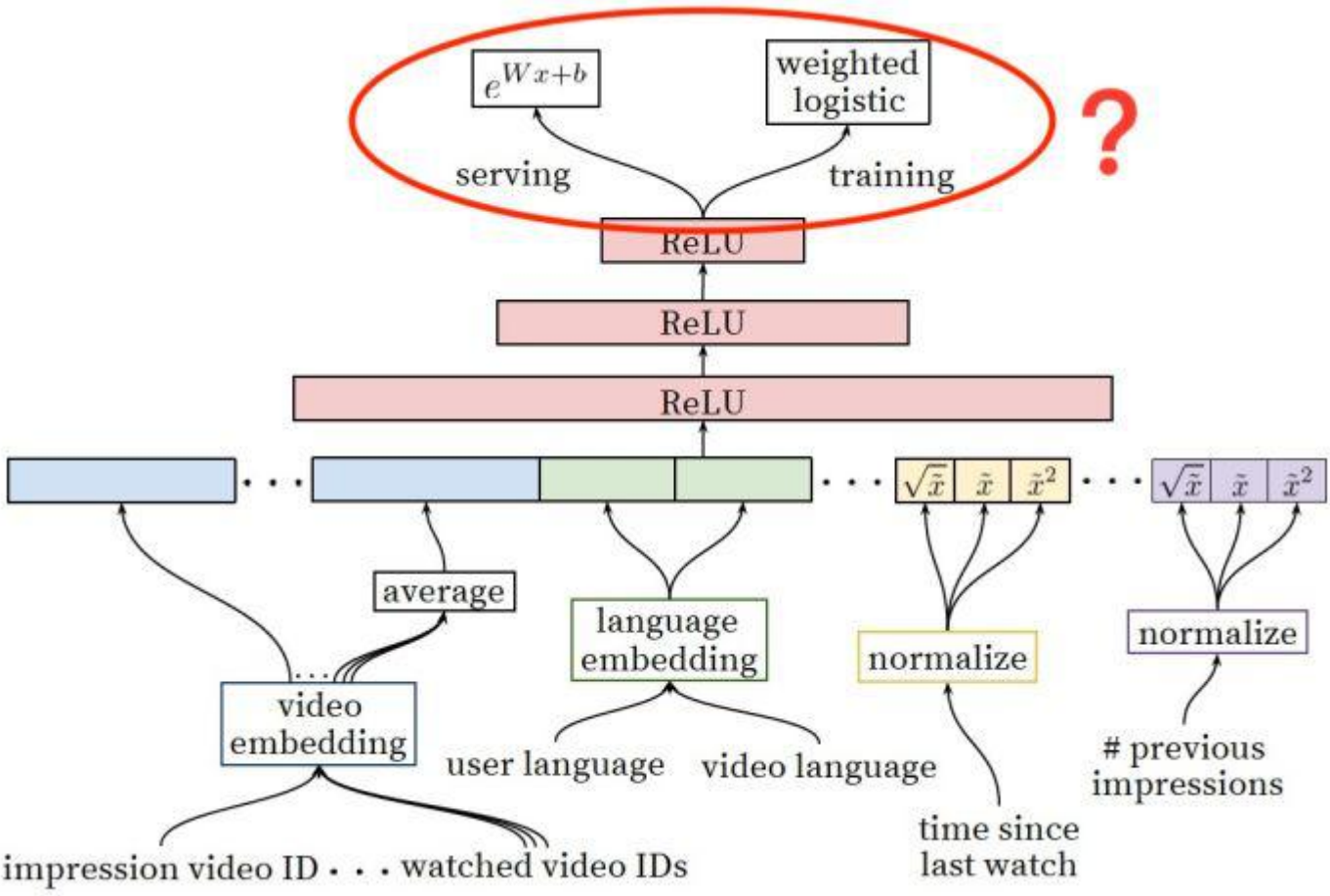
YouTube 深度学习推荐系统中模型 serving 的问题。

这里我们再详细陈述一下这个问题：



YouTube 深度学习推荐系统中 Ranking Model 的架构图

上图是 YouTube 推荐系统排序模型（Ranking Model）的架构图，我们不再重复讲解模型的细节，而是把关注的焦点放在最后的输出层：

为什么 Ranking Model 采用了 weighted logistic regression 作为输出层？在模型 serving 过程中又为何没有采用 sigmoid 函数预测正样本的 probability，而是使用 e^{Wx+b} 这一指数形式预测用户观看时长？

对于传统的深度学习架构，输出层往往采用 LR 或者 Softmax，在线上预测过程中，也是原封不动的照搬 LR 或者 softmax 的经典形式来计算点击率（广义地说，应该是正样本概率）。

而 YouTube 这一模型的神奇之处在于，输出层没有使用 LR，而是采用了 Weighted LR，模型 serving 没有采用 sigmoid 函数的形式，而是使用了e^{Wx+b}这一指数形式。按照原文说法，这样做预测的就是用户观看时长？没有任何其他表情能像这位小哥一样表达我初读论文的感受。

搞清楚这件事情并不是一件容易的事情，我们要从逻辑回归的**本质意义**上开始。

几乎所有算法工程师的第一堂课就是逻辑回归，也肯定知道逻辑回归的数学形式就是一个线性回归套 sigmoid 函数：

$$h_\theta(x) = \frac{1}{1 + e^{-\theta^T x}}$$

逻辑回归的数学形式

但为什么选择 sigmoid 函数？难道仅仅是 sigmoid 函数能把值域映射到 0-1 之间，符合概率的物理意义这么简单吗？

答案显然不会这么肤浅。

为解释这个问题，首先我们需要定义一个新的变量——**Odds**，中文可以叫**发生比**或者**机会比**。

$$Odds = \frac{p}{1-p}$$

Odds 的定义

假设一件事情发生的概率是 p，那么 **Odds 就是一件事情发生和不发生的比值**。

如果对 Odds 取自然对数，再让 ln(Odds)等于一个线性回归函数，那么就得到了下面的等式。

$$logit(p) = \ln(\frac{p}{1-p}) = \theta_0 + \theta_1 x_1 + \theta_2 x_2$$

其中 ln(p/(1-p))就是大名鼎鼎的 **logit 函数**，logistics regression 又名 logit regression，上面的式子就是逻辑回归的由来。我们再做进一步运算，就可以转变成我们熟悉的逻辑回归的形式：

$$ln(/fracp1-p) = /theta^T x + /Rightarrow + /fracp1-p+ = +e^{/theta^T x} + /Rightarrow + p = /frac11+e^{-/theta^T x} + /Rightarrow + p = sigmoid(/theta^T x)$$

到这里大家应该已经完全明白了 LR 的推导过程了。

那么再对 $ln(Odds) = \theta^T x$ 这个等式做一个小小的转换，两边取自然底数：

$$ln(Odds) = \theta^T x \Rightarrow Odds = e^{\theta^T x} = YouTubeServingFunction$$

YouTube 的 Serving 函数计算的不是别的，正是 Odds！

但我们还没有到达终点，因为 YouTube 要预测的明明是用户观看时长，怎么就成了 Odds 了？

这就要提到 YouTube 采用的独特的训练方式 Weighted LR，这里的 Weight，对于正样本 i 来说就是观看时长 Ti，对于负样本来说，则指定了单位权重 1。

Weighted LR 的特点是，正样本权重 w 的加入会让正样本发生的几率变成原来的 w 倍，也就是说样本 i 的 Odds 变成了下面的式子：

$$Odds(i) = \frac{w_i p}{1 - w_i p}$$

由于在视频推荐场景中，用户打开一个视频的概率 p 往往是一个很小的值，因此上式可以继续简化：

$$Odds(i) = \frac{w_i p}{1 - w_i p} \approx w_i p = T_i p = E(T_i)$$

而且由于 YouTube 采用了用户观看时长 Ti 作为权重，因此式子进一步等于 Tip，这里真相就大白了，由于 p 就是用户打开视频的概率，Ti 是观看时长，因此 Tip 就是用户观看某视频的期望时长！

因此，YouTube 采用 e^{Wx+b} 这一指数形式预测的就是曝光这个视频时，用户观看这个视频的时长的期望！利用该指标排序后再进行推荐，是完全符合 YouTube 的推荐场景和以观看时长为优化目标的设定的。

再简要总结一下 YouTube Ranking Model 的 Serving 过程要点。

1. e^{Wx+b} 这一指数形式计算的是 Weighted LR 的 Odds；
2. Weighted LR 使用用户观看时长作为权重，使得对应的 Odds 表示的就是用户观看时长的期望；
3. 因此，Model Serving 过程中 e^{Wx+b} 计算的正是观看时长的期望。

最后按惯例给大家留一个讨论的问题，欢迎大家各抒己见：

训练 Weighted LR 一般来说有两种办法：

1. 将正样本按照 weight 做重复 sampling，然后输入模型进行训练；
2. 在训练的梯度下降过程中，通过改变梯度的 weight 来得到 Weighted LR。

问题是这两种训练方法得到的结果有没有不同？有没有其他 Weighted LR 的训练方法？