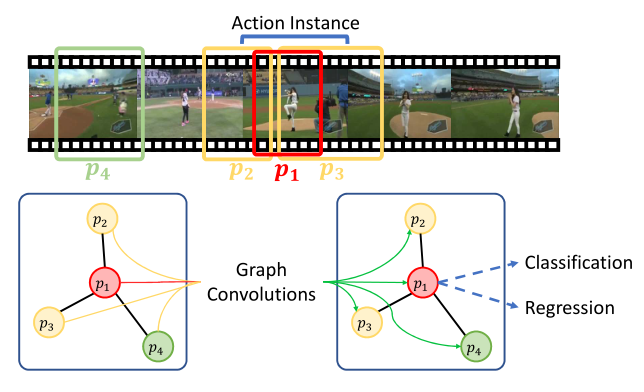
《Graph Convolutional Networks for Temporal Action Localization》

目前先进的时序动作定位系统都是单独处理每个proposal并没有考虑proposal之间的关系

然而proposal之间的联系在动作定位上有很大作用。本文使用GCNs来利用proposal之间的联系进行动作定位。首先构建一个图·，每个proposal作为节点，proposal之间的联系用边表示。使用两种联系，1.表示每个proposal的背景信息，2.不同动作之间的相关性。使用GCNs训练该图。

Introduction

目前大多数动作检测方法采用两级流水线：先生成一组1D的proposal,然后对proposal分别进行分类与时间边界回归



原来的卷积网络只用p1作预测，如果我们使用p2,p3辅助p1作预测，我们会获得更多上下问信息，更好的作时序边界回归。P4是背景，对动作分类有帮助。预测proposal之间的关系，一个应用于时序的sfle-attention机制可以使用，但是它计算一个proposal与其他所有proposal的关系，计算开销太大。GCNs被扩展到非网格数据（社交网络）的处理（图像就是网格数据）。这样。我们将proposal看作节点，可以使用GCNs处理相邻proposal的联系

我们使用两种边连接节点。一：类似p2,p3，为p1提供上下文信息的边；二：背景边，类似p4，p1从p4获得信息。

使用两个GCNs分别作分类，回归。并设计一个抽样的策略训练GCNs来降低运算开销

贡献：

第一个使用proposal之间的关系经行时序动作定位

对proposal之间的关系进行建模，并使用GCNS进行调整

相关工作：

时序动作定位，方法可以分为三类：

1：执行帧或段级分类的方法，其中需要平滑和合并步骤来获得时间边界

《Convolutional-de-convolutional networks for precise temporal action localiza-tion in untrimmed videos. 》

<Temporal activity detection in untrimmed videos with recurrent neural networks.>

<Iterative-winners-out networks for weakly supervised temporal action localization.>

2:采用两阶段框架的方法，包括proposal生成、分类和边界细化

<Temporal action localization in untrimmed videos via multi-stage cnns.>

<Temporal action detction with structured segment networks. >

<Region convolutional 3d network for temporal activity detection>

3:端到端的体系结构，包括proposal的生成和分类

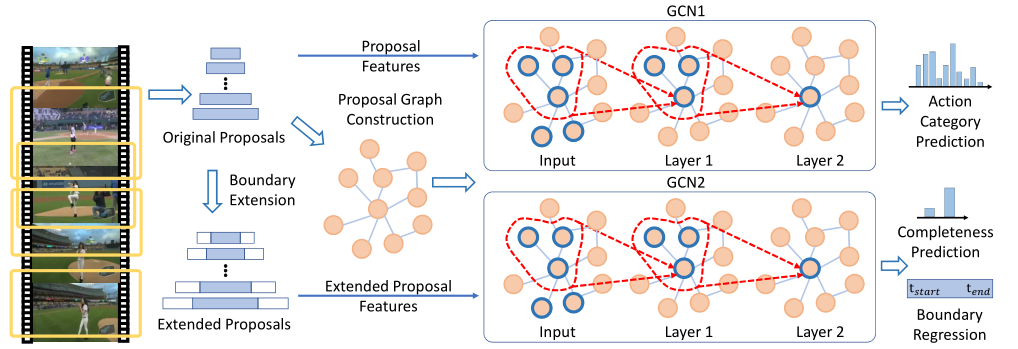
<End-to-end learning of action detection from frame glimpses in videos>

<Single shot temporal action detection. >

<End-to-end, single-stream temporal action detection in untrimmed videos. >

我们的工作使用第二个策略，首先生成proposal，然后使用这些proposal进行分类，回归。

滑动窗口，3D-ConvNet,FasterNet等都忽略了上下文背景信息



为提高GCN计算效率

节点层面：《Inductive representation learning on large graphs.》SAGE

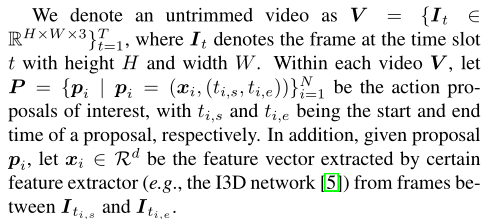
Layer-wise:《ast learning with graph convolutional networks via importance sampling.》FastGCN

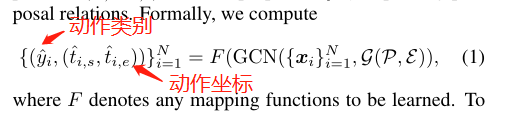
Layer-adapt《Adaptive sampling towards fast graph representation learning. 》AS-GCN

我们使用SAGE策略

方法

符号约定：





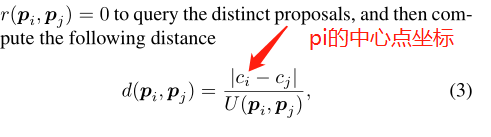
Proposal图与已提取的proposal特征作为输入，enhanced 特征也被用来预测类别与时序边界--P-GCN

怎样构建proposal图？

上下文边：如果R(pi,pj)大于阈值，则在pi与pj节点建立边。R(pi,pj)=I(pi,pj)/U(pi,pj)

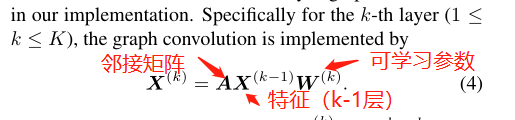
I(pi,pj):两个proposal的交集，U(pi,pj)：两个proposal的并集

背景边：先定位是远的proposal:R（pi,pj）=0,然后计算d(pi,pj)



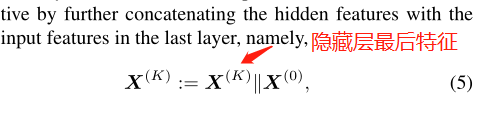
If d(pi,pj)小于阈值，创建边

如何使用GCN根据proposal图进行定位，与回归？



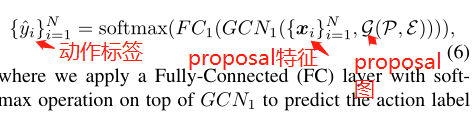
X（0）是输入特征。在每层输出，使用Relu激活函数

发现将输入特征与隐藏层最后一层特征连接更高效

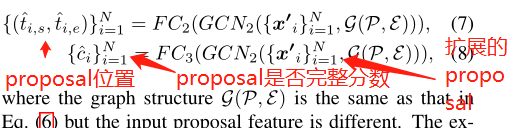


使用两个GCN分别训练

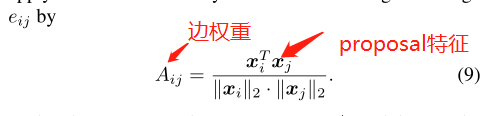
一个：输入原始proposal x(i)预测动作标签



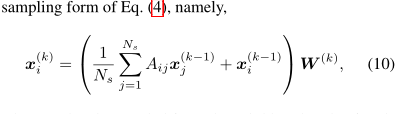
第二个：单独使用位置分数无法判断proposal是否完整，我门结合tlou与位置分数可以更好定位动作



邻接矩阵A：

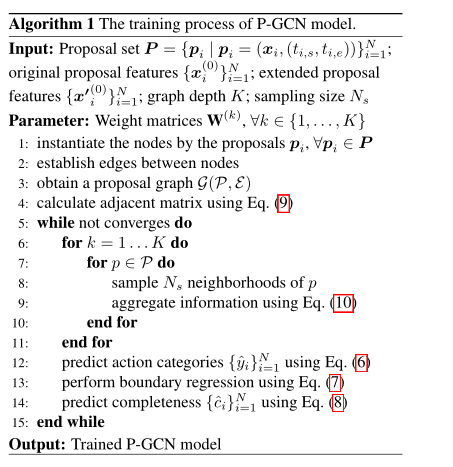


高效训练：自底向上抽样



k层特征由k-1层抽样Ns个邻居决定，并使用self attention 机制

算法描述：



通过比较MLP,与GCN的不同结合方式。来得出proposal信息传递对结果的影响

使用max pooling 一样可以让proposal之间的信息传递，同样做了消融实验