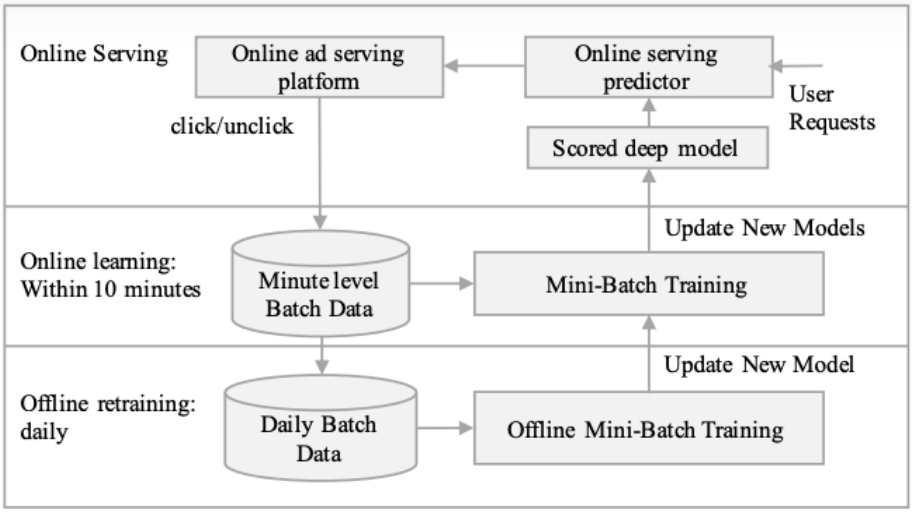


京东2023-全新CTR蒸馏模型，引领蒸馏精度提升新篇章

SmartMindAI
专注搜索、广告、推荐、大模型和人工智能最新技术，欢迎关注我
已关注

18 人赞同了该文章

Introduction

在许多实际场景中，例如大型电商、社交网络、推荐系统等，实时预测需求强烈，往往需要实时处理海量数据，同时实时调整模型参数。此外，由于这些场景通常涉及到大量用户行为和交互信息，因此模型训练和在线学习都需要非常高的效率和精度。然而，现有的知识蒸馏方法在这些场景中可能无法充分发挥其优势。

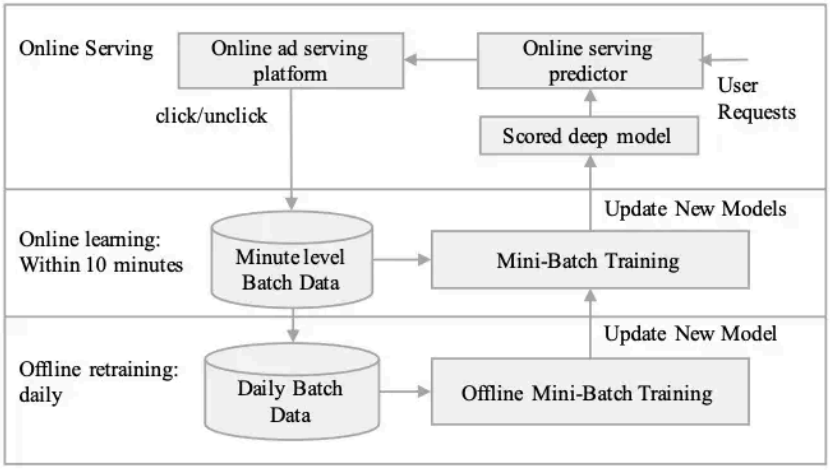
这主要是因为知识蒸馏方法通常依赖于教师模型的优异性能，并且假设教师模型和学生模型的参数差距较小。然而，在实时预测环境中，教师模型和学生模型的性能差距以及参数差距都可能较大，从而限制了知识蒸馏的效果。因而，如何在实时预测环境中有效利用知识蒸馏是一个亟待解决的问题。

本研究将探讨在实时预测环境下如何设计和优化知识蒸馏方法，以提高模型的效率和准确性。本文已经证明了我们提出的置信度排序损失能够有效地优化CTR预测，而且已经在实际广告系统中应用于精排排序阶段。此外，我们还给出了理论和实验证据，证实了我们提出的置信度排序损失优于传统的蒸馏方法。

Methods

Preliminaries

正如图所示，现实中CTR预测管道可以分为三个部分：



知乎 @SmartMindAI

Figure 1: brief system overview of CTR prediction pipeline.

1. 离线训练阶段： 样本x和真实标签y构成一个数据集$\mathcal{D}_{old}$。 使用优化方法（如交叉熵）优化模型参数θ以最小化输出概率z与真实标签之间的差距。
2. 在线训练阶段： 新的数据点不断添加到训练集中。 模型按照顺序在每个新的数据点上进行训练。
3. 在线预测阶段： 当有新的用户请求时，使用最新训练好的模型对新的数据点进行预测，得到预测概率z'。
   通过这种方式，我们能够在现实世界的CTR预测管道中充分利用我们的置信度排序损失。

$$\arg\min_{\theta} \mathcal{L}_{old}, \quad \text{where} \quad \mathcal{L}_{old} \triangleq \mathbb{E}_{(x,y)\sim\mathcal{D}_{old}}[\ell(y, f(x;\theta))]$$

给定指标函数$\mathcal{M}(y, \hat{y})$，其中y和$\hat{y}$分别代表真实标签和预测值，我们的目标是最小化分类风险，即期望在线学习后的模型能够生成更接近真实标签的预测结果。因此，我们的目标函数可以表示为：

$$J(f_{online}, \mathcal{D}_{new}) = E_{y_{new}\sim P(y|X)}[\mathcal{M}(y_{new}, f_{online}(X))]$$

其中，f_{online}是在线学习后的模型，$P(y|X)$是模型在数据集中生成的真实标签的概率分布，$E_{y_{new}\sim P(y|X)}$表示预期值。

$$\arg\min_{\theta} \mathcal{L} \quad s.t. \quad \mathcal{C}(f) > 0$$

在这个基础上，我们进一步提出了一个更灵活的优化方案，即将原模型的指标分数减去在线学习后的模型的指标分数作为评判标准。这样，我们就可以通过调整这两个分数的差来控制在线学习的强度。

具体的来说，我们可以设计一个两阶段的分批训练策略，如图所示。首先，我们将收集数据，使用在线学习策略在线优化模型。然后，我们再使用离线训练策略在所有数据上训练模型，并比较原模型和在线学习后的模型的指标分数的差。如果差值大于某个阈值，我们就增加在线学习的时间，否则就停止在线学习，继续离线训练。

Confidence Ranking

在这里，我们可以通过改变期望度量目标的方式来控制模型的精度。具体的来说，我们可以定义一个期望度量目标，其中包含了原模型和在线学习后的模型的指标分数的差。这样，我们就可以通过调整这个分数的差来控制模型的精度。 具体来说，我们可以定义一个期望度量目标，如$C_{acc}(f) = \frac{1}{n}\sum_{i=1}^{n}\mathbb{I}[y_i f(x_i) > y_i f_{online}(x_i)]$，

其中$y_i f(x_i)$表示原模型在第i个样本上的预测结果，$y_i f_{online}(x_i)$表示在线学习后的模型在第i个样本上的预测结果。 通过改变期望度量目标，我们就可以控制模型的精度。同时，我们也需要注意，不同的期望度量目标可能会产生不同的效果，因此我们需要根据实际情况选择最合适的期望度量目标。

$$\ell_{CR}(f) \triangleq$$

我们选择逻辑损失函数作为我们的评分函数。这种评分函数可以用来衡量两个向量之间的相似程度。 在我们的研究中，我们主要关注的是逻辑排序损失。它的定义为
$\phi_y(u,v) = log(1+exp^{-(u-v)})$,

其中u和v是两个向量。
当$u < v$时，$\phi_y(u,v) < 0$，表示u比v更加靠近正无穷大；
当$u = v$时，$\phi_y(u,v) = 0$;
当$u > v$时，$\phi_y(u,v) > 0$，表示u比v更加远离负无穷大。

在这种情况下，我们可以通过改变评分函数的定义来控制模型的性能。例如，如果我们希望模型的性能更加稳定，我们就可以选择更稳定的评分函数，如平方损失。反之，如果我们希望模型的性能更加敏感，我们就可以选择更敏感的评分函数，如逻辑排序损失。
$\ell_{CR} = \frac{1}{n}\sum_{i=1}^{n} y_i log(1+exp^{-(u-v)}) + (1-y_i) log(1+exp^{(u-v)})$

对于二元分类，为了进一步改进二元分类的双部分排序性能，我们将遵循优化二元分类双部分排序性能的方法。为了达到更好的二元分类双部分排序性能，我们定义了一个期望度量目标，如
$C_{auc}(f) = \frac{1}{nm}\sum_{i=1}^{n}\sum_{j=1}^{m} \mathbb{I}[(f(x_i^+) - f(x_j^-)) > (f_{online}(x_i^+) - f_{online}(x_j^-))]$,

其中x_i^+和x_j^-分别为第i个正样本和第j个负样本。 通过定义这个期望度量目标，我们就可以控制模型的二元分类双部分排序性能。同时，我们也需要注意，不同的期望度量目标可能会产生不同的效果，因此我们需要根据实际情况选择最合适的期望度量目标。
$\ell_{RCR}(f) \triangleq \mathbb{E}_{\{x^+,x^-\}\sim\{\mathcal{P}^+,\mathcal{P}^-\}}\left[\phi(d_f(x^+,x^-) - d_{f_{online}}(x^+,x^-))\right]$

在训练过程中，我们提出了一种新的性能评价方法，即点对点置信度排序损失。这种方法主要是通过对不同样本的预测结果进行排序，来评价模型的性能。 因此，最终的目标定义为：
$\ell_{ce} + \lambda_{CR_{acc}}\ell_{CR} + \lambda_{RCR_{auc}}\ell_{RCR}$

其中ℓ_{ce}是CTR预测的交叉熵损失，ℓ_{CR}和ℓ_{RCR}分别是点对点和关联关系置信度排序损失，$\lambda_{CR_{acc}}$和$\lambda_{RCR_{auc}}$是可以调整的超参数，以控制损失项。对于在提出的关联关系置信度排序损失中采样的pos/neg样本对，我们简单地使用给定迷你批次中的所有可能对。

(**置信度排序的偏差-方差界限**)[$prop\_cr$] 选择任何凸损失 ℓ。假设我们有一个教师 模型 p^t，其对应的经验置信度排序风险

$$\widehat{R}(f) = \frac{1}{N}\sum_{n\in N} y(x_n)\ell(\mathrm{f}(x_n) - f_t(x_n)))$$

和总体风险 $R(f) = \mathbb{E}_x\left[p^*(x)\ell(f(x))\right]$ 其中 $f_t(x_n)$ 是教师输出。对于任何预测器 f:
$\mathcal{X}\to\mathbb{R}^L, \mathbb{E}\left[(\widehat{R}(f) - R(f))^2\right] \le \mathbb{E}\left[(R(f_t))^2\right]$

我们提出了一种新的观点，即置信度排序可以提供一个总下界，该下界始终逼近基于教师模型性能的Bayes概率。但是，这个下界可能并不适合深度学习架构，并且在实际应用中可能会很宽松并且不稳定，特别是在逻辑置信度排序损失的情况下。 我们认为，要想获得置信度排序的全面下界，必须满足一些必要的条件。这些条件可能会对我们的实验产生影响，并且可能会影响到我们的模型性能。 然而，即使在满足这些条件的情况下，置信度排序仍然可以在实践中保持大部分情况下的最佳性能。这表明，置信度排序仍然是一种非常有用的工具，可以帮助我们更好地理解和评估模型的性能。

Experiments on CTR prediction

我们在工业、Avazu和Avito等数据集上评估了我们的方法。

Table 1: The statistic of CTR prediction datasets

Datasets	Users	Items	Fields	Feature size	Instances
Avazu	N/A	N/A	22	2018012	40428967
Avito	3163597	28529	16	3419165	190107687
Industrial	N/A	N/A	59	N/A	12 Billion

知乎 @SmartMindAI

所有的实验都在一个P40 GPU上进行公共数据集的实验，而在一个8 A100 GPU上的实验进行工业数据集的实验。我们通过时间戳将公共数据集分为训练/验证/测试集，最后一天的样本用作测试，倒数第二天的样本用作验证，其余的样本用作训练。为了分割工业数据集，我们将前15天的交通样本用作训练集，最后一天用作测试集。我们总结了数据集的统计信息表。我们采用了两种不同的设置来评估我们的方法。

详细配置如下： 我们采用单次训练策略来模拟在线学习。我们包括了一些标准监督学习和单次训练的基线，以便与当前最先进的结果进行比较。最简单的方法是

(1) ERM：我们使用二元交叉熵损失训练我们的网络；

(2) 多种常见的CTR预测网络架构，如DNN，PNN，DCN，DeepFM；

(3) SC将自我校正模块集成到CTR预测网络中；

(4) 知识蒸馏方法：我们也适应KD和RKD到我们的实验设置。

对于所有损失函数，我们都调整了其损失平衡项λ 的范围，从0.1到2.0。我们在实验中选择了最佳结果。对于公共数据集，最好的$\lambda_{CR_{acc}}$ 和$\lambda_{RCR_{auc}}$ 是$0.4, 0.5$，

对于工业数据集，最好的$\lambda_{CR_{acc}}$ 和$\lambda_{RCR_{auc}}$ 是$0.5, 1.0$。表中比较了我们在点击率预测任务上的测试集AUC分数。 我们观察到PNN在标准监督学习设置下表现最好，但在单次训练设置下表现不佳。我们选择 DeepFM 作为我们的实验基础。 我们的提议方法 CR 击败了所有基准方法。 在

Avazu 和 Avito 上，我们的 CR 和 RCR 都能在经过多个epoch训练后显著优于基准。对于单次训练设置，我们的提议方法优于基础模型，但差距较小。对于工业数据集，我们的方法分别提高了 0.25/0.61/0.31/0.16/0.14% 的 AUC 分数。

Table 2: AUC(%) of test-set performance on Avito, Avazu and Industrial datasets with various backbone and training strategy. * denotes one-pass learning. The results are averaged over 3 runs. Std $\leq$ 0.1%.

Methods	Avazu	Avito	Avazu*	Avito*	Industrial*
DNN	75.05	77.71	74.32	77.50	75.92
DCN	74.99	77.66	74.30	77.58	75.99
PNN	75.06	77.80	74.49	77.57	n/a
DeepFM	75.24	77.73	74.69	77.50	76.02
DeepFM+KD	75.41	78.01	74.83	77.54	76.18
DeepFM+RKD$_l$	75.34	77.83	74.90	77.58	76.10
DeepFM+SC	75.36	77.78	74.85	77.53	76.14
DeepFM+CR$_{acc}$	75.63	78.33	74.98	77.70	76.25
DeepFM+RCR$_{auc}$	75.59	78.59	75.05	**77.73**	76.20
DeepFM+Both	**75.66**	**78.62**	**75.14**	77.75	**76.32**

知乎 @SmartMindAI

Online A/B Experiments

我们的架构分为两部分：第一部分是在在线广告服务平台上，我们收集$f(x;\theta_{online})$的输出作为在线部署的预测，直接决定要展示哪些商品。第二部分是强迫我们的排序损失函数来鼓励网络在重训练和在线学习阶段学习得更好。

Conclusion

从实际应用的角度来看，我们发现了重训练和在线学习阶段学习模型以更好地推广到在线部署模型的问题。为了解决这个问题，我们提出了一种损失框架，名为"置信度排序"。这个框架比较各种模型的输出预测以最大化Surrogate距离指标得分。我们还将这种方法扩展到排序准确性和点击率预测中的Auc。

编辑于 2023-11-25 12:04 · IP 属地北京

京东 机器学习 ctr

理性发言，友善互动

5 条评论 默认 最新

louis
论文名能贴一下吗?
2023-11-17 · 广东 回复 喜欢

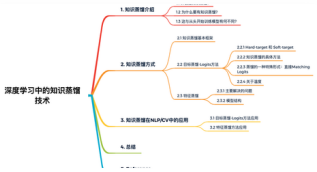
SmartMindAI 作者
辛苦关注后私信
2023-11-17 · 北京 回复 喜欢

黑夜的逐光者
这个avazu的baseline就75吗😳
2023-11-16 · 北京 回复 喜欢

黑夜的逐光者 ▸ **SmartMindAI**
不是 我记得avazu的sota远远不止75
2023-11-16 · 北京 回复 喜欢

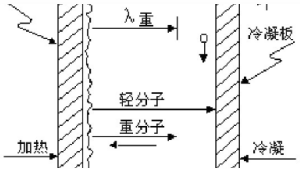
SmartMindAI 作者
工业级的都会比75高
2023-11-16 · 北京 回复 喜欢

推荐阅读



深度学习中的知识蒸馏技术（上）
忆臻 发表于机器学习算...

蒸馏浓缩（溶剂回收）过程中的风险与控制
八千里路 发表于PSM过程...



大宝鉴8：分子蒸馏技术
法式滚筒YEboss

分子蒸馏竟有传统蒸馏没大优势！！
前 言在液-液分离技术当中，（精馏）是最普遍也是应用最泛的技术之一了，我们听过普馏、萃取精馏、共沸精馏、热馏、反应精馏等等，但是短程听过没有？这又是个神马玩意
沧海Che... 发表于化