatrix}1&1&1&1&1\\2&2&2&2&2\\3&3&3&3&3\\4&4&4&4&4\end{bmatrix}=\begin{bmatrix}5&5&5&5&5\\1&1&1&1&1\\5&5&5&5&5\\3&3&3&3&3\end{bmatrix}$$

A　　H　　AH

存在问题

AH 只获得了某个节点的邻居信息，而忽略了节点本身信息

$$H^{(l+1)} = \sigma\left(\tilde{A}H^{(l)}W^{(l)}\right)$$

$$\tilde{A} = A + I_N$$

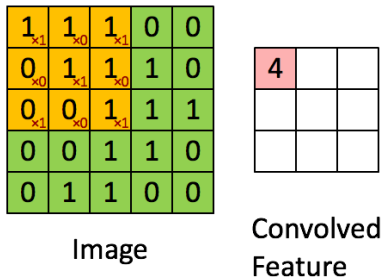
矩阵 A 没有归一化，这样经过多层卷积后向量的值会很大。

对称归一化

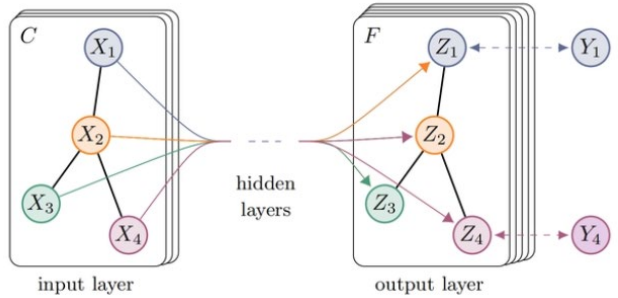
$$H^{(l+1)} = \sigma\left(\tilde{D}^{-\frac{1}{2}}\tilde{A}\tilde{D}^{-\frac{1}{2}}H^{(l)}W^{(l)}\right)$$

$$\tilde{A} = A + I$$

$$\tilde{D}_{ii} = \sum_j \tilde{A}_{ij}$$



卷积神经网络CNN



图积神经网络GCN

https://baijiahao.baidu.com/s?id=1670303834781877028&wfr=spider&for=pc

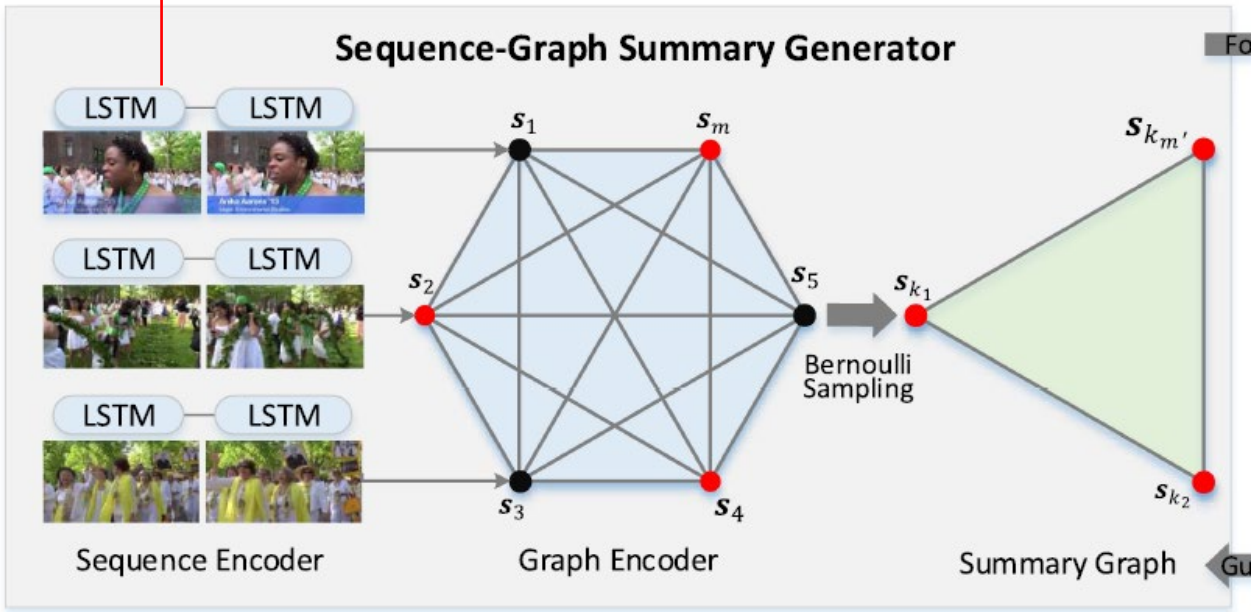
研究方法

- 设计了序列图模型，利用LSTM和GCN分层捕获镜头内时间依赖和镜头间依赖，有效地避免镜头位置距离造成的干扰。

单个视频的帧：$\{f_1, f_2, \cdots, f_n\}$
镜头边界：$\{b_0, b_1, \cdots, b_m\}$　$b_0 = 1, b_m = n$
第i个镜头的帧：$\{f_{b_{i-1}+1}, f_{b_{i-1}+2}, \cdots, f_{b_i}\}$

$$\boldsymbol{h}_t = \mathrm{B}iLSTM\left(\boldsymbol{f}_t, \boldsymbol{h}_{t-1}\right),\ t \in [b_{i-1}+1, b_i],$$

h_{b_i}最后一帧的隐藏特征，编码了第i个镜头的前向后向的时序依赖　$h_{b_i} = s_i$



Shot-Level Graph Encoder

$G = (V, E)$　V表示图的节点，E表示图的边

$V = \{s_1, s_2, \cdots, s_m\}$　$E = \{E_{11}, \cdots, E_{ij}, \cdots, E_{mm}\}$

使用不相似度作为边的权重，相似的镜头不需要共享信息，因为已经有相似的语义特征，而差异巨大的镜头需要更多交互，以全面建模整个视频。

点乘 $e_{ij} = -\phi(s_i)^{\mathrm{T}}\varphi(s_j).$

高斯函数 $e_{ij} = \exp\left\{-\phi(s_i)^{\mathrm{T}}\varphi(s_j)\right\}.$

拼接 $e_{ij} = W_e^{\mathrm{T}}\left[\phi(s_i), \varphi(s_j)\right].$

图卷积GCN: $\boldsymbol{R} = \mathrm{ReLU}(\boldsymbol{ESW}_G),$

$E = (e_{ij})_{m\times m}$

$S = [\tau(s_1); \tau(s_2); \cdots; \tau(s_m)]$

$R = [r_1; r_2; \cdots; r_m]$

编码了镜头间的关系

W_ϕ W_φ是linear embedding function$\phi(\cdot),, \varphi(\cdot)$的参数

被选为关键镜头的概率是由镜头特征及其与整个视频内容的关系共同决定的。

$$p_i = Sigmoid\left(\boldsymbol{W}_p[\tau(\boldsymbol{s}_i), r_i] + b_p\right),$$

$$\alpha_i = \mathrm{B}ernoulli(p_i),\ i = 1, 2, \ldots, m.$$

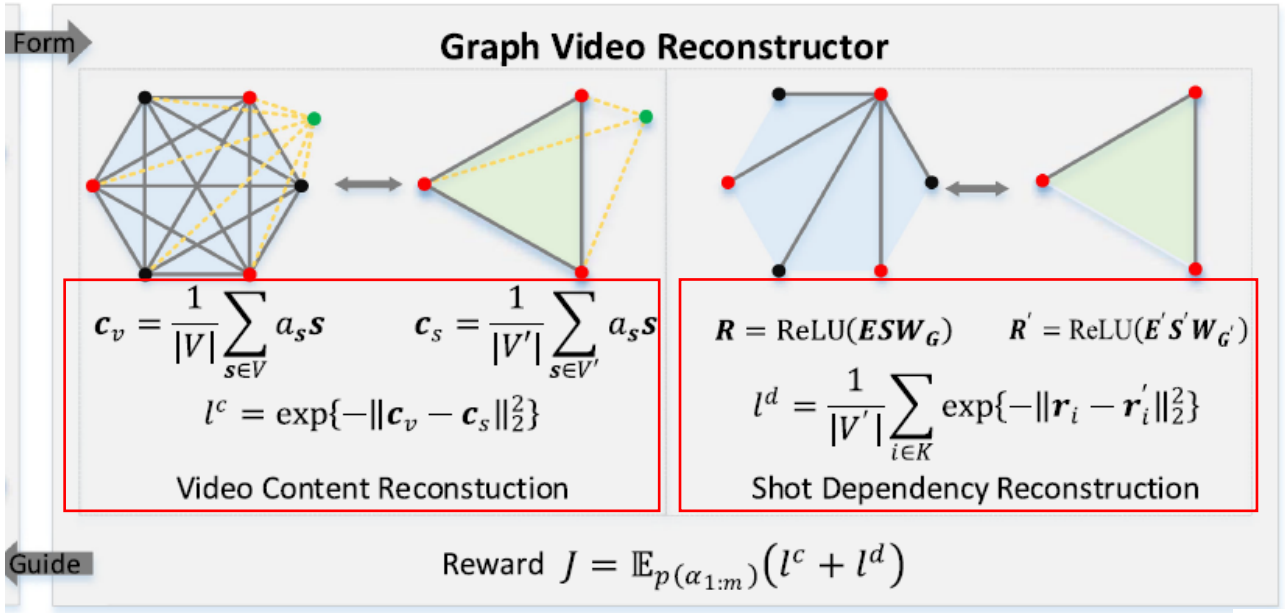
$\alpha_i \in \{0,1\}$

研究方法

■ 摘要的图模型被构造为重构器，以一种无监督的方式优化生成器，并保留**视频内容**和**镜头级依赖**关系。

选择的镜头的索引$K = \{k_j\}_{j=1}^{m'}$

a_s表示镜头s标注的重要性分数或者GCN输出的概率



$G' = (V', E')$ V'表示摘要图的节点， E'表示摘要图的边

$V' = \{s_{k_1}, s_{k_2}, \cdots, s_{k_{m'}}\}$ $E' = \{E_{k_1,k_1}, \cdots, E_{k_i,k_j}, \cdots, E_{k_{m'},k_{m'}}\}$

点乘 $e_{ij} = -\phi(s_i)^{\mathrm{T}}\varphi(s_j)$.

高斯函数 $e_{ij} = \exp\{-\phi(s_i)^{\mathrm{T}}\varphi(s_j)\}$.

拼接 $e_{ij} = W_e^{\mathrm{T}}[\phi(s_i), \varphi(s_j)]$.

图卷积GCN: $R' = \mathrm{ReLU}(E'S'W_{G'})$,

$E = (e_{ij})_{m\times m}$

$S = [\tau'(s_{k_1}); \tau'(s_{k_2}); \cdots; \tau'(s_{k_{m'}})]$

$R' = [r'_{k_1}; r'_{k_2}; \cdots; r'_{k_{m'}}]$ 编码了摘要镜头间的关系

W'_ϕ W'_φ是linear embedding function$\phi(\cdot), \varphi(\cdot)$的参数

监督损失: $l^p = \frac{1}{m}\|p - g\|_2^2$,

损失函数: $\mathcal{L}(\theta) = l^p + l^r - J$, $\nabla_\theta J = \frac{1}{n}\sum_{j=1}^{n}\sum_{i=1}^{m}(l^c + l^d)_j \nabla_\theta \log \pi_\theta(\alpha_i|s_i, r_i)$,

正则项: $l^r = \left(\frac{1}{m}\sum_{i=1}^{m} p_i - \varepsilon\right)^2$,

重构器倾向于选择更多的镜头来增加奖励

实验

- 数据集
  - **SumMe**：包含25个视频，时长从1.5分钟到6.5分钟不等。每个视频由15-18个用户注释，包括帧级别的重要性评分和基于镜头的视频摘要。
  - **Tvsum**：有50个视频，时长从2分钟到10分钟不等。每个视频由20个用户用镜头级别的重要性评分注释。
- 预处理

  采用GoogLeNet的pool5层进行帧特征提取，维数为1024。另外，利用KTS内核时间分割将每个视频分割成镜头

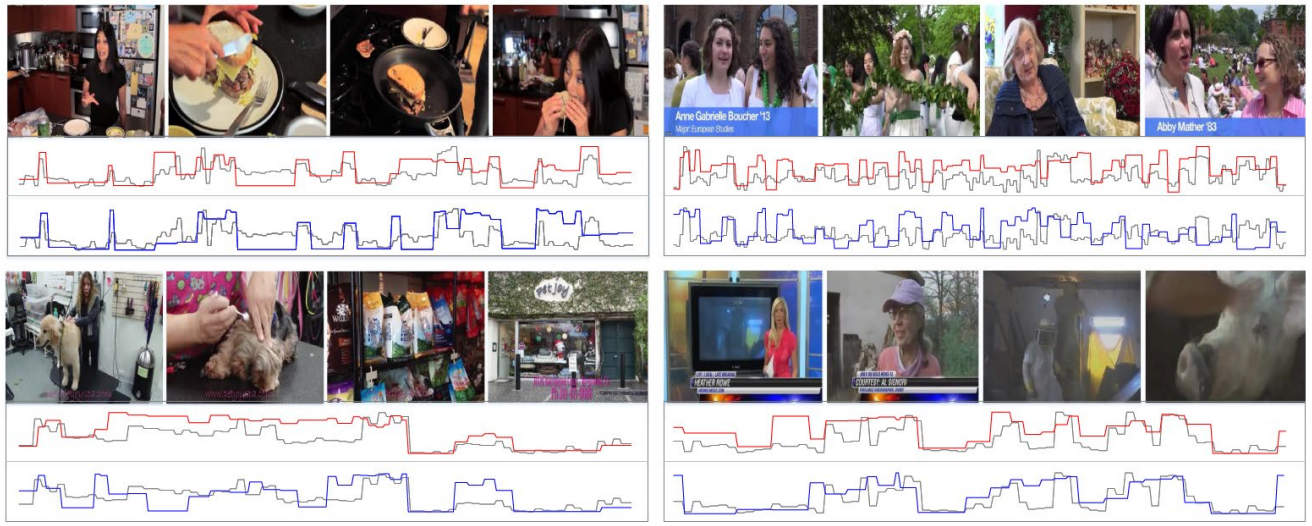


Fig. 3. The summarization results of RSGN and RSGN$_{uns}$. The images are sampled from the summaries generated by RSGN. The curves denote the distributions of importance scores. The gray curves depict the ground truth score, while the red/blue curves depict the score predicted by the supervised/unsupervised model, respectively.

TABLE 5
The Results With Different Training Settings on the SumMe and TVsum Datasets

Datasets	SumMe			TVsum		
Approaches	Canonical	Augmented	Transfer	Canonical	Augmented	Transfer
SUM-GAN [25]	0.387	0.417	–	0.508	0.589	–
DR-DSN [52]	0.414	0.428	0.424	0.576	0.584	0.578
vsLSTM [47]	0.376	0.416	0.407	0.542	0.579	0.569
dppLSTM [47]	0.386	0.429	0.418	0.547	0.596	0.587
SUM-GAN$_{sup}$ [25]	0.417	0.436	–	0.563	0.612	–
DR-DSN$_{sup}$ [52]	0.421	0.439	0.426	0.581	0.598	0.589
H-RNN [49]	0.421	0.438	–	0.579	0.619	–
HSA-RNN [50]	0.423	0.421	–	0.587	0.598	–
re-SEQ2SEQ [48]	0.425	0.449	–	**0.603**	**0.639**	–
VASNet [7]	0.424	0.425	0.419	0.589	0.585	0.547
RSGN$_{uns}$	0.423	0.436	0.412	0.580	0.591	0.597
RSGN	**0.450**	**0.457**	**0.440**	0.601	0.611	**0.600**

Q&A

合肥工業大學