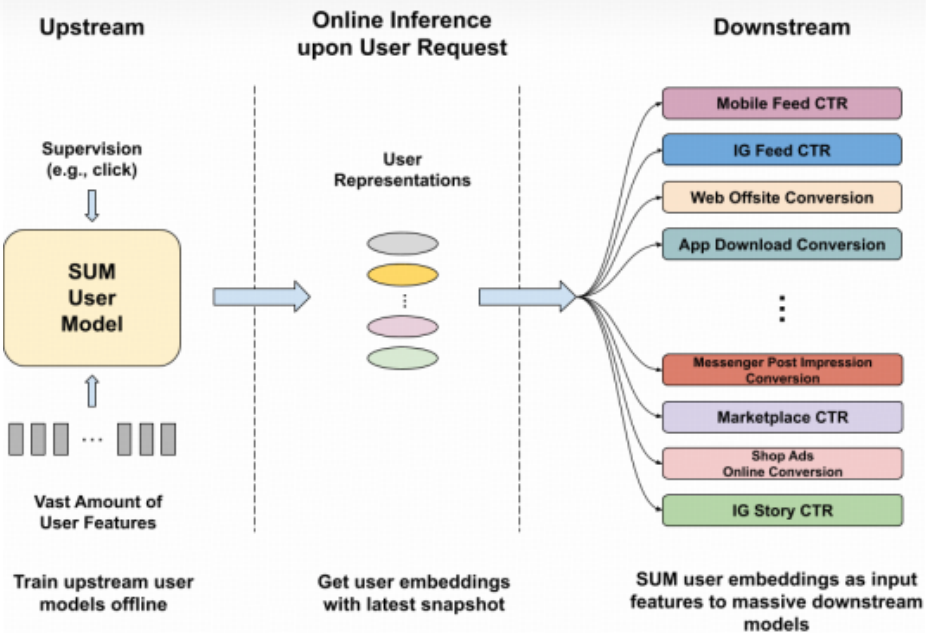
知乎



2024 META新作：SUM技术进行大规模在线用户表示，提升广告个性化效果

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

40 人赞同了该文章

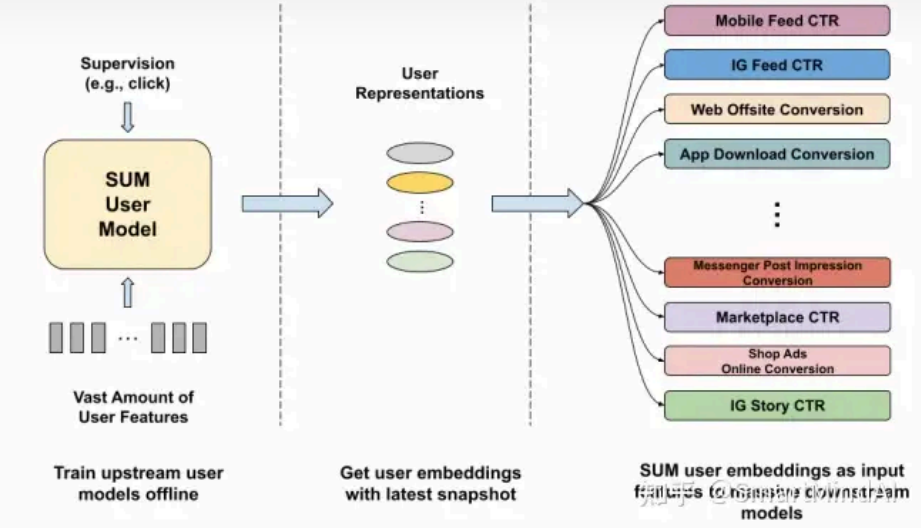
Introduction

个性化推荐是现代在线广告的基础，既提升了广告主的回报也优化了用户体验。个性化的核心在于对用户的理解，过去主要依赖于人工设计的特征和简化的架构。以深度学习为基础的推荐系统的发展改变了这一格局，其利用复杂的神经网络模型来学习微妙的用户表示。然而，实践中遇到的约束，如训练吞吐量、服务延迟以及主机内存限制，限制了它们对大量用户数据的全效利用。对于像Meta这样的全面系统，它包含大量具有不同特性的模型，每天处理数百亿的用户请求，这些限制问题尤为突出。在这样大规模系统内学习用户表示遇到了以下几大挑战：

- **次优的表示**：模型独立学习用户表示通常会导致较差的结果。
- **特征冗余**：模型间重复的用户特征导致训练管道中不必要的重复，从而产生较高的存储开销。
- **专有模型数据稀缺**：服务于利基细分市场的模型缺乏足够的训练数据量来进行全面的用户理解。
- **定制化强度**：为每个模型的具体性能需求专门定制架构和特征选择的策略不可扩展。

我们提出了*扩展用户建模（SUM）*，这是一个在线框架，彻底改变了Meta广告中的用户建模方式。SUM旨在利用高级建模技术，同时遵循实际约束，促进模型间有效的、可扩展的表示共享。

知乎

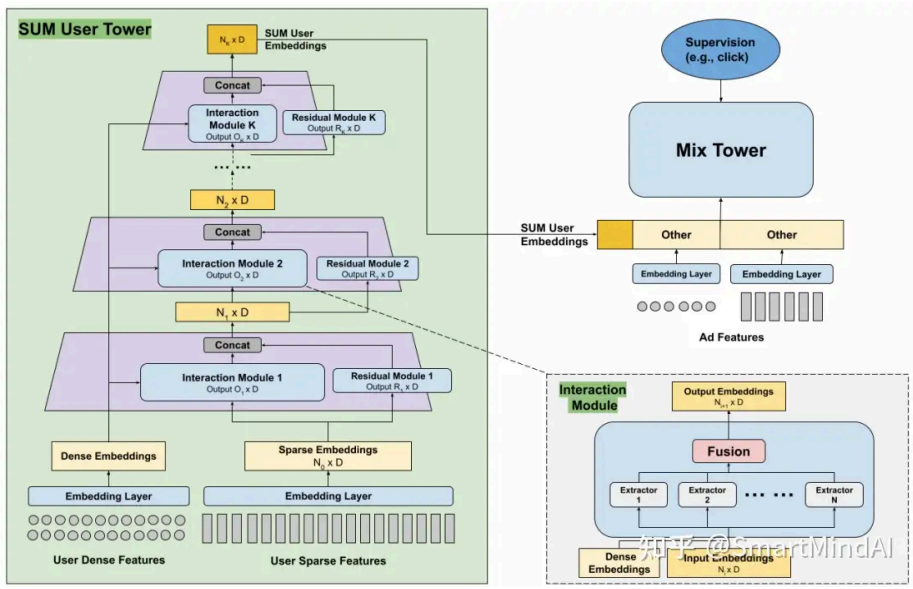


SUM采用了一种上游-下游范式（如图所示），借鉴了用户历史建模的最新工作。我们的方法涉及训练少量大型上游用户模型，使用复杂的架构和多样化的监督，如点击和转化率。用户模型通过处理大量的用户一侧信号（特征）来合成紧凑的用户嵌入（表示）。这些嵌入随后无缝地集成到各种下游生产模型中，传播高级用户建模和表示共享。

SUM的关键设计在于其对用户特征动态性的适应性。我们形成的产品服务系统，即SUM在线异步平台（SOAP），是核心组件，实现了无延迟、异步服务，既保证了嵌入的新鲜度，又克服了复杂模型带来的延迟限制，显著超越了传统的基于离线的方法。与SOAP相辅相成的是，我们开发了一种基于平均池化技术的增强循环训练制度，以保持模型的新鲜度并稳定用户嵌入。 自从其首次部署以来，SUM已经在Meta的数百个生产广告排名模型中部署，带来了离线和在线业务指标的显著改进。

Model Architecture

Preliminary



在图所示的设计中，SUM用户模型采用了著名的DLRM架构来进行用户表示的学习与提取。该模型结构主要分为两大组件:用户塔和混合塔。用户塔负责处理大规模且多元化的用户侧输入特征，并将其转换为精炼的SUM用户嵌入集合。这些嵌入随后与混合塔进一步整合，该层级会将之前生成的嵌入与广告等其他特征进行交互，以加强用户行为理解与预测的准确性。

在特征处理上，输入被划分为两大类

用户塔设计采用了层次化的金字塔结构，其中不同模块相互链接，旨在高效压缩输入特征。在训练过程中，使用残差连接技术来促进信息流动并保持原始特征信息不被丢失。在进行初始的嵌入转换之后，我们可得到N_0个稀疏嵌入以及N_{dense}个维度为D的密集嵌入。这些特征经过整合最终转化为一个共同维度N_K的输出用户嵌入，其中，稀疏嵌入的数量远高于输出嵌入的数量（$N_0 >> N_K$）。在第n个相互作用模块中，这些嵌入通过一系列运算进行结合和处理，以深化对用户特性的理解并提炼关键信息，从而为整体模型提供更丰富且精炼的用户表示。

$$X_n = Concat(Interaction(X_{n-1}), X_{dense}), Residual(X_{n-1}))$$

在图示中，Interaction和Residual分别对应图中的Interaction Module和Residual Module。第n个相互作用模块的输入标记为X_{n-1}。与此同时，用于操作的特征被明确分为两大类:X_{dense}代表密集特征，而X_{sparse}则标记稀疏特征。在实现这样的设计时，这些组件协同工作以优化特征处理和模型性能。∈### MLP（多层感知器⁺）MLP方案主要用于学习一系列复杂的非线性内部表示，旨在从特征中提取更为抽象和深入的含义。

注意力压缩的点降维

压缩点对的点积矩阵是增加模型容量和效率的有效方法。我们引入了高级的注意力压缩和残差连接来学习高级的显式表示，表达式为

$$Y = MLP(Concat(LC(X_{dense}), LC(X)))$$

$$Z = MLP'(Concat(LC'(X_{dense}), LC'(X)))$$

$output = X(X^TY + Z)$其中$FC(*)$表示没有非线性函数的全连接层⁺$LC(*)$表示线性压缩。对于密集特征

$$LC(X_{dense}) = FC(X_{dense})$$

$LC(X)$是X中的稀疏嵌入的一组加权和。我们使用$LC(*)$在$Concat(*)$之前的原因是为了减少$MLP(*)$的输入大小以提高模型效率。Y是注意力权重。Z是残差分支。

深层交叉网络（DCN）

DCN 通过有效的显式和隐式特征交叉学习表达性表示。其基本组件是交叉层，可以由以下方程式说明。

$$X_{n+1} = X_0 \odot (W_n X_n + \mathbf{b}_n) + X_n$$

其中W_n和b_n是可学习的权重和偏置。通过堆叠多个交叉层，模型可以捕捉高阶的向量级和位级的交互。

MLP-Mixer

MLP-Mixer 是一个全MLP架构，最初是为了计算机视觉设计的。这种架构可以被视为一种独特的CNN，其中1x1卷积⁺用于通道融合，单通道深度卷积用于令牌融合，如方程 和 方程所示。

$$Y = X + W_2 \cdot \mathrm{ReLU}\left(W_1 \cdot \mathrm{LayerNorm}(X)^T\right)^T$$

$$Z = Y + W_4 \cdot \mathrm{ReLU}\left(W_3 \cdot \mathrm{LayerNorm}(Y)\right)$$

W_1 W_2 W_3 W_4是可学习的权重。

Mix Tower

混合塔采用了一种类似于DHEN的架构 。除了来自用户塔的SUM用户嵌入，它不包含任何用户侧特征作为输入。这种设计决策的目的是增强上游的训练效率，并促使模型主要通过用户塔来精细调整用户表示。我们的经验表明，尽管更复杂的混合塔架构可以增强上游模型的预测性能，但它并不

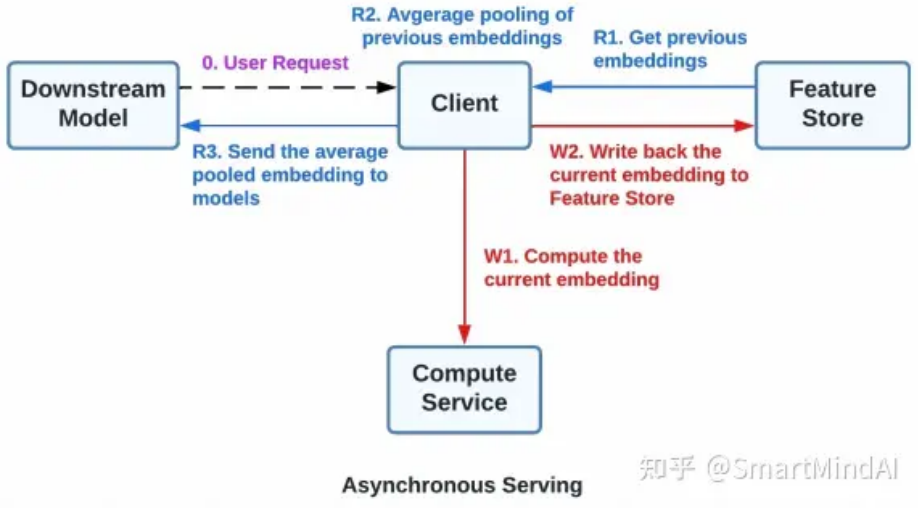
知乎

$$\mathcal{L} = -\frac{1}{N}\sum_{i=1}^{N}\sum_{t=1}^{T} w_t \left(y_{ti}\log(\hat{y}_{ti}) + (1-y_{ti})\log(1-\hat{y}_{ti})\right)$$

其中w_t是任务t的权重$t = 1, 2, \ldots T$，代表最后损失中的其重要性。$y_{ti} \in \{0, 1\}$是任务t中样本i的标签。$\hat{y}_{ti}$是模型在任务t中对样本i的预测值。N是样本的数量。

Online Serving System: SOAP

与相对稳定的项目特征形成对比，SUM用户塔内部的用户特征变动频繁。这些特征中，大多数为类别型数据，面临着新用户ID的引入和现有用户ID语义的转变，这些变化对基于离线服务的解决方案提出了挑战。离线服务的解决方案，包括批处理推理和事件触发推理，都会导致嵌入过时，进而影响下游模型的性能。因此，为了更新用户特征的表示，我们采取了在线推理方式，在每次用户请求时执行模型的计算。 然而，对于在线推理，主要的挑战在于极短的延迟预算:每接收到用户请求后的30ms内需要完成推理过程，并将用户嵌入传递给下游的排序模型。这种严格的时效要求限制了用户模型的复杂性，进而影响了SUM用户嵌入的表达能力。为了解决这些问题，设计了一种能够精准响应用户请求同时满足低延迟要求的在线推理策略，以平衡性能与实时性的需求。



针对上述挑战，我们创新设计了SUM在线异步平台(SUM Online Asynchronous Platform, SOAP)，引入了适用于在线SUM推理的异步服务模式。如图所示，当下游模型接收到用户请求时，SOAP计算中心⁺实时生成当前用户的最新嵌入，随后存储供后续访问。同时，客户端⁺即刻从存储中检索用户的前期嵌入，发送至下游模型进行处理，无需等待中心的计算完成。这种设计将特征的写入与读取操作分离，使当前嵌入的计算与更新作为异步操作执行。分离写入与读取路径降低了延迟影响，理论上可支持长时间推理延迟的复杂用户模型，尽管实际中推理延迟仍会影响服务性能。SOAP通过提高实时性能和优化成本效益，为在线场景提供了灵活性。为了进一步优化推理性能，SOAP专门服务SUM用户模型的用户塔，只生成并提供用户嵌入，而非整个模型的输出。

Productionization

Model Training

SUM用户模型通过离线的循环训练⁺方式构建，允许模型既保留历史趋势，又逐步适应用户的动态偏好。这种周期性的快照更新与在线推理的结合，确保了SUM可以连续、及时地为下游模型提供最新版本的用户嵌入信息。这种方式不仅维持了模型对于历史数据的理解，而且增强了其对当前和未来用户行为模式的预测能力。

Embedding Distribution Shift

策略1:在将嵌入用于推理之前，确保下游模型已经接触到了嵌入的新版本。此策略需调整模型的训练、提供服务和特征记录流程，引入更多复杂性。 策略2:通过减少分布偏移⁺来追求新版与前版的相似性，方法包括正则化、知识蒸馏⁺以及后处理中的平均池化。 我们的建议采用最近两个缓存用户嵌入与当前计算结果的平均池化作为最终的观点。平均池化在广告排名和推荐系统领域已有广泛

知乎

Feature Storage Optimization

在实际应用中，我们通常设定$K = 2$和$D = 96$。这意味着每个用户模型会产生两个SUM用户嵌入，每嵌入的维度大小为96。通过分析不同K的设定，我们发现使用$K = 2$能够在性能提升和特征存储效率之间取得良好平衡。 为了进一步优化在线和离线的存储空间使用，我们实施了用户嵌入的格式量化，即将两个用户嵌入从浮点32位（即fp32）格式转化为浮点16位（即fp16）格式。我们对这种量化策略进行评估后发现，这并未对下游模型的性能产生负面影响，从而实现了有效的存储优化，提高了整体系统效率。

Distributed Inference

为了优化内存使用并进一步提升性能，我们引入了分布式推理✦（DI）技术来处理用户塔任务。通过这种方式，原本的在线推理负载被有效地分配到多个计算节点上，提高了系统的整体效率和响应速度，值得注意的是，这种分布式处理对于提升用户模型的性能表现尤为关键。

Experiments

Dataset

该研究中，所有的实验均基于工业级数据集实施，我们未采用公共数据集，原因在于现有公共数据集与内部模型间存在显著差异，这种差异使得它们不适合直接应用于我们的服务系统中的下游实验。

Evaluation Metric

我们采用了标准化熵(NE)作为离线评估中衡量模型预测准确性的指标，其定义如公式所示。该标准化熵测度了模型在预测用户何时对广告产生点击行为上的准确度。具体而言，这相当于将每千次展示的平均对数损失与预测每次展示都为平均点击率(CTR)时的平均对数损失进行比较，数值越低表示预测性能越佳。

$$NE = \frac{\frac{-1}{N}\sum_{i=1}^{n}(y_i \log(p_i)+(1-y_i)\log(1-p_i))}{-(p\log(p)+(1-p)\log(1-p))}$$

标准化熵(NE)用公式定义，作为离线评估时衡量模型预测性能的指标，具体来说，它量化了模型在预测每个展示被用户点击时间上的准确性。通过比较每千次展示的平均对数损失与预测每次展示均是平均点击率（CTR）时的损失，NE体现了预测准确度，数值越低表示模型性能越好。

为了评估特征的相对重要性，我们采用特征重要性排序(FI)。在用户模型生成的SUM嵌入被用于各种生产模型作为输入后，通过比较SUM嵌入与其他模型可用特征的FI排名，我们可以深入了解SUM用户嵌入在评估模型✦中的价值和影响，进而识别出它们在预测决策中的关键性。

FB CTR SUM User Model

在实践应用中，为了适应不同的数据集和任务需求，我们维护了一系列的SUM用户模型。以Facebook点击率(SUM)用户模型为例，该模型采用了Facebook移动应用程序内动态流的训练数据集，数据规模达到每日约60亿条，专用于点击率预测任务。模型结构包含一个包含四个串联交互模块的用户塔，其中特征提取✦器运用了全连接层、注意力压缩和MLP-Mixer技术。此模型处理的用户侧特征总数约为600个稀疏特征和1000个密集特征，用户塔的总大小为160GB，推理过程所需的浮点运算✦（FLOPs）大约为390M。

Downstream Offline Results

CTR	Mobile feed	-0.19	2
CTR	Home feed	-0.12	2
CTR	Mobile story	-0.16	1
CTR	Dynamic ads	-0.12	4
CTR	Marketplace	-0.08	3
CTR	Channel watch	-0.12	2
CTR	Instagram story	-0.05	15
CTR	Messenger inbox	-0.35	1
Post-imp offsite CVR	Mobile feed	-0.09	16
Post-click offsite CVR	Mobile feed	-0.05	13
MAI CVR	Mobile feed	-0.04	24
Inline CVR	Mobile feed	-0.09	5

表格展示了对Facebook点击率(SUM)模型的离线评估指标标准化熵(NE)在不同下游任务中的应用效果。结果显示，SUM嵌入在广告点击率✦预测、原位转换预测、移动应用安装预测以及离线转换预测等跨域任务中均能显著提升NE值，这凸显了SUM模型的优秀代表性和广泛适用性。预期在点击率预测任务中的收益更高，原因是在这些任务上模型是以点击率数据为基础进行训练的。而对于Instagram模型，取得了较小的增益，这提示了开发针对Instagram特性的SUM模型来弥补Facebook与Instagram之间的领域差异的重要性。值得注目的是，在像Messenger收件箱这样模型规模较小且特征数量有限的情况下，共享大规模用户表示所带来的利益更加明显。有趣的是，增加SUM嵌入对下游模型的训练效率或其他基础设施指标几乎没有负面影响，因为引入的嵌入特征是紧密表示的，无需进行计算密集型的稀疏特征嵌入查找操作。

Online Performance

SUM在Meta公司的数百个营销应用中被广泛采用，对提升广告效果产生了显著影响，覆盖了Instagram（IG）、Facebook广告管理器（FAM）、商店广告（Shop Ads）及Messenger广告等多类平台。通过在线A/B测试的验证，SUM带来了总广告指标2.67%的提升，这在内部统计标准✦下被认为是显著的（达到0.2%的提升被视为统计显著）。特别值得称赞的是，尽管收获了如此显著的性能增加，SUM的部署并未导致服务容量的显着增加，相反，与引入等同复杂性模型至每个下游应用相比，仅引发了15.3%的服务容量增长，展现了其高效路径与能力优化的价值。

Async Serving

为了评估不同服务解决方案的性能，我们进行了实验对比。四个服务解决方案包括冻结用户模型、离线批处理、在线实时服务和在线异步服务。

1. 冻结用户模型:采取一次性的用户模型训练，利用原始快照不断评估新数据并生成用户嵌入。这种方法的使用取决于输入特征的稳定性，限制了特征空间✦，可能无法充分挖掘用户模型的潜能。

2. 离线批处理:经过初次训练后，每日进行模型重训，使用在特定时间点上训练得到的快照评估当前数据并生成用户嵌入。参数的时滞通常在一天到三天，反映了下游模型数据处理流程的差异。

3. 在线实时服务:用户请求即时响应，计算服务提供当前嵌入，特征存储提供历史嵌入，通过平均池化作为最终当前嵌入传递给下游模型。计算服务未能在规定时间完成推理时被视为失败。

4. 在线异步服务:详细解释见第4节。

知乎

Baseline, no SUM	0
Baseline + SUM (Frozen, 1-month staleness)	-0.034
Baseline + SUM (Offline Batch, 1-day staleness)	-0.094
Baseline + SUM (Online Realtime Serving)	-0.141
Baseline + SUM (Online Async Serving)	-0.126

通过一个试验证实:在对紧凑型（20M推理FLOPs）用户模型进行转换后，服务从实时更新到异步服务的损失仅为10%，显著低于从实时转换到离线批处理方案所见的损失。此发现强调了异步服务的优势，并增强了采用更复杂用户模型的可能性。

Embedding Distribution Shift

为了揭示嵌入分布偏移的影响，我们执行了一组离线实验，采用离线批处理模式来训练和更新SUM用户模型。在这个过程中，模型每天都会更新一次，从而得到当日的移动快照，用以生成用户嵌入。我们定义了两个嵌入向量$Embedding0$和$Embedding1$，通过分析这些嵌入之间的连续日期上的余弦相似性+和L2范数变化，以量化它们之间的异动。实验结果汇总在表中，显示了这些变化的平均数值。此外，实验还揭示了$Embedding1$较之$Embedding0$，能够为下游模型带来更大的训练NE增益，而$Embedding0$在$Embedding1$的基础上则还能额外增加一些增益。尽管我们对为何一个嵌入比另一个更稳定，以及为何更稳定的嵌入能够在训练中提供更大的增益这一问题感兴趣，但并非此次研究的重点。这一发现可能为未来的工作开辟了探究的途径，特别是针对提高模型性能的潜在改进方向。

Model setting	Train NE (%)	Eval NE (%)
Baseline	0	0
Baseline + SUM w/o AP	-0.081	-0.014
Baseline + SUM w/ AP	**-0.109**	**-0.127**

根据表的数据显示，当未应用平均池化技术时，我们观察到了训练过程中的良好训练NE增益，这揭示了下游模型能够适应未经平均池化的嵌入分布偏移的情况。然而，与之形成对比的是评估NE增益较为有限，这是可以预见到的现象:在这里，嵌入特征由新生成的用户模型快照提供，在模型训练期间，这些新版本的嵌入未曾接触或学习过。这种突如其来的变化导致了评估过程中的性能下降，即降低了评估NE值。

进一步地，通过在3个嵌入上实施平均池化操作，我们发现能够显著减少嵌入间的偏移问题并提升整体性能，特别是能显著提升评估NE的性能表现。这一方法有效平滑了嵌入的变化，提供了更加稳定和一致的特征，从而对下游模型的评估过程产生了积极的影响。

原文《Scaling User Modeling: Large-scale Online User Representations for Ads Personalization in Meta》

发布于 2024-06-28 11:04 · IP 属地北京

Meta (Facebook) 个性化推荐 推荐系统实现