软件学报 ISSN 1000-9825, CODEN RUXUEW
Journal of Software,2022,33(9):3195−3209 [doi: 10.13328/j.cnki.jos.006621]

E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel: +86-10-62562563

基于空时变换网络的视频摘要生成*

李 群, 肖 甫, 张子屹, 张 锋, 李延超

(南京邮电大学 计算机学院、软件学院、网络空间安全学院, 江苏 南京 210023)

通信作者: 肖甫, E-mail: xiaof@njupt.edu.cn

摘 要: 视频摘要生成是计算机视觉领域必不可少的关键任务, 这一任务的目标是通过选择视频内容中信息最丰富的部分来生成一段简洁又完整的视频摘要, 从而对视频内容进行总结. 所生成的视频摘要通常为一组有代表性的视频帧 (如视频关键帧) 或按时间顺序将关键视频片段缝合所形成的一个较短的视频. 虽然视频摘要生成方法的研究已经取得了相当大的进展, 但现有的方法存在缺乏时序信息和特征表示不完备的问题, 很容易影响视频摘要的正确性和完整性. 为了解决视频摘要生成问题, 提出一种空时变换网络模型, 该模型包括 3 大模块, 分别为: 嵌入层、特征变换与融合层、输出层. 其中, 嵌入层可同时嵌入空间特征和时序特征, 特征变换与融合层可实现多模态特征的变换和融合, 最后输出层通过分段预测和关键镜头选择完成视频摘要的生成. 通过空间特征和时序特征的分别嵌入, 以弥补现有模型对时序信息表示的不足; 通过多模态特征的变换和融合, 以解决特征表示不完备的问题. 在两个基准数据集上做了充分的实验和分析, 验证了所提模型的有效性.

关键词: 视频摘要生成; 空时变换网络; ViLBERT; 特征融合; 多模态

中图法分类号: TP391

中文引用格式: 李群, 肖甫, 张子屹, 张锋, 李延超. 基于空时变换网络的视频摘要生成. 软件学报, 2022, 33(9): 3195–3209. http://www.jos.org.cn/1000-9825/6621.htm

英文引用格式: Li Q, Xiao F, Zhang ZY, Zhang F, Li YC. Video Summarization Based on Spacial-temporal Transform Network. Ruan Jian Xue Bao/Journal of Software, 2022, 33(9): 3195–3209 (in Chinese). http://www.jos.org.cn/1000-9825/6621.htm

Video Summarization Based on Spacial-temporal Transform Network

LI Qun, XIAO Fu, ZHANG Zi-Yi, ZHANG Feng, LI Yan-Chao

(School of Computer Science, Nanjing University of Posts and Telecommunications, Nanjing 210023, China)

Abstract: Video summarization is an indispensable and critical task in computer vision, the goal of which is to generate a concise and complete video summary by selecting the most informative part of a video. A generated video summary is a set of representative video frames (such as video keyframes) or a short video formed by stitching key video segments in time sequence. Although the study on video summarization has made considerable progress, the existing methods have the problems of deficient temporal information and incomplete feature representation, which can easily affect the correctness and completeness of a video summary. To solve the problems, this study proposes a model based on a spatiotemporal transform network, which includes three modules, i.e., the embedding layer, the feature transformation and fusion layer, and the output layer. Specifically, the embedding layer can simultaneously embed spatial and temporal features, and the feature transformation and fusion layer can realize the transformation and fusion of multi-modal features; finally, the output layer generates the video summary by segment prediction and key shot selection. The spatial and temporal features are embedded separately to fix the problem of deficient temporal information in existing models, and the transformation and fusion of multi-modal features can solve the problem of incomplete feature representation. Sufficient experiments and analyses on two benchmark datasets are conducted, and the results verify the effectiveness of the proposed model.

Key words: video summarization; spacial-temporal transform network; ViLBERT; feature fusion; multi-modal

---

* 基金项目: 国家自然科学基金 (61906099, 61906098)
本文由“融合媒体环境下的媒体内容分析与信息服务技术”专题特约编辑汪萌教授、张勇东教授、俞俊教授以及张伟高级工程师推荐.
收稿时间: 2021-06-29; 修改时间: 2021-08-15; 采用时间: 2022-01-14; jos 在线出版时间: 2022-02-22

在短视频流行火爆的今天, 每分钟都有长达数百乃至上千小时的视频被上传到视频网站或者社交媒体平台上. 这些网络平台的用户群体经常有一种普遍的行为, 那就是快速跳转到一个视频中最感兴趣的或最有趣的部分, 这种行为可被称为“略览”. 很多网络视频平台为了顺应用户的这一行为, 提供了诸如“视频预览” “高能进度条”等功能, 以此来提升用户的浏览体验. 然而, 要在这些包含庞大信息量的视频中提取出有用的信息, 通常情况下需要耗费大量的人力. 若能够智能的进行视频重要片段的检测, 就能够自动模仿人类的“略览”行为, 并尽可能地免除人工干预. 利用计算机视觉技术对视频片段自动进行重要性检测, 一旦部署完毕后便不再需要频繁的人工介入, 相比之下更能满足当前网络大数据量之下的检测需求.

视频重要片段的检测是计算机视觉领域目前必不可少的关键任务, 这一任务通常又被称为视频摘要生成任务[1]. 视频摘要生成可以定义为将长视频转换成只包含基本片段的较短视频, 从而便于观看者快速理解视频内容. 视频摘要生成技术的目标是通过选择视频内容中信息最丰富的部分来生成一段简洁又完整的视频摘要, 从而对视频内容进行总结. 所生成的视频摘要通常为一组有代表性的视频帧 (如视频关键帧) 或按时间顺序将关键视频片段缝合所形成的一个较短的视频. 前一种形式被称为视频故事板, 后一种被称为视频略览. 视频略览的优点是能够包括音频和运动元素, 提供更自然的故事叙述, 并潜在地增强表达性和视频概要所传达的信息量. 此外, 对于观众来说, 观看视频比观看幻灯片往往更有趣. 但是视频故事板不受时间或同步问题的限制, 因此这种形式为那些以浏览和导航为目的使用视频摘要技术的媒体组织提供了更多在数据内容管理上的灵活性.

目前最经典的方法是依赖于深度神经网络的深度学习方法[1−3], 并且取得了一定的成果, 这证明了计算机视觉技术在解决上述问题和挑战所具有的良好前景. 虽然该技术已经取得了相当大的进展, 但现有的视频摘要方法存在缺乏时序信息和特征表示不完备的问题, 很容易影响视频摘要的正确性和完整性. 视频摘要生成方法通常分 3 个步骤[3]: 1) 视频帧特征提取; 2) 视频帧重要性分数预测; 3) 关键镜头选择. 从目前已有的关于视频摘要生成方法的相关研究中可以得知, 视频帧特征提取这一步骤对于视频摘要任务至关重要, 如何能更准确更完备的表示视频帧, 为视频帧重要性分数预测打好基础, 依旧是当前该任务所面临的最大挑战. 虽然先前的相关工作已经着重研究了这一问题, 但是大多数都只拘泥于使用预训练卷积神经网络 (convolutional neural networks, CNN) 提取的视觉图像特征来进行图像分类[4], 忽略了其他不同类型的视觉特征 (比如运动特征) 以及文本特征在重要性分数预测中可能起到的作用.

为了解决上述问题, 本文提出一种空时变换网络模型, 该模型可同时嵌入空间特征和时序特征, 以弥补现有模型对时序信息表示不足的问题. 此外, 特征变换与融合层可实现多模态特征的变换和融合, 以解决现有模型对特征表示不完备的问题. 我们在 SumMe 和 TVSum 两个基准数据集上做了充分的实验和分析, 验证了我们模型的有效性. 本文的创新点主要包括以下 3 点.

(1) 提出一种基于空时变换网络的视频摘要生成方法, 该方法以视频帧的时序和空间特征提取为基础, 迁移学习视觉基础模型的多模态表征, 并通过时序提议生成策略, 有效地解决了视频摘要生成问题. 我们提出的方法成功地将视觉基础模型迁移到了有监督的视频摘要生成模型中, 实现了很好的效果.

(2) 联合时序、空间和文本等多模态表征到视频摘要生成问题中, 以解决视频摘要生成模型对视频中时序和运动特征表示不足的问题. 为了提取视频的时序特征, 我们设计了轻量级的卷积自编码器. 该卷积自编码器作用于梯度化的视觉图像, 能够很好地表征视频中的时序和运动特征, 弥补了视频摘要生成模型对时序和运动特征表征不足的缺点.

(3) 在空时变换网络中, 我们引入自注意力和共同注意力机制. 与原自注意力变换层相比, 共同注意力变换层增加了视觉流和自然语言流的交互, 从而体现了视觉流中图像视觉条件下的自然语言注意力和自然语言流中语言条件下的图像注意力.

1 相关工作

为了完成视频摘要生成任务, 研究者们提出了无监督和有监督的机器学习方法. 下面分别介绍无监督视频摘要和有监督视频摘要的生成方法.

1.1 无监督视频摘要生成

无监督学习的方法具有自动学习视频特征从而生成相应摘要的能力, 而不需要人工标注的视频摘要参与训练, 旨在将特定模型推广到各种不同领域. 早期的无监督方法多是基于聚类的, 比如 Hadi 等人[5]直接使用中心点聚类算法生成视频摘要, Avila 等人[6]提出的 VSUMM 模型先从视频中提取颜色特征, 然后执行中心点聚类算法来获取关键帧. 这些方法主要利用了底层的外观特征和运动信息, 虽然取得了较为良好的性能, 但不能有效地处理在摄像机运动较大、光照不足和场景复杂的情况下采集的视频.

近年来, 又有许多先进的无监督学习的方法被提出, 这些方法大致可以分为 4 类: 基于字典学习的方法、基于子集选择的方法、基于强化学习的方法、基于对抗学习的方法. 基于字典学习的方法将视频摘要定义为一个稀疏优化问题, 如 Elhamifar 等人[7]利用字典中的代表性元素对原视频进行了重构. 基于子集选择的方法通过选择能提供有用信息的视频帧子集来确定要保留的视频帧, 如 Elhamifar 等人[8]通过对比视频帧源集和目标集之间的相异性来从源集中选择样本子集, 使其能充分表达目标集. 基于强化学习的方法主要通过对视频进行离散采样来生成摘要, 如 Zhou 等人[9]提出了一种基于强化学习的深度摘要网络来进行视频摘要预测. 基于对抗学习的方法是为了克服缺乏视频摘要训练数据的问题而诞生的, 该方法能够学习训练数据中那些难以区分的视频摘要, 更全面高效地利用训练数据. Mahasseni 等人[10]提出了一种对抗长短期记忆 (long short-term memory, LSTM) 网络 SUM-GAN, 为之后逐渐涌现的各种对抗性视频摘要模型奠定了基础. Jung 等人[11]使用 SUM-GAN 作为基线, 提出了一种块与跨步网络 (chunk and stride network, CSNet), 解决了处理长视频输入时的梯度衰减问题, 以及由于每帧输出重要性分数的平坦分布而导致无效的特征学习的问题.

1.2 有监督视频摘要生成

有监督学习的方法通过训练分类器来学习帧或片段对于视频摘要的重要性, 由于可以利用人工标注的训练数据, 因此其效果往往优于无监督的方法. 早期的有监督学习的研究大多基于传统的机器学习方法, 这些方法通常都以视频分割作为第一步, 要么将视频均匀地分成相同大小的块, 要么使用核时域分割 (kernel temporal segmentation, KTS) 算法[12]来进行视频分割. 例如, Gygli 等人[13]结合底层时空特征开发了一种线性模型, 对不同特征进行加权求和来计算每个分割片段的兴趣得分; Song 等人[14]提出了训练因素分解模型来对视频帧的重要性进行预测; Potapov 等人[15]提出了一种基于支持向量机和 KTS 的视频帧分类方法.

随着深度学习的深入发展, 也诞生了许多基于深度学习的视频摘要生成方法. Zhang 等人[16]利用双向 LSTM 来预测视频帧的重要性分数, Zhao 等人[17,18]分别使用固定长度的分层循环神经网络 (recurrent neural network, RNN) 和分层结构自适应的 LSTM 来揭示视频的底层结构. 此外, 注意力模型也已经被引入到最近的视频摘要生成方法中, 并取得了良好的效果. 例如, Ji 等人[19]构建了一种基于注意力机制的编码器-解码器网络, Fajtl 等人[1]提出了一种利用自注意力机制的 VASNet 模型. 另外, Rochan 等人[2]提出的全卷积序列网络 (fully convolutional sequence networks, FCSN) 模型将卷积序列网络引入到视频摘要任务中. 以上有监督学习方法没有时序连续性的约束, 因此容易导致同一视频分段中的帧重要性分数不能准确表示该段在整体视频中重要性. 为了解决这一问题, Zhu 等人[3]提出了一种从检测到摘要的网络 (detect-to-summarize network, DSNet), 把视频摘要生成定义为一个时序检测过程, 预测视频中语义分段的时间坐标以及相应的重要性分数. 以上大多数已有的方法都使用预训练 CNN 提取视频帧的视觉特征, 忽略了其他不同类型的视觉特征 (比如运动特征、时序特征) 以及文本特征在重要性分数预测中可能起到的作用. 为了引入时序特征, 研究者们也做出了很多努力. 如 Yao 等人[20]提出了一种成对深度排序模型, 以同时结合视频的时域和空域信息, Zhang 等人[21]将视频摘要定义为序列到序列学习的问题, Huang 等人[22]构建了一种可以学习多阶段时空特征的深度网络模型.

由于视频摘要任务的复杂性, 即使代表了目前视频摘要技术发展水平的 DSNet, 其在 SumMe 和 TVSum 数据集上的实验结果也不够理想, 仍有很大的进步空间. 另外, 考虑到注意力机制和多模态融合在解决计算机视觉问题中的广泛应用和优良效果, 如视觉问答问题 (visual question answering, VQA)[23,24]和图像的自然语言描述问题[25,26]. 我们以 DSNet 模型框架为基础, 基于自注意力和共同注意力机制增加了多模态特征嵌入、变换和融合, 以此解决视频摘要生成问题.

2 基于空时变换网络的视频摘要生成模型

本文中我们提出了一种基于空时变换网络的视频摘要生成模型, 该模型主要包括嵌入层、特征变换与融合层和输出层 3 个模块. 本节将主要介绍模型整体框架以及各模块的实现细节.

2.1 问题定义及整体框架

视频摘要生成问题的输入是从原视频中采样获取的视频帧, 其可以用不同的视觉特征来表示, 本文中记为 $X=(x_0,x_1,...,x_t,...,x_T)$, $x_t\in\mathbb{R}^n$, $t\in 0,1,...,T$, x_t 是 t 时刻视频帧的视觉特征, 其维度为 n, T 为视频总时长, T+1 为视频帧的总数. 视频摘要生成模型的任务是给出每帧的重要性分数 $Y=(y_0,y_1,...,y_t,...,y_T)$. 进一步地, 我们通过每帧的重要性分数, 取出其中重要性分值高的若干帧, 便可以生成视频摘要.

本文提出的基于空时变换网络的视频摘要生成模型包括 3 大模块, 分别为: 嵌入层、特征变换与融合层、输出层. 该模型的整体框架如图 1 所示. 首先, 嵌入层为模型的输入层. 作为模型输入的视频帧视觉特征可以是深度卷积神经网络特征, 如 GoogLeNet[4], 也可以是基于内容的图像特征. 然而, 这些特征通常只包含视频帧的空间信息, 而忽略了运动和时序信息. 例如, 对于包含动作的视频, 不同的动作信息依赖于空间和时序的双重特征, 而不单单依赖于静态的图像内容和目标类别等空间特征. 基于以上考虑, 与以往的工作不同, 我们分别构建了空间网络和时序网络, 分别提取视频帧的空间特征和时序特征, 以增强视频帧的视觉信息表示. 进一步地, 空间特征和时序特征会同时送入特征变换与融合层. 该层主要包括空时变换网络和特征融合两部分, 其中, 空时变换网络的构建是基于自注意力机制, 主要完成视频帧的空间表示和时序表示, 特征融合主要完成空间表示和时序表示的融合. 最后, 融合特征被送入到输出层. 输出层经过时序提议生成、分类和回归, 最终生成视频摘要.

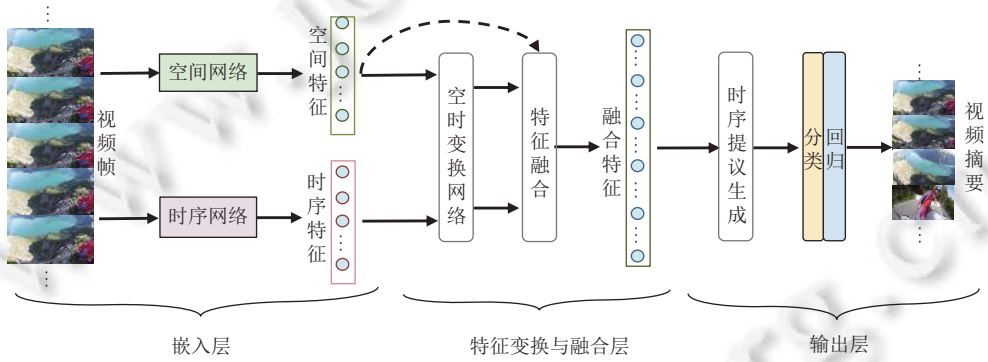


图 1 基于空时变换网络的视频摘要生成模型框架

概括地说, 模型所包含的 3 大模块的作用如下.

(1) 嵌入层: 基于空间网络和时序网络, 提取视频的空间特征和时序特征, 并作为下一模块的输入;

(2) 特征变换与融合层: 基于空时变换网络, 提取视频帧的空间表示和时序表示, 两者融合后获取融合特征;

(3) 输出层: 基于时序提议生成、分类和回归, 输出视频帧重要性分数, 并最终生成视频摘要.

2.2 基于空时变换网络的视频摘要生成

(1) 空间特征和时序特征提取

如图 2 所示, 为了提取视频帧的时序特征, 我们首先把彩色图像转化为灰度图像, 然后取当前 t 时刻的视频帧, 以及与其相邻的 $t-3$ 时刻和 $t+3$ 时刻的视频帧用于生成梯度图像. 随后, 原彩色图像输入到掩码区域卷积神经网络 (mask region-based convolutional neural network, Mask-RCNN)[27], 输出区域卷积特征. 在训练 Mask-RCNN 网络的过程中生成的区域提议会被同时送入两个卷积自编码器, 用于学习时序特征提取网络. 最后, 提取的空间和时序特征以及文本特征一同送入下一层的空时变换网络. 图 2 中, 我们用 V-BERT 和 L-BERT 代表空时变换网络, 该网络将在下一节做详细介绍.

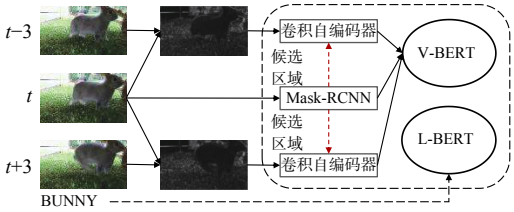


图 2 空间特征和时序特征提取框架

如上所述, 我们训练得到 3 个特征提取网络, 其中基于原视频帧训练的 Mask-RCNN 主要学习视觉特征, 基于梯度图像训练的卷积自编码器主要学习隐藏的运动特征. 这里 Mask-RCNN 网络我们选用先前用于 VQA 问题的网络模型[27], 该模型在 VQA 问题中表现了非常好的性能. Mask-RCNN 网络是快速区域卷积神经网络 (faster region-based convolutional neural network, Faster-RCNN)[28]的扩展网络, 其能够在有效检测目标的同时输出高质量的实例分割掩码. 与 Faster-RCNN 相比, 它用感兴趣区域 (region of interest, RoI) 对齐替换了 RoI 池化, 从而实现了像素级的对齐. 另外, 它并行添加了第 3 个分割掩码的分支, 从而实现了以像素到像素的方式预测分割掩码. 我们选用 Mask-RCNN 网络的原因还在于它具有很好的泛化能力, 能够和多种卷积神经网络框架结合.

Mask-RCNN 网络模型包括主干网络、区域候选网络 (region proposal network, RPN)、RoI 分类器、边界框回归器和分割掩码生成等模块. Mask-RCNN 网络模型的主干网络基于残差网络 ResNet101[29]和特征金字塔网络 (feature pyramid network, FPN)[30]. 它是在一个标准的 ResNet101 网络上扩展了 FPN, 其作为特征提取器把视频帧从 $1024\times1024\times3$ (RGB) 的张量转换成 $32\times32\times2048$ 的特征图. 该主干网络能够在多尺度上更好地表征目标, 实现高级和低级特征的相互结合. 主干网络学习到的特征作为输入被送到区域候选网络, 该网络是一个轻量级神经网络, 它用滑动窗来扫描锚区域, 以此寻找存在目标的区域. 使用 RPN, 模型可以选出最优的包含目标的锚区域, 并对其位置和尺寸实现精调. 随后, 获取的区域提议将被传递到 RoI 分类和边界框回归模块, 该模块主要实现分类和边界精调. 最终, RoI 分类器选择的正样本区域被输入到掩码卷积网络, 并生成 28×28 像素掩码.

我们构建的卷积自编码器同样是一个轻量级网络, 它的编码器只有 3 个卷积和最大池化模块, 解码器只有 3 个上采样和卷积模块, 并在解码器中附加一个卷积层作为输出层. 所有的卷积层都使用 3×3 滤波器, 除了解码器附加的卷积层, 其他卷积层后都加了线性整流函数 (rectified linear unit, ReLU)[31]完成激活. 编码器的前两个卷积层包含 32 个滤波器, 最后一层包含 16 个滤波器, 最大池化层是基于步长为 2 的 2×2 滤波. 在解码器中, 每个上采样层使用最近邻方法将输入激活值上采样两倍. 解码器中的第一个卷积层包含 16 个过滤器, 后面的两个卷积层各包含 32 个过滤器, 第 4 个 (也是最后一个) 卷积层包含 1 个过滤器. 我们采用自适应矩估计优化方法 (adaptive moment estimation, Adam)[32]对自编码器进行优化, 并选用像素均方误差作为损失函数, 其定义为:

$$Loss(I,I^*)=\frac{1}{g\times w}\sum_{j=1}^{g}\sum_{k=1}^{w}(I_{jk}-I_{jk}^*)^2 \tag{1}$$

其中, I 和 I^* 为输入和输出图像, 其像素大小为 $g\times w$.

进一步地, 由 Mask-RCNN 和卷积自编码器提取的空间特征和时序特征被送入空时特征变换网络进行特征变换.

(2) 空时特征变换与融合

为了实现空时特征变换, 我们引入了基于注意力机制的变换器, 该变换器的整个网络结构均使用注意力机制组成. 受视觉-语言双向编码变换器 (vision-and-language bidirectional encoder representation from transformers, ViLBERT)[33,34]的启发, 我们将空间特征、时序特征和文本特征作为模型的输入, 引导模型学习更加丰富的视觉和自然语言的无任务偏好的联合表示. 本文中, 我们把 ViLBERT 看做一种通用的视觉基础模型, 迁移学习其视觉信息表示方法到视频摘要生成问题上. ViLBERT 模型从 Conceptual Captions 数据集[35]中学习到了和任务无关的视觉基础知识, 我们把该模型迁移到我们的视频摘要生成模型中, 通过精调使其适应我们的视频摘要生成任务.

ViLBERT 模型的实现框架为一个图像和文本的双流结构, 即视觉流 V-BERT 和自然语言流 L-BERT, 如图 3 所示. 该双流架构对图像和文本两种模态分别进行建模, 可以实现对各个模态采用可变的网络深度 (如视觉流采

用 G 层, 自然语言流采用 G*层), 并通过自注意力和共同注意力模块实现跨模态的交互和链接. ViLBERT 模型应用于计算机视觉领域的 VQA、基于内容的图像检索和视觉常识推理等任务, 均取得了令人满意的效果. 同时考虑到多模态特征的重要性[36,37], 本文中, 我们把 ViLBERT 模型学习到的视觉基础迁移到视频摘要生成任务中, 并把两种模态的嵌入扩展为 3 种, 即增加了时序特征的嵌入.

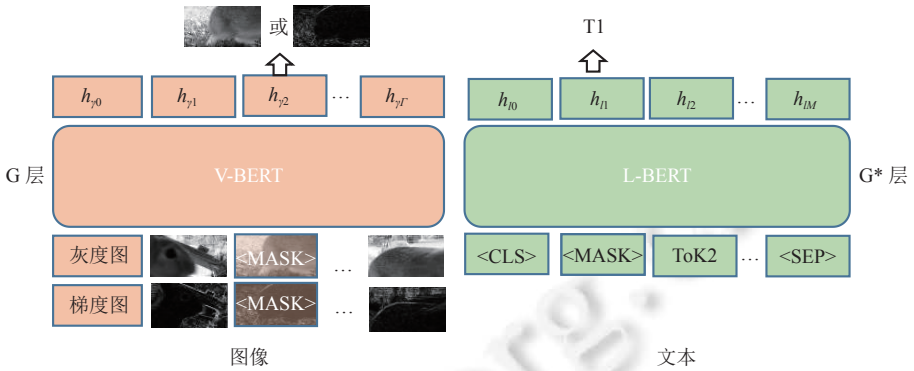


图 3 空时 ViLBERT 实现框架

如图 3 所示. 给定图像区域特征 $R=(\gamma_0,\gamma_1,...,\gamma_\tau,...,\gamma_\Gamma)$,$\tau\in 0,1,...,\Gamma$, $\Gamma+1$ 为区域特征的个数, γ_τ 为第 τ 个区域特征, 和文本输入 $L=(l_0,l_1,...,l_m,...,l_M), m\in 0,1,...,M$, $M+1$ 为文本词汇的个数, l_m 为第 m 个文本词汇, 该模型输出表示为 $H_R=(h_{\gamma 0},h_{\gamma 1},...,h_{\gamma\tau},...,h_{\gamma\Gamma})$ 和 $H_L=(h_{l0},h_{l1},...,h_{lm},...,h_{lM})$. 需要说明的是, 这里的图像区域特征包含空间区域特征 R_{spl} 和时序区域特征 R_{tmp} , 两种特征分两次单独输入到 ViLBERT 模型. 最终, ViLBERT 模型两次输出向量 $H_{R_{\text{spl}}}$ 和 $H_{R_{\text{tmp}}}$ 相加, 作为最终的融合特征 H_R , 即 $H_R=H_{R_{\text{spl}}}+H_{R_{\text{tmp}}}$.

● 空时变换网络构建方法

空时变换网络延用 ViLBERT 模型中的双流结构, 由若干自注意力变换层和共同注意力变换层组成. 如图 4 所示, 视觉流输入为上一阶段提取的空间特征和时序特征, 自然语言流输入为本文特征, 双流结构通过共同注意力变换层实现信息的交互. 其中, 虚线框标记的区域为模型中重复使用的模块.

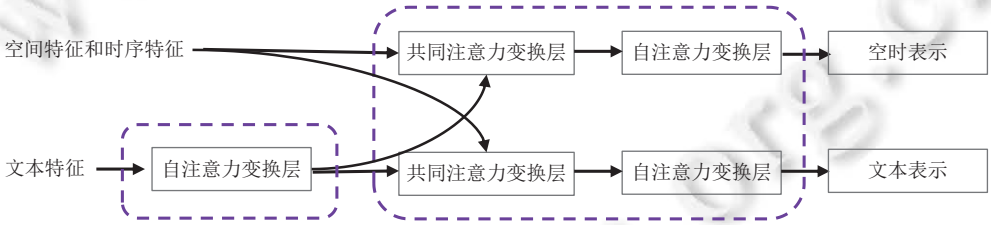


图 4 空时特征变换网络架构

自注意力变换层包括多头注意力、相加和归一化、前馈 3 部分, 其结构和双向编码变换器 (bidirectional encoder representation from transformers, BERT)[38]的结构相同. 变换器的本质是自编码器, 在编码阶段, 嵌入向量经过多头注意力模块得到加权特征向量 Z. 多头注意力模块是多个自注意力模块的融合, 一个自注意力模块的输出可以用公式 (2) 表示为:

$$Z:\ Attention(Q,K,V)=Softmax\left(QK^{\mathrm{T}}/\sqrt{d_k}\right)V \tag{2}$$

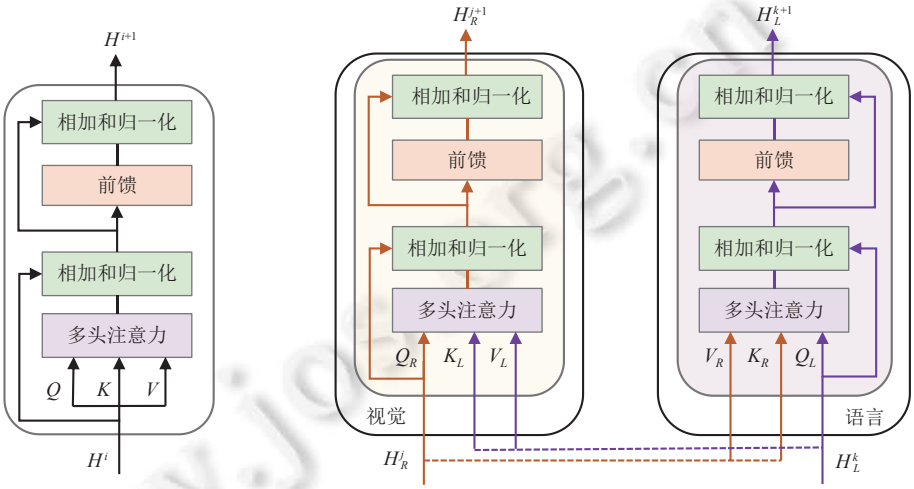
其中, $Q\in\mathbb{R}^{n\times d_k}$ 为查询词, $K\in\mathbb{R}^{n\times d_k}$ 为模型根据 Q 匹配的键, $V\in\mathbb{R}^{n\times d_v}$ 为根据 Q 和 K 的相似度得到的匹配内容, d_k,d_v 分别为 K 和 V 的维度. 这里 QK^{T} 计算得到的便是相似度分数, 并用 $\sqrt{d_k}$ 做归一化. 其中 Q,K,V 的计算方法为:

$$\begin{cases}\Lambda\times W^Q=Q\\ \Lambda\times W^K=K\\ \Lambda\times W^V=V\end{cases} \tag{3}$$

其中, $\Lambda \in \mathbb{R}^{n\times d_{\text{model}}}$ 为嵌入向量, $W^Q \in \mathbb{R}^{d_{\text{model}}\times d_k}$ 、 $W^K \in \mathbb{R}^{d_{\text{model}}\times d_k}$ 和 $W^V \in \mathbb{R}^{d_{\text{model}}\times d_v}$ 是计算 Q, K, V 所需要的权重, 即模型需要学习的权重参数, d_{model} 为模型的维度. 嵌入向量分别输入到多个自注意力模块, 即多头注意力, 如图 5 所示. 多头注意力是把 Q, K, V 通过不共享的参数矩阵进行线性变换, 然后执行自注意力机制, 并重复多次. 由此, 多头注意力可定义为:

$$\begin{cases} head_i = Attention(QW_i^Q, KW_i^K, VW_i^V) \\ Multi\_head(Q,K,V) = Concat(head_1,...,head_i,...,head_\hbar) \end{cases} \tag{4}$$

其中, $i \in 1,2,...,\hbar$, $\hbar$ 为多头注意力模块中头的个数. 多个自注意力并行处理后, 结果按列拼接即为多头注意力的输出.



(a) 自注意力变换层 (b) 共同注意力变换层

图 5 自注意力变换层和共同注意力变换层结构图

进一步地, 多头注意力输出的特征矩阵与原嵌入向量完成残差相加并归一化, 然后送入前馈网络. 注意力变换层中的前馈网络为简单的全连接网络 (position-wise feed-forward networks, FFN), 其对每个位置的向量 $\chi \in \mathbb{R}^n$ 分别做两次线性变换和一次 ReLU 激活, 定义如下:

$$FFN(\chi) = \max(0, \chi W_1^{FFN} + b_1)W_2^{FFN} + b_2 \tag{5}$$

其中, (W_1^{FFN}, b_1) 和 (W_2^{FFN}, b_2) 为线性变换的权重和偏置, 经过一层 FFN 后输出最终的加权特征向量 Z. 如图 5(a) 所示, 加权特征向量 Z 再重复与初始化表征的残差相加和归一化. 其输出作为下一个注意力层的输入. 给定中间层视觉和自然语言表示 H_R 和 H_L, 共同注意力变换层同样计算 Q, K, V 矩阵, 但该模块将每个模态的 K 和 V 输入到另外一个模态的多头注意力模块, 如图 5(b) 所示. 由此, 与原自注意力变换层相比, 共同注意力变换层增加了视觉流和自然语言流的交互, 从而体现了视觉流中图像视觉条件下的自然语言注意力和自然语言流中语言条件下的图像注意力.

● 嵌入向量表示方法

对于文本, 模型对由文本词汇和一小组特殊标记 SEP、CLS 和 MASK 组成的离散标记序列进行操作. 对于给定的标记, 输入表示是特定于标记的学习嵌入 (即词向量[39]等)、位置编码 (即序列中标记的索引) 和片段 (即标记所在句子的索引) 的总和.

对于图像, 我们利用上一步预训练的空间和时序网络提取边界框及其视觉特征来生成图像区域特征. 与文本中的单词不同, 图像区域缺乏自然排序. 为了表示图像区域的位置信息, 该模型中使用一个 5 维的向量对图像区域位置进行了编码. 这 5 维向量的组成包括归一化后图像区域边界框的左上角和右下角的坐标, 以及图像区域的覆盖占比, 其维数通过映射后与视觉特征维度保持一致. 此外, 我们用特殊标记<IMG> 作为图像区域序列的起始, 并取<IMG> 最后的输出表征整个图像 (如图 3 所示).

● 模型预训练

与 ViLBERT 相同, 我们考虑两个预训练任务: 掩码多模态建模和多模态对齐预测. 掩码多模态建模任务 (如图 3 所示) 遵循标准 BERT 中的掩码语言建模任务, 掩码大约 15% 的单词和图像区域输入, 并在给定剩余输入的情况下对模型进行重构. 掩码语言建模任务随机将输入词汇分为掩码词汇 X_{MASK} 和观察到的词汇 X_{UNMASK} (X_{MASK} 与 X_{UNMASK} 无交集). 80% 的情况下, 掩码词汇会被特殊的 <MASK> 标记替换, 另 10% 会被随机单词替换, 剩下的 10% 不做更改, 然后训练模型在给定观察集的情况下重建这些掩码词汇. 掩码图像区域的图像特征在 90% 的情况下被归零, 10% 的情况下保持原特征不变. 掩码图像建模任务不是直接回归掩码特征值, 而是预测相应图像区域的语义类别. 为了监督这一点, 我们从特征提取中使用的相同预训练检测模型中获取该区域的输出分布. 我们的模型训练通过最小化这两个分布之间的 K-L 散度来实现.

在多模态对齐任务中, 模型以图像-文本对表示为 $\{\mathrm{IMG},\gamma_0,\gamma_1,...,\gamma_\Gamma,\mathrm{CLS},l_0,l_1,...,l_M,\mathrm{SEP}\}$, 并且预测图像和文本是否对齐, 即文本是否描述了图像. 我们将输出 h_{IMG} 和 h_{CLS} 作为视觉和自然语言输入的整体表示, h_{IMG} 和 h_{CLS} 元素点乘后, 作为最终的融合表示. 然后, 学习一个线性层来进行二值预测图像和文本是否对齐.

概括地说, 我们要完成的任务分别为: 1) 给定输入, 预测被遮蔽的文本词汇和图像区域的语义; 2) 预测图像和文本是否语义对齐. 与原 ViLBERT 模型只输入图像空间特征不同, 我们同时输入图像的空间表示和时序表示, 以预测被遮蔽的图像区域的语义, 以及图像和文本是否对齐.

需要特别说明的是, 本文中我们把 ViLBERT 看做一种通用的视觉基础模型, 目的是在视频摘要生成任务中更好地表征视频帧特征. 因此, 我们使用了 Lu 等人[33]在 12 个图像数据集上预训练的通用模型, 然后遵循原作者在下游任务上的微调方法, 对通用模型进行了微调以适应我们的任务. 其中, 文本词汇来源是 Conceptual Captions 数据集[35]和视频帧的类别标签.

(3) 视频摘要生成

视频摘要生成阶段主要包括段预测和关键镜头选择两部分工作. 通过上一阶段的工作, 我们获得了视频帧最终的特征表示, 接下来我们需要预测每个视频帧的重要性分数、段边界和中心度分数, 而且每帧重要性分数需要被转换为镜头级重要性分数. 需要说明的是, 真实的视频摘要是基于生成的分段 $C=\{(t_c^b,t_c^e,s_c)\}_1^{N_C}$ 通过 KTS[12]和 0/1 背包算法[3]创建的. 其中, t_c^b、t_c^e 和 s_c 分别表示第 c 个片段的开始时间、结束时间和重要性分数, N_C 是视频摘要中的分段数.

● 分段预测

分段预测模块由 3 个分支组成, 分别为重要性分类分支、位置回归分支和中心度回归分支, 旨在学习重要性分类和分段位置. 给定视频帧, 该模块需要学习重要性分数, 位置边界和中心度分数. 具体来说, 在训练阶段, 当第 t 帧在真实视频摘要中被选中时, 则该帧被视为正样本. 否则, 归为负样本. 由于每一帧只属于某个特定的片段, 因此视频帧的标签分配是明确的.

此外, 对于每个正样本, 学习一个真实的边界向量 $\Delta_{t^*}=(\Delta_{l*},\Delta_{r*})$, 其中 Δ_{l*} 和 Δ_{r*} 是当前位置 t^* 与片段 c 左右边界 t_c^b 和 t_c^e 之间的间隔, 定义为:

$$\begin{cases}\Delta_{l*}=t^*-t_c^b\\ \Delta_{r*}=t_c^e-t^*\end{cases} \tag{6}$$

我们应用焦点损失 $Loss_{\mathrm{cls}}$[40]做重要性分类, 它通过降低分类良好的样本的损失来处理类不平衡问题, 并且我们利用时间交并比 (temporal intersection over union, tIoU) 损失 $Loss_{\mathrm{reg}}$[3]进行位置回归, 其对不同间隔的时序兴趣具有鲁棒性. 于是, 训练损失被定义为:

$$Loss=\frac{1}{N_p}\sum_t Loss_{\mathrm{cls}}(s_t,s_t^*)+\frac{\lambda}{N_p}\sum_k Loss_{\mathrm{reg}}(\Delta_k,\Delta_k^*) \tag{7}$$

其中, s_t 和 s_t^* 是第 t 个预测的和真实的帧级重要性分数, Δ_k 和 Δ_k^* 分别表示第 k 个正样本预测的和真实的位置, N_p 为正样本的个数, λ 用作平衡分类和回归损失.

由于许多正样本的时间位置靠近相应真实值的边界, 因此我们的方法将生成许多低质量片段. 为了解决这个问题, 我们在位置回归和重要性分类的基础上, 增加了中心度约束, 利用中心度约束来确保时间位置靠近预测片段

的中心. 真实中心度得分定义为:

$$s^{\text{center}} = \frac{\min(\Delta_{l*}, \Delta_{r*})}{\max(\Delta_{l*}, \Delta_{r*})} \tag{8}$$

我们利用具有平衡权重 μ 的二元交叉熵 (binary cross entropy, BCE) 损失 $Loss_{\text{center}}$ [3]来获得中心度分数. 最后, 我们将 3 个损失相加, 则最终的多任务损失函数为:

$$\begin{aligned} Loss* &= Loss + \frac{\mu}{N_p}\sum_k Loss_{\text{center}}(s_k^{\text{center}}, s_k^{\text{center}*}) \\ &= \frac{1}{N_p}\sum_t Loss_{\text{cls}}(s_t, s_t^*) + \frac{\lambda}{N_p}\sum_k Loss_{\text{reg}}(\Delta_k, \Delta_k^*) + \frac{\mu}{N_p}\sum_k Loss_{\text{center}}(s_k^{\text{center}}, s_k^{\text{center}*}) \end{aligned} \tag{9}$$

其中, s_k^{center} 和 $s_k^{\text{center}*}$ 分别表示第 k 个正样本预测的和真实的中心度分数.

分段预测模块包含一个共享的全连接层和 3 个独立的分支, 分别用于帧重要性分类、位置回归和中心度回归. 其中, 中心度分支的结构与回归分支的结构相同. 在训练阶段, 我们通过使用公式 (9) 给出的多任务损失函数来优化模型的参数.

● 关键镜头选择

为了生成视频摘要, 我们需要将视频序列分割成镜头并估计镜头级别的重要性分数. 首先, 我们遵循之前的工作[3], 应用 KTS[12]镜头检测方法将视频序列分割成视频镜头.

在测试阶段, 通过使用训练模型, 我们可以获得每个时间位置 t 的重要性分数 s_t 、位置预测 Δ_l 和 Δ_r 以及中心度分数 s_t^{center} . 随后, 我们计算每个预测段的开始和结束时间 t_c^b 和 t_c^e , 计算方法如下:

$$\begin{cases} t_c^b = t - \Delta_l \\ t_c^e = t + \Delta_r \end{cases} \tag{10}$$

其中, t 表示视频帧的时间索引. 该预测段的重要性分数定义为 $s_c = s_t \times s_t^{\text{center}}$, 这表明一个好的片段应该具有很高的重要性分数, 同时相关帧应该在段的中心部分.

另外, 由于预测片段的高重叠率和低置信度, 我们通过非最大抑制算法 (non-maximum suppression, NMS) [41]过滤掉冗余和低质量的片段. 然后, 将第 t 个时间位置的预测片段的最大值指定为第 t 个帧级重要性得分. 一旦获得帧级重要性分数, 我们通过平均同一镜头内的帧级重要性分数来计算镜头级重要性分数, 计算方法如下:

$$y_o = \frac{1}{N_o}\sum_{i=1}^{No} s_{o,i} \tag{11}$$

其中, N_o 是第 o 个镜头的长度, $s_{o,i}$ 是第 o 个镜头中第 i 帧的重要性分数. 为了和已有方法公平比较, 我们遵循视频摘要中帧的数目占总视频帧 15% 的约束[3]. 最后, 应用 0/1 背包算法来选择视频镜头, 定义如下:

$$\max \sum_{o=1}^{O} u_o y_o, \qquad \text{s.t.} \sum_{o=1}^{O} u_o y_o \leqslant 15\% \times T \tag{12}$$

其中, $u_o \in \{0,1\}$ 表示是否在摘要中选择第 o 个镜头, O 为镜头数, T 为视频长度. 我们应用动态规划方法来解决这个最大化问题. 通过选择 $u_o = 1$ 的镜头来生成最终摘要.

3 实验及分析

3.1 数据集介绍

我们在 SumMe 数据集[13]和 TVSum 数据集[14]上评估了模型的性能. SumMe 数据集共包含 25 个视频序列, 涵盖假期、烹饪和运动等各种类型. 该数据集来自于用户自己录制的视频, 每个视频有 15–18 个用户标注, 时长最短 32 s, 最长 324 s, 平均时长 146 s. TVSum 数据集由从 YouTube 下载的 50 个视频序列组成, 其包含各种类型的视频 (例如新闻、指南、游行等). 该数据集中每个视频包含 20 个用户标注, 并提供了每个视频的标题和类别, 时长最短 83 s, 最长 647 s, 平均时长 235 s. 两个数据集都提供了多个用户标注的视频摘要. 具体来说, 我们遵循以前的工作并将最初以 30 fps 捕获的所有视频下采样到 2 fps, 以处理时间冗余并减少计算. 我们还使用了另外两个数

据集, 即 OVP [6]和 YouTube [6], 来扩充训练数据集. OVP 数据集有 50 个视频序列, YouTube 数据集包含 39 个视频序列 (除去了卡通视频). 这两个数据集最短时长均为 83 s, 最长时长均为 647 s, 平均时长 235 s.

对于位置回归, 需要真实分段标签. 然而, SumMe 和 TVSum 数据集都只提供了帧级重要性分数. 因此, 我们遵循传统方法并应用 KTS 将视频分割成几个镜头, 其中镜头级别的重要性分数由公式 (11) 计算. 然后, 应用 0/1 背包算法来生成基于关键镜头的摘要. 当一个片段包含关键帧时, 我们首先将它们的关键帧转换为关键镜头候选, 然后应用 0/1 背包算法来满足原始视频长度的 15% 的约束.

我们利用 3 种评估设置来评估所提出模型的性能, 即规范、增强和转移设置. 对于规范和增强设置, 我们将数据集随机分为 5 个部分. 在规范设置中, 80% 的数据集用于训练, 其余 20% 用于评估. 在增强设置中, 使用另外 3 个数据集增强的数据集的 80% 用于训练. 在转移设置中, 3 个数据集用于训练, 剩下的一个数据集用于评估. 在实验中, 我们设定规范设置为默认设置. 我们对每个设置运行我们的模型 5 次, 并报告这 5 次运行的平均性能.

3.2 评估指标

我们利用 F_β-测量来评估模型生成的摘要. 若生成的摘要表示为 gs, 用户创建的摘要表示为 gt, 则精度 p 和召回率 r 计算方法分别为:

$$p = \frac{length(gs \cap gt)}{length(gs)} \tag{13}$$

$$r = \frac{length(gs \cap gt)}{length(gt)} \tag{14}$$

具体地, F_β-测量计算方法为:

$$F_\beta = \frac{(1+\beta^2)\times p \times r}{(\beta^2 \times p) + r} \tag{15}$$

在我们的实验中, 我们采用调和平均 F_1 测量 ($\beta = 1$) 作为默认 F-分数计算方法. 遵循 SumMe 和 TVSum 的评估协议 [42], 对于每一个视频, 我们通过计算模型生成的摘要与多个用户创建的摘要之间的 F-分数来评估预测摘要的质量.

3.3 实验结果及分析

我们从 Mask-RCNN 中提取了 2 048 维特征, 多头注意力层中的头部数设置为 8, 从特征变换与融合层输出的特征表示为 1 024 维. 公式 (9) 中平衡参数 λ 和 μ 设置为 1. 对于焦点损失, α 和 γ 分别设置为 0.25 和 2. 此外, 我们使用 Adam 优化方法对模型训练迭代了 300 次, 学习率为 5×10^{-5}, 权重衰减为 1×10^{-5}. NMS 阈值设置为 0.3. 我们的模型除了嵌入视频帧特征还嵌入文本特征. 对于 SumMe 数据集, 我们取视频名称为嵌入文本, 如“Base jumping”. 对于 TVSum 数据集, 我们取视频的标题为嵌入文本.

(1) 与现有先进方法的比较

为了验证所提出模型的有效性, 我们在 3 个数据集设置 (包括规范设置、增强设置和迁移设置) 下将我们的方法与最先进的方法进行了比较. 表 1 展示了在 SumMe 和 TVSum 数据集上使用不同视频摘要方法的实验结果. 为了表述方便, 我们把本文中提出的空时变换网络模型 (spacial-temporal transform network, STTN) 简写为 STTN. 从表中我们可以观察到, 在规范设置下, 在 SumMe 数据集上, 我们的 F-分数达到 52.9%, 与已有的最优的方法比较提升了 1.7%; 在 TVSum 数据集上, 我们的 F-分数达到 63.3%, 与已有的最优的方法比较提升了 1.4%. 在增强设置下, 我们的模型也取得了最好的性能, 在 SumMe 数据集上实现了 1.3% 的效果提升, 在 TVSum 数据集上实现了 1.6% 的效果提升. 此外, 在增强设置下, 我们模型的性能均优于规范设置下. 由此可见, 数据增强通过数据的补充确实能够提升模型的性能. 在迁移设置下, 除了 TVSum 数据集上的 DR-DSN[9]之外, 我们的方法在两个数据集上都优于其他最先进的方法.

STTN 模型是无锚算法, 我们保持原有模型的嵌入层、特征变换与融合层不变, 替换输出层为 DSNet 模型中的基于锚的提议生成和分类回归, 由此获得基于锚的模型. 为了公平比较, 基于锚的模型中, 附加的参数均和 DSNet 模型中设定参数一致[3]. 表 2 中列出了基于锚的方法和无锚方法的 F-分数. 从表 2 可以看出, 基于锚的方法在 TVSum 数据集上的结果要优于原模型. 原因可能是 TVSum 数据集的真实片段的持续时间往往比 SumMe 数据

集的长, 并且基于锚的方法更容易处理长片段. 我们基于锚的方法在两个数据集上的 F-分数均高于基于锚的 DSNet 方法, 由此也说明了我们模型所提取的空时变换特征的有效性.

表 1 在 SumMe 和 TVSum 数据集上与现有先进方法的 F-分数比较 (%)

模型	SumMe			TVSum		
	C	A	T	C	A	T
vsLSTM[16]	37.6	41.6	40.7	54.2	57.9	56.9
dppLSTM[16]	38.6	42.9	41.8	54.7	59.6	58.7
SUM-GAN[10]	41.7	43.6	—	56.3	61.2	—
DR-DSN[9]	42.1	43.9	42.6	58.1	59.8	**58.9**
A-AVS[19]	43.9	44.6	—	59.4	60.8	—
M-AVS[19]	44.4	46.1	—	61.0	61.8	—
FCSN[2]	48.8	50.2	45.0	58.4	59.1	57.4
DSNet[3]	51.2	53.3	47.6	61.9	62.2	58.0
STTN	**52.9**	**54.6**	**48.9**	**63.3**	**63.8**	58.5

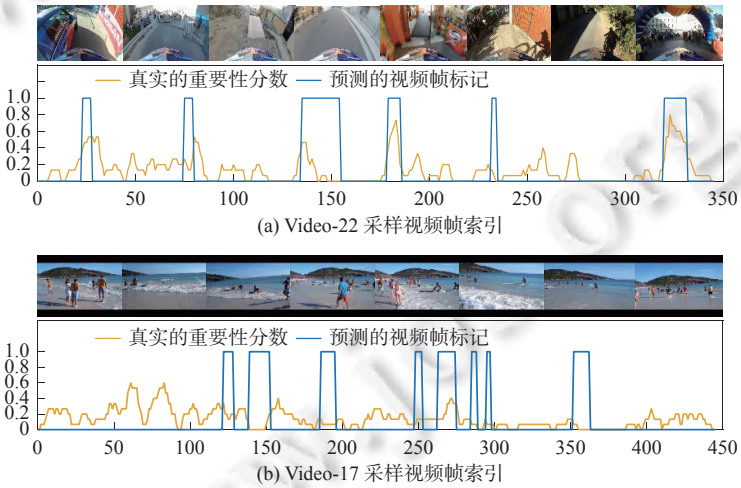
注: C表示规范设置下的结果, A表示增强设置下的结果, T表示迁移设置下的结果.

表 2 基于锚的方法在 SumMe 和 TVSum 数据集上的 F-分数比较 (%)

模型	SumMe			TVSum		
	C	A	T	C	A	T
DSNet[3]	51.2	53.3	47.6	61.9	62.2	58.0
基于锚的DSNet[3]	50.2	50.7	46.5	62.1	63.9	59.4
STTN	**52.9**	**54.6**	**48.9**	63.3	63.8	58.5
基于锚的STTN	51.5	52.8	47.2	**63.5**	**64.1**	**59.9**

注: C表示规范设置下的结果, A表示增强设置下的结果, T表示迁移设置下的结果.

图 6 展示了视频摘要生成实例, 黄色曲线为用户标记的真实的帧级重要性分数, 蓝色曲线为我们模型预测的视频帧标记, 若某帧被选为关键帧则标记为 1, 未选中则标记为 0. 其中, 图 6(a) 为 Video_22 “Valparaiso_Downhill” 的视频摘要生成情况, 从图中可以看出, 我们的算法几乎捕捉到了全部真实的关键镜头, 效果较好; 图 6(b) 为 Video_17 “Saving dolphines”的视频摘要生成情况, 从图中可以看出, 我们的算法遗漏了大部分关键镜头, 效果较差. 对于 Video_17, 效果较差的原因主要为该视频背景 (海滩) 和事件 (拯救海豚) 均比较单一和平稳, 对于标注者来说该视频每帧的区分性不强, 容易审美疲劳, 所以也导致了真实的重要性分数在刚开始有两段是高于 0.5 的, 后面均低于 0.4. 对于模型来说, 无法很好地适应这种因审美疲劳引发的标注误差.



(a) Video-22 采样视频帧索引

(b) Video-17 采样视频帧索引

图 6 视频摘要生成效果较好和较差的实例图

(2) 参数分析

为了评估参数的敏感性, 我们使用不同的参数值训练和测试模型.

● 参数 λ 和 μ 分析

我们采用网格搜索策略, 以 0.25 的间隔在 [0.25, 2.00] 范围内采样参数 λ 和 μ. 因此, 参数 λ 和 μ 的取值集均为

{0.25, 0.5, 0.75, 1.00, 1.25, 1.50, 1.75, 2.00}. 实验是通过组装这些值的每一种可能的组合而形成的, 因此总共有 64 次实验. 图 7 给出了 SumMe 数据集上不同 λ 和 μ 的可视化实验结果, 由此可以观察到我们的方法对范围内的参数 λ 和 μ 不敏感. 为简单起见, 我们在实验中都将 λ 和 μ 值设置为 1.00.

● NMS 阈值研究

由于 NMS 过滤掉了低质量和冗余的片段并且对最终结果非常重要, 我们进一步进行了 NMS 阈值分析. 我们同样以 SumMe 数据集为例, 图 8 显示了使用不同 NMS 阈值的实验结果. 从图 8 我们可以看出, 当 NMS 阈值为 0.3 时, 我们的方法实现了最佳性能. 因此, 我们在实验中将 NMS 阈值的默认值设置为 0.3.

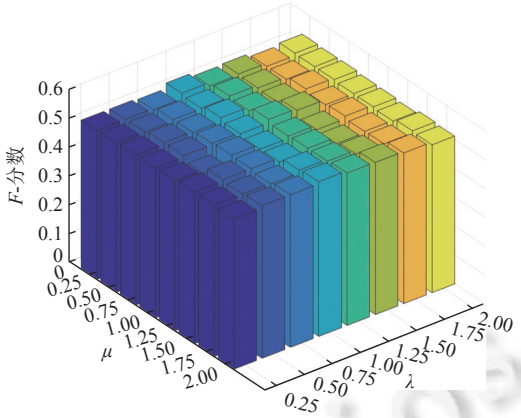


图 7 在 SumMe 数据集上不同 λ 和 μ 取值下的 F-分数可视化图

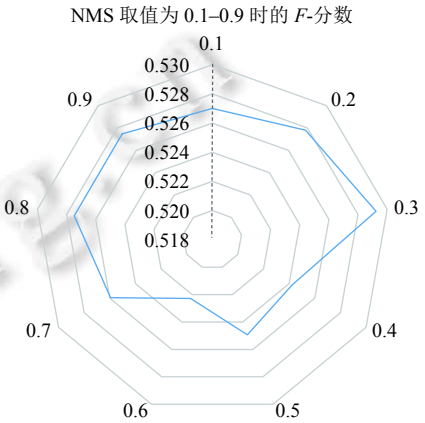


图 8 NMS 阈值分析

● 多样性分析

多样性是视频摘要的一个关键属性. 我们遵循文献 [16] 中的实验设置, 并使用其多样性度量来评估 SumMe 和 TVSum 数据集上生成的摘要的多样性. 更多样化的摘要对应于更高的多样性得分. 表 3 显示了使用不同方法的多样性得分, 其中 dppLSTM 和 DR-DSN 都利用了多样性约束. 从表 3 可以观察到, 与 dppLSTM 和 DR-DSN 相比, 我们的方法获得了更高的分数.

表 3 在 SumMe 和 TVSum 数据集上的多样性分数

数据集	dppLSTM[16]	DR-DSN[9]	DSNet[3]	**STTN**
SumMe	0.591	0.594	0.664	**0.718**
TVSum	0.463	0.464	0.477	**0.592**

3.4 消融实验

为了更好地说明我们模型各个模块的有效性, 我们设计了 4 个消融实验.

a) 去掉空时变换网络中的文本特征嵌入, 表示为 STTN-txt;

b) 去掉空时变换网络中的时序特征嵌入, 表示为 STTN-tmp;

c) 去掉空时变换网络中的注意力层, 表示为 STTN-atn;

d) 在最终的融合特征中加入 GoogLeNet 特征, 表示为 STTN^{+}.

需要说明的是, 去掉空时变换网络中的文本特征嵌入后, STTN-txt 模型仍然保留了自注意力机制. 然而, 去掉空时变换网络中的注意力层的 STTN-atn 模型用 LSTM 模型替换了原有注意力层.

表 4 给出了规范设置下消融实验的结果, 如表 4 前 4 行结果所示, 其中不管是去掉空时变换网络中的文本特征嵌入、时序特征嵌入还是注意力层均降低了模型的性能, 因此文本特征、时序特征和注意力机制均对性能的提升起到了必不可少的作用. 另外, 去掉注意力层后, 模型的性能降低最明显, 说明了注意力机制的重要性. 消融实验

模型 STTN-txt 去掉了文本特征但保留了自注意力机制, 与同样使用自注意力机制的 DSNet 模型相比, 其 *F*-分数提高了 1.2%, 从另外一个角度说明了时序嵌入和模型构建的有效性. 考虑现有的先进方法均采用 GoogLeNet 特征作为视频帧的视觉特征, 如 DSNet 模型在提取的长程特征的基础上直接加上 GoogLeNet 特征作为最终的视频帧表示. 为了说明我们特征提取的有效性, 我们在现有模型提取的融合特征基础上也直接加上 GoogLeNet 特征作为最终的特征表示, 以此设定了消融实验模型 $STTN^+$. 从表 4 的结果可以看出, $STTN^+$与 STTN 相比只有 0.2%–0.3% 的提升, 可见我们模型的融合特征提取策略还是非常有效的.

表 4 在 SumMe 和 TVSum 数据集上的消融实验结果 (*F*-分数) (%)

消融模型	SumMe	TVSum
STTN-txt	52.4	62.1
STTN-tmp	52.5	62.8
STTN-atn	51.0	61.6
$STTN^+$	**53.1**	**63.6**
$STTN\text{-}grad^1$	**52.8**	**63.4**
$STTN\text{-}grad^2$	**53.0**	**63.2**
$STTN\text{-}grad^4$	**52.8**	**63.3**

注: STTN-txt去掉了空时变换网络中的文本特征嵌入, STTN-tmp去掉了空时变换网络中的时序特征嵌入, STTN-atn去掉了空时变换网络中的注意力层, $STTN^+$在最终的融合特征中加入了GoogLeNet特征.

另外, 在提取时序特征时, 我们只是简单的选取了当前 t 时刻的视频帧, 以及与其相邻的 $t-3$ 时刻和 $t+3$ 时刻的视频帧用于生成梯度图像. 为了说明我们的模型对帧间时间距离的不敏感性, 我们设计了 3 个消融实验, 分别为:

a) 提取 $t-1$, t, $t+1$ 时刻的视频帧用于生成梯度图像, 表示为 $STTN\text{-}grad^1$;

b) 提取 $t-2$, t, $t+2$ 时刻的视频帧用于生成梯度图像, 表示为 $STTN\text{-}grad^2$;

c) 提取 $t-4$, t, $t+4$ 时刻的视频帧用于生成梯度图像, 表示为 $STTN\text{-}grad^4$.

如表 4 后 3 行结果所示, $STTN\text{-}grad^1$、$STTN\text{-}grad^2$、$STTN\text{-}grad^4$ 和 STTN 相比, 其结果仅有 0.1%–0.2% 的波动, 因此我们的模型对时序特征提取时的帧间时间距离是鲁棒的. 分析其原因主要为: 1) 我们的任务不需要实现目标的追踪, 故帧之间的时间距离并不重要; 2) SumMe 和 TVSum 数据集本身是场景连续的视频源, 在梯度计算所需合理的时间间隔内均对结果影响较小.

4 结 论

视频摘要生成可以应用于自动媒体生成技术中, 并可以广泛应用于媒体信息的浏览、检索和推广等, 其研究具有重要的理论意义和实用价值. 本文提出了一种基于空时变换网络的视频摘要生成模型, 该模型主要关注于视频特征的有效提取. 空时变换网络基于自注意力和共同注意力机制, 实现了多模态特征的嵌入、变换和融合, 使得视频表示更加准确和完备. 考虑到媒体信息的多样性, 多模态特征的应用使得我们的模型可以更好地应用到自动媒体生成技术中. 目前, 该模型的工作集中于空时变换和融合阶段. 在未来的工作中, 我们将关注视频摘要生成阶段, 研究如何设计更加合理的分类和回归方案.

References:

[1] Fajtl J, Sokeh HS, Argyriou V, Monekosso D, Remagnino P. Summarizing videos with attention. In: Proc. of the 14th Asian Conf. on Computer Vision). Perth: Springer, 2019. 39–54. [doi: 10.1007/978-3-030-21074-8_4]

[2] Rochan M, Ye LW, Wang Y. Video summarization using fully convolutional sequence networks. In: Proc. of the 15th European Conf. on Computer Vision. Munich: Springer, 2018. 358–374. [doi: 10.1007/978-3-030-01258-8_22]

[3] Zhu WC, Lu JW, Li JH, Zhou J. DSNet: A flexible detect-to-summarize network for video summarization. IEEE Trans. on Image Processing, 2021, 30: 948–962. [doi: 10.1109/TIP.2020.3039886]

[4] Ghauri JA, Hakimov S, Ewerth R. Supervised video summarization via multiple feature sets with parallel attention. In: Proc. of the 2021 IEEE Int'l Conf. on Multimedia and Expo. Shenzhen: IEEE, 2021. 1–6s. [doi: 10.1109/ICME51207.2021.9428318]

[5] Hadi Y, Essannouni F, Thami ROH. Video summarization by k-medoid clustering. In: Proc. of the 2006 ACM Symp. on Applied Computing. Dijon: ACM, 2006. 1400–1401. [doi: 10.1145/1141277.1141601]

[6] De Avila SEF, Lopes APB, da Luz Jr A, de Albuquerque Araújo A. VSUMM: A mechanism designed to produce static video summaries and a novel evaluation method. Pattern Recognition Letters, 2011, 32(1): 56–68. [doi: 10.1016/j.patrec.2010.08.004]

[7] Elhamifar E, Sapiro G, Vidal R. See all by looking at a few: Sparse modeling for finding representative objects. In: Proc. of the 2012 IEEE Conf. on Computer Vision and Pattern Recognition. Providence: IEEE, 2012. 1600–1607. [doi: 10.1109/CVPR.2012.6247852]

[8] Elhamifar E, Sapiro G, Sastry SS. Dissimilarity-based sparse subset selection. IEEE Trans. on Pattern Analysis and Machine Intelligence, 2016, 38(11): 2182–2197. [doi: 10.1109/TPAMI.2015.2511748]

[9] Zhou KY, Qiao Y, Xiang T. Deep reinforcement learning for unsupervised video summarization with diversity-representativeness reward. In: Proc. of the 32nd AAAI Conf. on Artificial Intelligence. New Orleans: AAAI, 2018. 7582–7589.

[10] Mahasseni B, Lam M, Todorovic S. Unsupervised video summarization with adversarial LSTM networks. In: Proc. of the 2017 IEEE Conf. on Computer Vision and Pattern Recognition. Honolulu: IEEE, 2017. 2982–2991. [doi: 10.1109/CVPR.2017.318]

[11] Jung Y, Cho D, Kim D, Woo S, Kweon IS. Discriminative feature learning for unsupervised video summarization. In: Proc. of the 33rd AAAI Conf. on Artificial Intelligence. Honolulu: AAAI, 2019. 8537–8544. [doi: 10.1609/aaai.v33i01.33018537]

[12] Zhang K, Chao WL, Sha F, Grauman K. Summary transfer: Exemplar-based subset selection for video summarization. In: Proc. of the 2016 IEEE Conf. on Computer Vision and Pattern Recognition. Las Vegas: IEEE, 2016. 1059–1067. [doi: 10.1109/CVPR.2016.120]

[13] Gygli M, Grabner H, Riemenschneider H, Van Gool L. Creating summaries from user videos. In: Proc. of the 13th European Conf. on Computer Vision. Zurich: Springer, 2014. 505–520. [doi: 10.1007/978-3-319-10584-0_33]

[14] Song YL, Vallmitjana J, Stent A, Jaimes A. TVSum: Summarizing web videos using titles. In: Proc. of the 2015 IEEE Conf. on Computer Vision and Pattern Recognition. Boston: IEEE, 2015. 5179–5187. [doi: 10.1109/CVPR.2015.7299154]

[15] Potapov D, Douze M, Harchaoui Z, Schmid C. Category-specific video summarization. In: Proc. of the 13th European Conf. on Computer Vision. Zurich: Springer, 2014. 540–555. [doi: 10.1007/978-3-319-10599-4_35]

[16] Zhang K, Chao WL, Sha F, Grauman K. Video summarization with long short-term memory. In: Proc. of the 14th European Conf. on Computer Vision. Amsterdam: Springer, 2016. 766–782. [doi: 10.1007/978-3-319-46478-7_47]

[17] Zhao B, Li XL, Lu XQ. Hierarchical recurrent neural network for video summarization. In: Proc. of the 25th ACM Int'l Conf. on Multimedia. Mountain View: ACM, 2017. 863–871. [doi: 10.1145/3123266.3123328]

[18] Zhao B, Li XL, Lu XQ. TTH-RNN: Tensor-train hierarchical recurrent neural network for video summarization. IEEE Trans. on Industrial Electronics, 2021, 68(4): 3629–3637. [doi: 10.1109/TIE.2020.2979573]

[19] Ji Z, Xiong KL, Pang YW, Li XL. Video summarization with attention-based encoder-decoder networks. IEEE Trans. on Circuits and Systems for Video Technology, 2020, 30(6): 1709–1717. [doi: 10.1109/TCSVT.2019.2904996]

[20] Yao T, Mei T, Rui Y. Highlight detection with pairwise deep ranking for first-person video summarization. In: Proc. of the 2016 IEEE Conf. on Computer Vision and Pattern Recognition. Las Vegas: IEEE, 2016. 982–990. [doi: 10.1109/CVPR.2016.112]

[21] Zhang K, Grauman K, Sha F. Retrospective encoders for video summarization. In: Proc. of the 15th European Conf. on Computer Vision. Munich: Springer, 2018. 391–408. [doi: 10.1007/978-3-030-01237-3_24]

[22] Huang SY, Li X, Zhang ZF, Wu F, Han JW. User-ranking video summarization with multi-stage spatio-temporal representation. IEEE Trans. on Image Processing, 2019, 28(6): 2654–2664. [doi: 10.1109/TIP.2018.2889265]

[23] Nguyen DK, Okatani T. Improved fusion of visual and language representations by dense symmetric co-attention for visual question answering. In: Proc. of the 2018 IEEE/CVF Conf. on Computer Vision and Pattern Recognition. Salt Lake City: IEEE, 2018. 6087–6096. [doi: 10.1109/CVPR.2018.00637]

[24] Yu Z, Yu J, Cui YH, Tao DC, Tian Q. Deep modular Co-attention networks for visual question answering. In: Proc. of the 2019 IEEE/CVF Conf. on Computer Vision and Pattern Recognition. Long Beach: IEEE, 2019. 6274–6283. [doi: 10.1109/CVPR.2019.00644]

[25] Yu J, Li J, Yu Z, Huang QM. Multimodal transformer with multi-view visual representation for image captioning. IEEE Trans. on Circuits and Systems for Video Technology, 2020, 30(12): 4467–4480. [doi: 10.1109/TCSVT.2019.2947482]

[26] Cornia M, Stefanini M, Baraldi L, Cucchiara R. Meshed-memory transformer for image captioning. In: Proc. of the 2020 IEEE/CVF Conf. on Computer Vision and Pattern Recognition. Seattle: IEEE, 2020. 10575–10584. [doi: 10.1109/CVPR42600.2020.01059]

[27] He KM, Gkioxari G, Dollár P, Girshick R. Mask R-CNN. In: Proc. of the 2017 IEEE Int'l Conf. on Computer Vision. Venice: IEEE, 2017. 2980–2988. [doi: 10.1109/ICCV.2017.322]

[28] Ren SQ, He KM, Girshick R, Sun J. Faster R-CNN: Towards real-time object detection with region proposal networks. IEEE Trans. on Pattern Analysis and Machine Intelligence, 2017, 39(6): 1137–1149. [doi: 10.1109/TPAMI.2016.2577031]

[29] He KM, Zhang XY, Ren SQ, Sun J. Deep residual learning for image recognition. In: Proc. of the 2016 IEEE Conf. on Computer Vision

and Pattern Recognition. Las Vegas: IEEE, 2016. 770–778. [doi: 10.1109/CVPR.2016.90]

[30] Lin TY, Dollár P, Girshick R, He KM, Hariharan B, Belongie S. Feature pyramid networks for object detection. In: Proc. of the 2017 IEEE Conf. on Computer Vision and Pattern Recognition. Honolulu: IEEE, 2017. 2117–2125. [doi: 10.1109/CVPR.2017.106]

[31] Glorot X, Bordes A, Bengio Y. Deep sparse rectifier neural networks. Journal of Machine Learning Research, 2011, 15(1): 315–323.

[32] Kingma DP, Ba J. Adam: A method for stochastic optimization. In: Proc. of the 3rd Int'l Conf. on Learning Representations. San Diego: ICLR, 2015. 1–15.

[33] Lu JS, Goswami V, Rohrbach M, Parikh D, Lee S. 12-in-1: Multi-task vision and language representation learning. In: Proc. of the 2020 IEEE/CVF Conf. on Computer Vision and Pattern Recognition. Seattle: IEEE, 2020. 10434–10443. [doi: 10.1109/CVPR42600.2020.01045]

[34] Lu JS, Batra D, Parikh D, Lee S. ViLBERT: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. In: Proc. of the 33rd Int'l Conf. on Neural Information Processing Systems. Vancouver: NIPS, 2019. 2.

[35] Sharma P, Ding N, Goodman S, Parikh D, Lee S. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In: Proc. of the 56th Annual Meeting of the Association for Computational Linguistics. Melbourne: ACL, 2018. 2556–2565. [doi: 10.18653/v1/P18-1238]

[36] Chen YC, Li LJ, Yu LC, Kholy AE, Ahmed F, Gan Z, Cheng Y, Liu JJ. UNITER: Universal image-text representation learning. In: Proc. of the 16th European Conf. on Computer Vision. Glasgow: Springer, 2020. 104–120. [doi: 10.1007/978-3-030-58577-8_7]

[37] Yu Z, Cui YH, Yu J, Wang M, Tao DC, Tian Q. Deep multimodal neural architecture search. In: Proc. of the 28th ACM Int'l Conf. on Multimedia. Seattle: ACM, 2020. 3743–3752. [doi: 10.1145/3394171.3413977]

[38] Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, Kaiser Ł, Polosukhin I. Attention is all you need. In: Proc. of the 31st Int'l Conf. on Neural Information Processing Systems. Long Beach: Curran Associates Inc., 2017. 6000–6010.

[39] Wu YH, Schuster M, Chen ZF, *et al*. Google's neural machine translation system: Bridging the gap between human and machine translation. arXiv: 1609.08144, 2016.

[40] Lin TY, Goyal P, Girshick R, He KM, Dollar P. Focal loss for dense object detection. In: Proc. of the 2017 IEEE Int'l Conf. on Computer Vision. Venice: IEEE, 2017. 2999–3007. [doi: 10.1109/ICCV.2017.324]

[41] Xu HJ, Das A, Saenko K. Two-stream region convolutional 3D network for temporal activity detection. IEEE Trans. on Pattern Analysis and Machine Intelligence, 2019, 41(10): 2319–2332. [doi: 10.1109/TPAMI.2019.2921539]

[42] Rochan M, Wang Y. Video summarization by learning from unpaired data. In: Proc. of the 2019 IEEE/CVF Conf. on Computer Vision and Pattern Recognition. Long Beach: IEEE, 2019. 7894–7903. [doi: 10.1109/CVPR.2019.00809]

李群(1984－), 女, 博士, 副教授, 主要研究领域为计算机视觉.

张锋(1989－), 男, 博士, 讲师, CCF 专业会员, 主要研究领域为模式识别, 计算机视觉.

肖甫(1980－), 男, 博士, 教授, 博士生导师, CCF 高级会员, 主要研究领域为计算机网络.

李延超(1990－), 男, 博士, 讲师, CCF 专业会员, 主要研究领域为机器学习, 人体动作识别.

张子屹(1998－), 男, 硕士生, 主要研究领域为深度学习, 人体姿态估计.