

PLS网络模型

时间槽 $T = \{1, 2, \dots\}$

V_p 内容提供商

V_s CDN服务器端

V_c 观看者

两个节点延迟 \Rightarrow 一个 slot

一个视频 i 在时间 t 的可用带宽 $C_e(t)$ (时间变量)
 $C_e(t) \in [0, C_e^{\max}]$

在切片PLS中的FOV预测

$$F = \{f_1, f_2, \dots, f_{|F|}\}$$

一个视频 $\Rightarrow n$ 个帧

\downarrow
PLS库

不同帧的识别
内容

将 f_i 的帧定义为 $F_i = \{f_i^1, \dots, f_i^t, \dots\}$

等长矩形帧集 \Rightarrow FOV预测可以表示为

$$M_{L \times K} = \begin{bmatrix} m_{1,1} & \dots & m_{1,L} \\ \vdots & m_{k,l} & \vdots \\ m_{k,1} & \dots & m_{k,L} \end{bmatrix} \quad \begin{matrix} \text{行} \\ \text{列} \\ k, L \end{matrix}$$

$$k \in [1, K]$$

$$l \in [1, L]$$

$$m_{k,l} \in \{0, 1\}$$

$$\text{真实视野} = M^*(t)$$

$B = \{b_1, b_2 \dots b_{|B|}\}$ 不同比特率

$Q_v = \begin{bmatrix} d_{1,1} & \dots & d_{1,L} \\ \vdots & d_{K,L} & \vdots \\ d_{K,1} & \dots & d_{K,L} \end{bmatrix}$ 观察者在 t 时刻对 v frame 的比特率需求

$d_{k,l} \in B.$

CDN 有 video 所有切片, 但观察者 只有自己需求的切片

在信息上,
时间偏移可以被算出来.

$T_{u,v}(t)$ \Leftarrow 一对节点 (u,v) 在时间 t 上的时间偏移

Fedlive 系统框架

两个阶段 { 分布式学习阶段
有 3 个区域的传输阶段 (服务器, CDN, 内容提供)

三种主要类型节点 { 内容提供商
CDN (内容分发网络)
观看者

1. 分布式学习阶段

① 训练及传播 { 本地观看记录
PVR 记录及传播提供的观看记录

② 输出 { 本地信息
观看记录
本地信息被 CDN 服务器拦截
作为同步要素源 ($m^* + m$) 的输入

损失函数 $M + M^*$ (假设 M 的 FOV = 预测交互数 + 实际上用户的 FOV = 预测交互数)

{	2	TP	✓	预测用户观看, 用户实际也看
	1	FP FN	✗	未预测准确的
	0	TN	✓	预测用户不会观看正是, 用户确实未看

③ CDN服务器
收集完所有损失信息 \rightarrow 调用基于梯度的用户聚类算法。

CDN总结每个集群的损失值 \rightarrow 与聚类结果一起
更新推荐内容提供商 \rightarrow 内容提供商用加权平均损失值更新集群的损失值。

④ 统一推荐作为推荐内容台发布内容
会根据预测结果, 确定不同 time 的推荐优先级

2. 传输阶段

① 内容提供商根据服务器提交的请求结果

向CDN服务器提交设备模型和所需分辨率的切片的360°视频

② CDN服务器将model 给观众, 并根据观众要求提供基于切片的直播.

若服务器请求超过了服务器容量。为卸载流量, CDN服务器有以下几种策略

将观众请求重新定向到其他CDN服务器

重新定向到同样访问该服务器并愿意提供取走闲资源的 (同一集群观众是首选, 因为可以节省切片以及模型)