

1. 利用机器学习来对现阶段 **RTO** 的值进行修改，定义一个更加精准的 **RTO** 的值？

$$\text{新的 RTTs} = (1-\alpha) * (\text{旧的 RTTs}) + \alpha * (\text{新的 RTT 样本})$$

$$\text{RTO} = \text{RTTs} + 4 * \text{RTT}_D$$

RTT_D 中第一次设置 RTT_D 应该为 RTT 样本值的一半。之后使用下面的计算公式：

$$\text{新的 RTT}_D = (1-\beta) * (\text{旧的 RTT}_D) + \beta * |\text{RTTs} - \text{新的 RTT 样本}|$$

找到相应的 feature：

(1)RTT 2. 上一个 RTT 3. α 的值 4. β 的值

和历史信息有关系，具体采用何种机器学习算法？

LSTM：时序数据

2. 利用机器学习对丢包的情况进行分类，判定哪一种情况是拥塞丢包，哪种是因为噪音产生的丢包？

论文《基于往返延迟抖动区分丢包的 TCPW 改进》

计算往返延时抖动的测量值：

$$j_{\text{sample}}(k) = rtt_k - rtt_{k-1}$$

$j_{\text{sample}}(k)$ —指的是第 k 个往返延迟抖动测量值， rtt_k 为第 k 个往返延迟

对往返延迟抖动进行估算： $j_k = \alpha * j_{k-1} + (1-\alpha) * j_{\text{sample}}(k)$

j_k 指的是第 k 个往返延迟抖动的估计值， $j_{\text{sample}}(k)$ 为测量值。 α 的值给定为 0.125，是否可以通过训练进行改进？

提出网络拥塞标识 NCI。通过 NCI 对网络拥塞程度进行细粒度划分。

$\text{NCI} = (j_k - T)/T$ 。常量 T 设置为 0.5ms，是否可以通过机器学习的方法进行改进？

根据 NCI 的值，将网络拥塞划分为 5 个等级：

NCI 值的范围	$(-\infty, -4)$	$(-4, -1)$	$[-1, 0.5]$	$(0.5, 1)$	$(1, +\infty)$

网络拥塞等级	1	2	3	4	5
网络拥塞程度	严重欠载	轻微欠载	负载正常	轻度拥塞	严重拥塞

拥塞等级小于 4 时，网络状况较好，此刻丢包认为是无线丢包；拥塞等级大于等于 4 时，出现了拥塞，认为是拥塞丢包。

可否借鉴这个思想，判断为拥塞丢包的话就是用自动分配不同的网络拥塞控制算法，如果是第 1, 2 等级的丢包，就加速传播从而充分的利用带宽？