alimama · tech

2023

ALIMAMA
TECHNICAL
YEARBOOK

PRODUCED BY ALIMAMA TECH



ALIMAMA TECHNICAL YEARBOOK

2023

阿里妈妈技术年刊

广告算法与工程实践精选

阿里妈妈技术出品

序

阿里妈妈成立于 2007 年，是淘天集团商业数智营销中台。秉承着“让每一份经营都算数”的使命，阿里妈妈技术团队深耕 AI 在互联网广告领域的探索和大规模应用，并通过技术创新驱动业务高速成长，让商业营销更简单高效。

2021 年 5 月，我们开始通过「阿里妈妈技术」微信公众号持续分享我们的技术实践与经验，覆盖广告算法实践、AI 平台及工程引擎、智能创意、风控、数据科学等多个方向。

每年此时，我们都会整理过去一年颇具表性和创新性的工作沉淀制作成册。《2023 阿里妈妈技术年刊》涵盖机制策略、召回匹配、预估模型、智能创意、算法工程 / 引擎 / 系统建设等内容，这些工作有的已为业务创造实际收益，有的是一些常见问题的新解法，希望可以为相关领域的同学带来一些新的思路。

期待明年此时，每位朋友都有新的收获，而我们也带着更多探索实践来与大家分享交流 ~

如果对这本电子书有想要探讨的问题，或有更好的建议，也欢迎通过「阿里妈妈技术」微信公众号与我们联系。

最后，祝大家新春快乐 ~ 祝福如初，愿不负追求与热爱，万事尽可期待！

本书共 435 页，全部内容近 48 万字。如果觉得还不错，别忘了分享给身边的朋友 ~

阿里妈妈技术团队

欢迎关注「阿里妈妈技术」微信公众号

目录

机制策略 1

迈步从头越 – 阿里妈妈广告智能决策技术（自动出价 & 拍卖机制）的演进之路 1

Bidding 模型训练新范式：阿里妈妈生成式出价模型（AIGB）详解 26

万字长文，漫谈广告技术中的拍卖机制设计（经典篇） 36

PerBid：在线广告个性化自动出价框架 55

Auction Design in the Auto-bidding World 系列一：面向异质目标函数广告主的拍卖机制设计 69

自动出价下机制设计系列（二）：面向私有约束的激励兼容机制设计 79

增广拍卖——二跳页下的拍卖机制探索 89

Score-Weighted VCG：考虑外部性的智能拍卖机制设计 99

合约广告中端到端流量预估与库存分配 108

强化学习在广告延迟曝光情形下的保量策略中的应用 123

MiRO：面向对抗环境下约束竞价的策略优化框架 134

预估模型 142

排序和准度联合优化：一种基于混合生成 / 判别式建模的方案 142

转化率预估新思路：基于历史数据复用的大促转化率精准预估 154

基于特征自适应的多场景预估建模 174

HC^2：基于混合对比学习的多场景广告预估建模 183

AdaSparse: 自适应稀疏网络的多场景 CTR 预估建模 193

贝叶斯分层模型应用之直播场景打分校准 203

召回匹配 216

代码开源！阿里妈妈展示广告 Match 底层技术架构最新进展 216

BOMGraph：基于统一图神经网络的电商多场景召回方法 220

CC-GNN：基于内容协同图神经网络的电商召回方法 229

RGIB：对抗双边图噪声的鲁棒图学习 241

Memorization Discrepancy：利用模型动态信息发现累积性注毒攻击 251

智能创意 262

ACM MM'23 | 4 篇论文解析阿里妈妈广告创意算法最新进展 262

上下文驱动的图上文案生成 267

基于无监督域自适应方法的海报布局生成 273

基于内容融合的字体生成方法 278

化繁为简，精工细作——阿里妈妈直播智能剪辑技术详解 286

视频分割新范式：视频感兴趣物体实例分割 VOIS 297

风控技术 305

阿里妈妈内容风控模型预估引擎的探索和建设 305

大模型时代的阿里妈妈内容风控基础服务体系建设 323

隐私计算 344

广告营销场景下的隐私计算实践：阿里妈妈营销隐私计算平台 SDH 344

阿里妈妈营销隐私计算平台 SDH 在公用云的落地实践 353

算法工程 / 引擎 / 系统建设 363

积沙成塔——阿里妈妈动态算力技术的新演进与展望 363

阿里妈妈智能诊断工程能力建设 380

广告深度学习计算：向量召回索引的演进以及工程实现 390

Dolphin：面向营销场景的超融合多模智能引擎 398

阿里妈妈 Dolphin 智能计算引擎基于 Flink+Hologres 实践 414

Dolphin Streaming 实时计算，助力商家端算法第二增长曲线 424

机制策略

迈步从头越 – 阿里妈妈广告智能决策技术（自动出价 & 拍卖机制）的演进之路

作者：妙临、霁光、玺羽

导读

随着智能化营销产品和机器学习的发展，阿里妈妈将深度学习和强化学习等 AI 技术越来越多地应用到广告智能决策领域。在阿里妈妈技术同学们的持续努力下，我们推动了业界广告决策智能技术的代际革新。本文结合时代发展的视角分享了阿里妈妈广告智能决策技术的演化过程，希望能给从事相关工作的朋友带来一些新思路。

1. 前言

在线广告对于大多数同学来说是一个既熟悉又陌生的技术领域。「搜广推」、「搜推广」等各种组合耳熟能详，但广告和搜索推荐有本质区别：**广告解决的是“媒体 – 广告平台 – 广告主”等多方优化问题**，其中媒体在保证用户体验的前提下实现商业化收入，广告主的诉求是通过出价尽可能优化营销目标，广告平台则在满足这两方需求的基础上促进广告生态的长期繁荣。

广告智能决策技术在这之中起到了关键性的作用，如图 1 所示，它需要解决如下问题在内的一系列智能决策问题：1. 为广告主设计并实现自动出价策略，提升广告投放效果；2. 为媒体设计智能拍卖机制来保证广告生态系统的繁荣和健康。

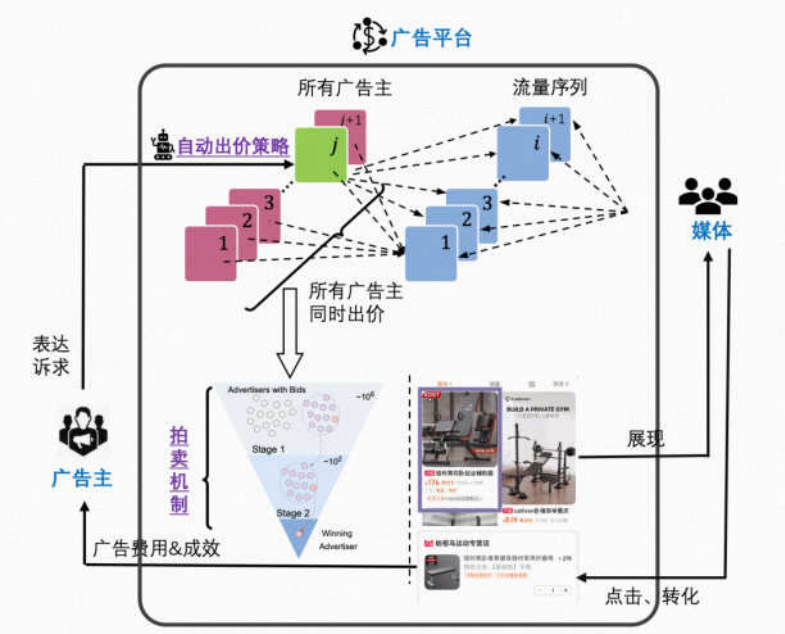


图 1　广告智能决策通过自动出价和拍卖机制等方式实现多方优化

随着智能化营销产品和机器学习的发展，阿里妈妈将深度学习和强化学习等 AI 技术越来越多地应用到广告智能决策领域，如 RL-based Bidding（基于强化学习的出价）帮助广告主显著提升广告营销效果，Learning-based Auction Design（基于学习的拍卖机制设计）使得多方利益的统筹优化更加高效。我们追根溯源，结合时代发展的视角重新审视广告智能决策技术的演化过程，本文将以阿里妈妈广告智能决策技术的演进为例，分享我们工作和思考。也希望能以此来抛砖引玉，和大家一块探讨。

2. 持续突破的自动出价决策技术

广告平台吸引广告主持续投放的核心在于给他们带来更大的投放价值，典型的例子就是自动化的出价产品一经推出便深受广告主的喜爱并持续的投入预算。在电商场景下，我们不断地探索流量的多元化价值，设计更能贴近营销本质的自动出价产品，广告主只需要简单的设置就能清晰的表达营销诉求。



图 2　出价产品逐步的智能化 & 自动化，广告主只需要简单的设置即可清晰的表达出营销诉求

极简产品背后则是强大的自动出价策略支撑，其基于海量数据自动学习好的广告投放模式，以提升给定流量价值下的优化能力。考虑到广告优化目标、预算和成本约束，自动出价可以统一表示为带约束的竞价优化问题。

$$
\begin{aligned}
&\max_{xi} \sum_i v_i x_i \\
&s.t. \sum_i c_i x_i \le B \\
&\frac{\sum_i c_{ij} x_i}{\sum_i p_{ij} x_i} \le k_j, \forall j \\
&x_i \le 1, \forall i \\
&x_i \ge 0, \forall i
\end{aligned}
$$

其中 B 为广告主的预算，k_j 为成本约束，该问题就是要对所有参竞的流量进行报价，以最大化竞得流量上的价值总和。如果已经提前知道要参竞流量集合的全部信息，包括能够触达的每条流量的价值 v_i 和成本 c_i 等，那么可以通过线性规划（LP）方法来求得最优解。然而在线广告环境的动态变化以及每天到访用户的随机性，竞争流量集合很难被准确的预测出来。因此常规方法并不完全适用，需要构建能够适应动态环境的自动出价算法。

对竞价环境做一定的假设（比如拍卖机制为单坑下的 GSP，且流量竞得价格已知），通过拉格朗日变换构造最优出价公式，将原问题转化为最优出价参数的寻优问题 [9]：

$$
b_i^* = w_0^* v_i - \sum_j w_j^* (q_{i,j}(1 - 1_{CR_j}) - k_j p_{ij})
$$

对于每一条到来的流量按照此公式进行出价，其中 $v_i, q_{i,j}$ 为在线流量竞价时可获得的流量信息，w_0, w_j 为要求解的参数。而参数并不能一成不变，需要根据环境的动态变

化不断调整。参竞流量的分布会随时间发生变化，广告主也会根据自己的经营情况调整营销设置，前序的投放效果会影响到后续的投放策略。因此，出价参数的求解本质上是动态环境下的序列决策问题。

2.1 主线：从跟随到引领，迈向更强的序列决策技术

如何研发更先进的算法提升决策能力是自动出价策略发展的主线，我们参考了业界大量公开的正式文献，并结合阿里妈妈自身的技术发展，勾勒出自动出价策略的发展演进脉络。

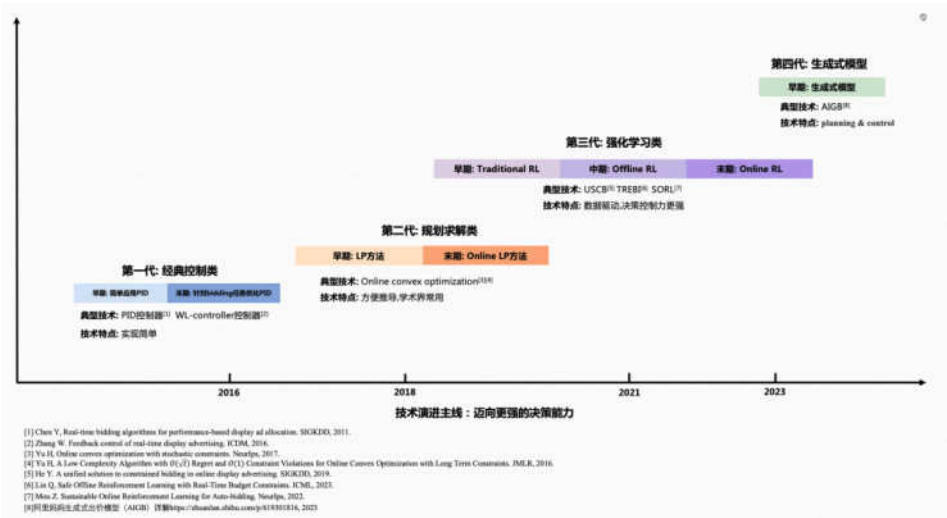


图 3　自动出价策略的演进主线：迈向更强的决策能力

整体可以划分为 4 个阶段：

第一代：经典控制类

- 把效果最大化的优化问题间接转化为预算消耗的控制问题。基于业务数据计算消耗曲线，控制预算尽可能按照设定的曲线来消耗。PID[1] 及相关改进 [2][10] 是这一阶段常用的控制算法。当竞价流量价值分布稳定的情况下，这类算法能基本满足业务上线之初的效果优化。

第二代：规划求解类

- 相比于第一代，规划求解类（LP）算法直接面向目标最大化优问题来进行求解。可基于前一天的参竞流量来预测当前未来流量集合，从而求解出价参数。自动

出价问题根据当前已投放的数据变成新的子问题，因此可多次持续的用该方法进行求解，即 Online LP[3][4]。这类方法依赖对未来参竞流量的精准预估，因此在实际场景落地时需要在未来流量的质和量的预测上做较多的工作。

第三代：强化学习类

- 现实环境中在线竞价环境是非常复杂且动态变化的，未来的流量集合也是难以精准预测的，要统筹整个预算周期投放才能最大化效果。作为典型的序列决策问题，第三阶段用强化学习类方法来优化自动出价策略。其迭代过程从早期的经典强化学习方法落地 [5][6][8][9]，到进一步基于 Offline RL 方法逼近「在线真实环境的数据分布」[9]，再到末期贴近问题本质基于 Online RL 方法实现和真实竞价环境的交互学习 [13]。

第四代：生成模型类

- 以 ChatGPT 为代表的生成式大模型以汹涌澎湃之势到来，在多个领域都表现出令人惊艳的效果。新的技术理念和技术范式可能会给自动出价算法带来革命性的升级。阿里妈妈技术团队提前布局，以智能营销决策大模型 AIGA（AI Generated Action）为核心重塑了广告智能营销的技术体系，并衍生出以 AIGB（AI Generated Bidding）[14] 为代表的自动出价策略。

为了让大家有更好地理解，我们以阿里妈妈的实践为基础，重点讲述下强化学习在工业界的落地以及对生成式模型的探索。

2.1.1 强化学习在自动出价场景的大规模应用实践

跟随：不断学习、曲折摸索

作为典型的序列决策问题，使用强化学习（RL）是很容易想到的事情，但其在工业界的落地之路却是充满曲折和艰辛的。最初学术界 [8] 做了一些探索，在请求粒度进行建模，基于 Model-based RL 方法训练出价智能体（Agent），并在请求维度进行决策。如竞得该 PV，竞价系统返回该请求的价值，否则返回 0，同时转移到下一个状态。这种建模方法应用到工业界遇到了很多挑战，主要原因在于工业界参竞流量巨大，请求粒度的建模所需的存储空间巨大；转化信息的稀疏性以及延迟反馈等问题也给状态构造和 Reward 设计带来很大的挑战。为使得 RL 方法能够真正落地，需要解决这几个问题：

「**MDP 是什么?**」由于用户到来的随机性，参竞的流量之间其实并不存在明显的马尔

可夫转移特性，那么状态转移是什么呢？让我们再审视下出价公式，其包含两部分：流量价值和出价参数。其中流量价值来自于请求粒度，出价参数为对当前流量的出价激进程度，而激进程度是根据广告主当前的投放状态来决定的。一种可行的设计是将广告的投放信息按照时间段进行聚合组成状态，上一时刻的投放策略会影响到广告主的投放效果，并构成新一时刻的状态信息，因此按照时间段聚合的广告主投放信息存在马尔可夫转移特性。而且这种设计还可以把问题变成固定步长的出价参数决策，给实际场景中需要做的日志回流、Reward 收集、状态计算等提供了时间空间。典型的工作 [5][6][7][8][9][12] 基本上都是采用了这样的设计理念。

「**Reward 如何设计?**」Reward 设计是 RL 的灵魂。出价策略的 Reward 设计需要让策略学习如何对数亿计流量出价，以最大化竞得流量下的价值总和。如果 Reward 只是价值总和的话，就容易使得策略盲目追求好流量，预算早早花光或者成本超限，因此还需要引导策略在约束下追求更有性价比的流量。另外，自动出价是终点反馈，即直到投放周期结束才能计算出完整的投放效果；且转化等信号不仅稀疏，还存在较长时间的回收延迟。因此我们需要精巧设计 Reward 让其能够指导每一次的决策动作。实践下来建立决策动作和最终结果的关系至关重要，比如 [9] 在模拟环境中保持当前的最优参数，并一直持续到终点，从而获取到最终的效果，以此来为决策动作设置较为精准的 Reward。另外，在实际业务中，为了能够帮助模型更好的收敛，往往也会把业务经验融入到 Reward 设计中。

「**如何训练?**」强化学习本质是一个 Trail-and-Error 的算法，需要和环境进行交互收集到当前策略的反馈，并不断探索新的决策空间进一步更新迭代策略。但在工业界，由于广告主投放周期的设置，一个完整的交互过程在现实时间刻度上通常为一天。经典的 RL 算法要训练好一般要经历上万次的交互过程，这在现实系统中很难接受。在实践中，通常构造一个模拟竞价环境用于 RL 模型的训练，这样就摆脱现实时空的约束提升模型训练效率。当然在线竞价环境非常复杂，如何在训练效率和训练效果之间平衡是构造模拟环境中需要着重考虑的事情。这种训练模式，也一般称之为 **Simulation RL-based Bidding**（简称 SRLB），其流程如下图所示：

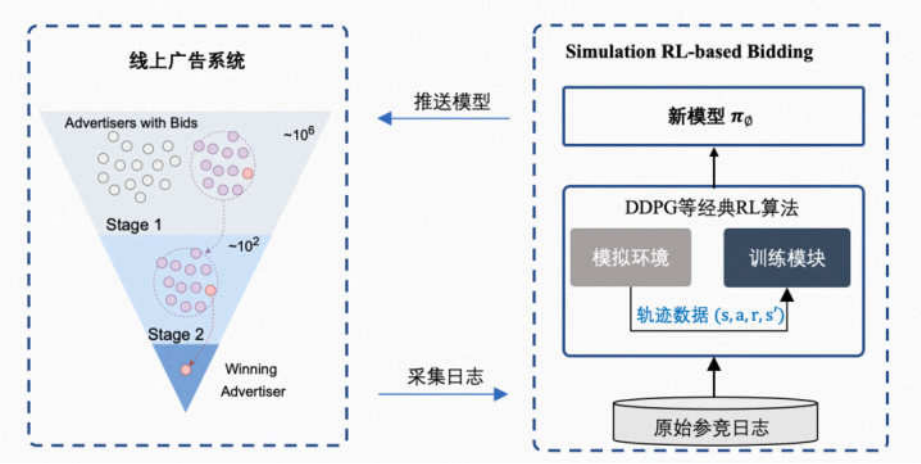


图 4　Simulation RL-based Bidding（SRLB）训练模式

基于 SRLB 训练模式，我们实现了强化学习类算法在工业界场景的大规模落地。根据我们的调研，在搜广推领域，RL 的大规模落地应用较为少见。

创新：立足业务、推陈出新

随着出价策略不断的升级迭代，“模拟环境和在线环境的差异”逐渐成为了效果进一步提升的约束。为了方便构造，模拟环境一般采用单坑 GSP 来进行分配和扣费且假设每条流量有固定的获胜价格（Winning Price）。但这种假设过于简单，尤其是当广告展现的样式越来越丰富，广告的坑位的个数和位置都在动态变化，且 Learning-based 拍卖机制也越来约复杂，使得模拟环境和在线实际环境差异越来越大。基于 Simulation RL-based Bidding 模式训练的模型在线上应用过程中会因环境变化而偏离最优策略，导致线上效果受到损失。模拟环境也可以跟随线上环境不断升级，但这种方式成本较高难度也大。因此，我们期待能够找到一种不依赖模拟环境，能够对标在线真实环境学习的模式，以使得训练出来的 Bidding 模型能够感知到真实竞价环境从而提升出价效果。结合业务需求并参考了 RL 领域的发展，我们先后调研了模仿学习、Batch RL、Offline RL 等优化方案，并提出的如下的 **Offline RL-based Bidding** 迭代范式，期望能够以尽可能小的代价的逼近线上真实的样本分布。

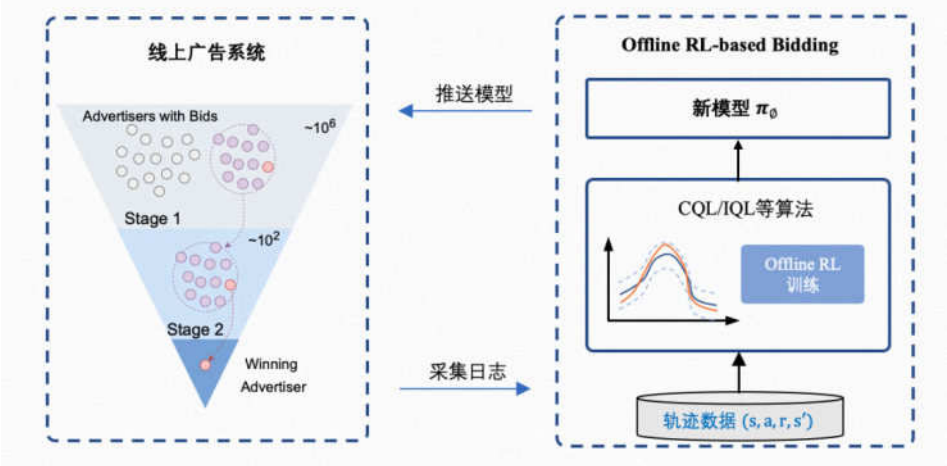


图 5　Offline RL-based Bidding 训练模式，与 SRLB 模式差异主要在训练数据来源和训练方式

在这个范式下，直接基于线上决策过程的日志，拟合 reward 与出价动作之间的相关性，从而避免模拟样本产生的分布偏差。尽管使用真实决策样本训练模型更加合理，但在实践中往往容易产生策略坍塌现象。核心原因就是线上样本不能做到充分探索，对样本空间外的动作价值无法正确估计，在贝尔曼方程迭代下不断的高估。对于这一问题，我们可以假设一个动作所对应的数据密度越大，支撑越强，则预估越准确度越大，反之则越小。基于这一假设，参考 CQL[21] 的思想，构建一种考虑数据支撑度的 RL 模型，利用数据密度对价值网络估值进行惩罚。这一方法可以显著改善动作高估问题，有效解决 OOD 问题导致的策略坍塌，从而使得 Offline RL-based 能够部署到线上并取得显著的效果提升。后续我们又对这个方法做了改进，借鉴了 IQL[22]（Implicit Q learning）中的 In-sample learning 思路，引入期望分位数回归，基于已有的数据集来估计价值网络，相比于 CQL，能提升模型训练和效果提升的稳定性。

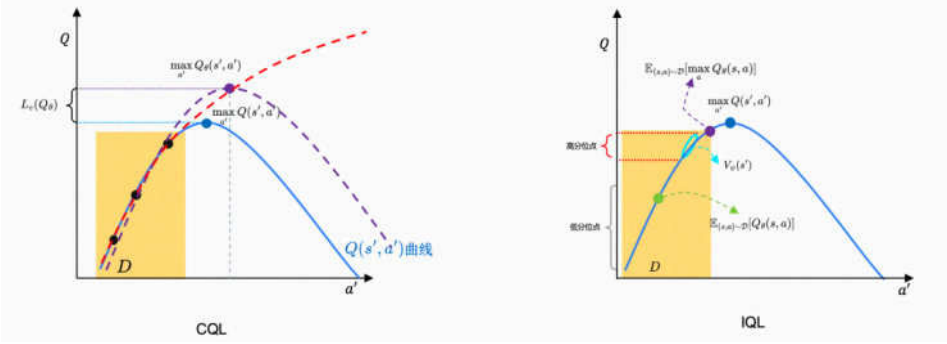


图 6　从 CQL 到 IQL，Offline RL-based Bidding 中训练算法的迭代

总结下来，在这一阶段我们基于业务中遇到的实际问题，并充分借鉴业界思路，推陈出新。Offline RL-based Bidding 通过真实的决策数据训练出价策略，比基于模拟环境训练模式（SRLB）能够更好的逼近「线上真实环境的数据分布」。

突破：破解难题、剑走偏锋

让我们再重新审视 RL-based Bidding 迭代历程，该问题理想情况可以通过「与线上真实环境进行交互并学习」的方式求解，但广告投放系统交互成本较高，与线上环境交互所需要的漫长「训练时间成本」和在线上探索过程中可能需要遭受的「效果损失成本」，让我们在早期选择了 Simulation RL-based Bidding 范式，随后为解决这种范式下存在的环境不一致的问题，引入了 Offline RL-based Bidding 范式。

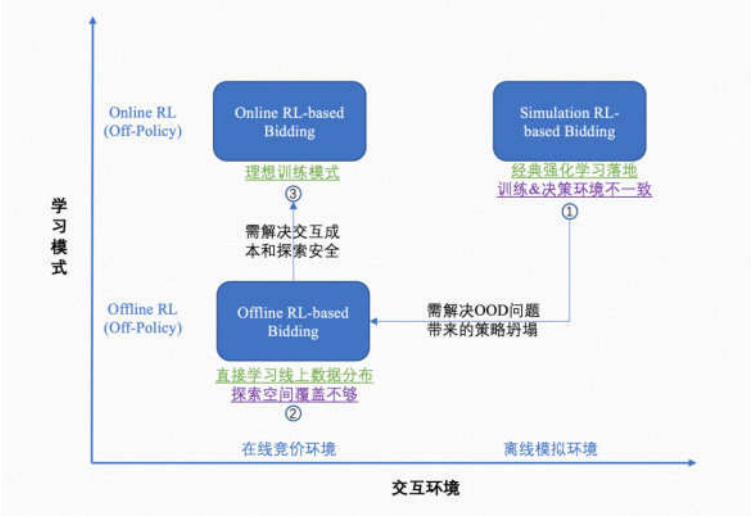


图 7　重新审视 RL-based Bidding 发展脉络

为了能够进一步突破效果优化的天花板，我们需要找到一种新的 Bidding 模型训练范式：能够不断的和线上进行交互探索新的决策空间且尽可能减少因探索带来的效果损失。还能够在融合了多种策略的样本中进行有效学习。即控制「训练时间成本」和「效果损失成本」下的 **Online RL-based Bidding** 迭代范式，如下图所示：

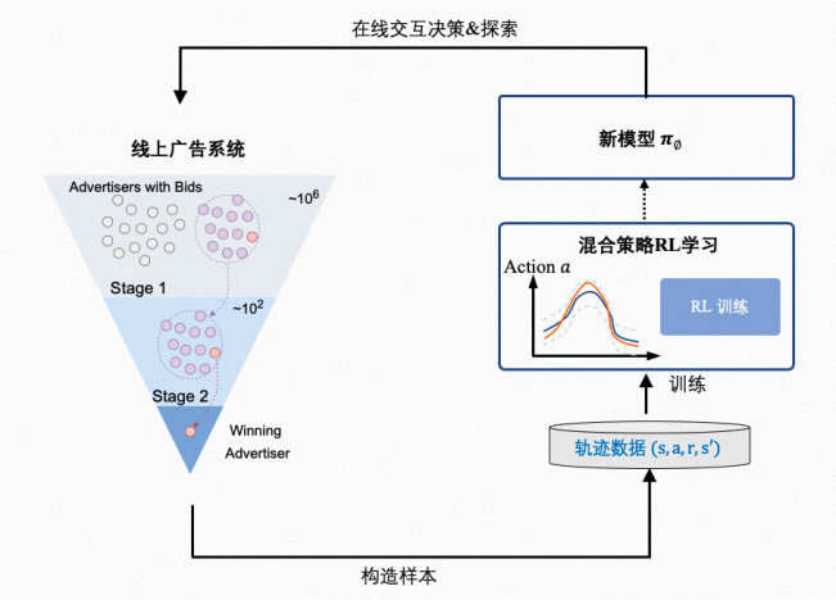


图 8　Online RL−based Bidding 训练模式，与前两种模式的差别在于能够和环境进行直接交互学习

[13] 提出了可持续在线强化学习（SORL），与在线环境交互的方式训练自动出价策略，较好解决了环境不一致问题。SORL 框架包含探索和训练两部分算法，基于 Q 函数的 Lipschitz 光滑特性设计了探索的安全域，并提出了一个安全高效的探索算法用于在线收集数据；另外提出了 V−CQL 算法用于利用收集到的数据进行离线训练，V−CQL 算法通过优化训练过程中 Q 函数的形态，减小不同随机种子下训练策略表现的方差，从而提高了训练的稳定性。

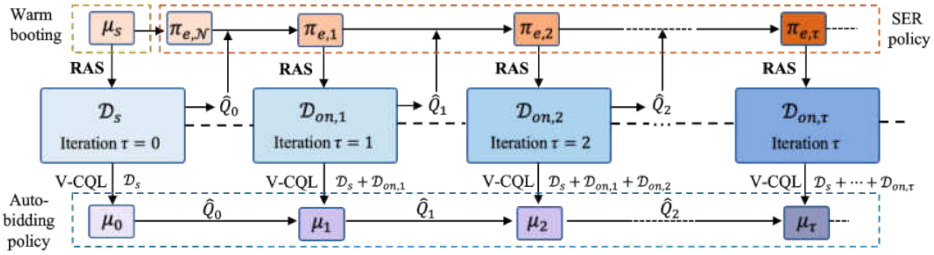


图 9　SORL 的训练模式

在这一阶段中，不断思考问题本质，提出可行方案从而使得和在线环境进行交互训练学习成为可能。

2.1.2 引领生成式 Bidding 的新时代（AIGB）

ChatGPT 为代表的生成式大模型以汹涌澎湃之势到来。一方面，新的用户交互模式会孕育新的商业机会，给自动出价的产品带来巨大改变；另一方面，新的技术理念和技术范式也会给自动出价策略带来革命性的升级。我们在思考生成式模型能够给自动出价策略带来什么？从技术原理上来看，RL 类方法基于时序差分学习决策动作好坏，在自动出价这种长序列决策场景下会有训练误差累积过多的问题。因此，我们提出了一种基于生成式模型构造的出价策略优化方案（AIGB – AI Generative Bidding）[14]。与强化学习的视角不同，如图 9 所示，AIGB 直接关联决策轨迹和回报信息，能够避免训练累积，更适合长序列决策场景。

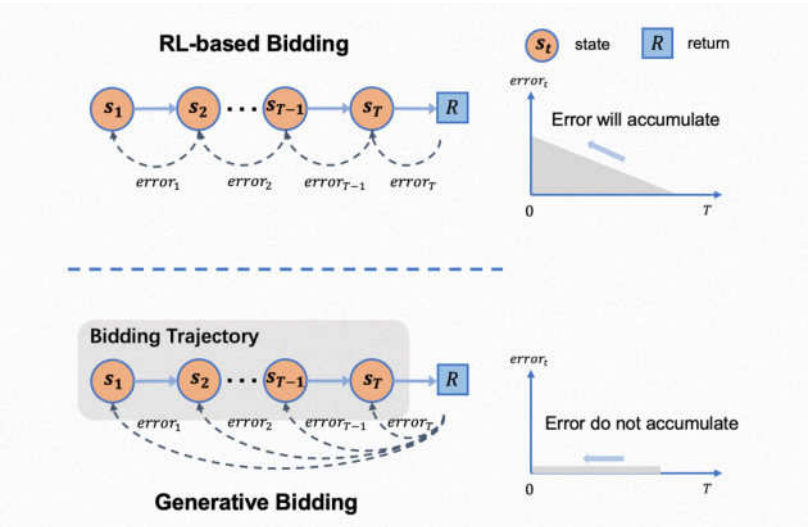


图 10　Generative Bidding 相比 RL-based Bidding 模式能够避免训练误差累积，更适合长序列决策场景

从生成式模型的角度来看，我们可以将出价、优化目标和约束等具备相关性的指标视为一个联合概率分布，从而将出价问题转化为条件分布生成问题。图 10 直观地展示了生成式出价模型的流程：在训练阶段，模型将历史投放轨迹数据作为训练样本，以最大似然估计的方式拟合轨迹数据中的分布特征。这使得模型能够自动学习出价策略、状态间转移概率、优化目标和约束项之间的相关性。在线上推断阶段，生成式模型可以基于约束和优化目标，以符合分布规律的方式输出出价策略。

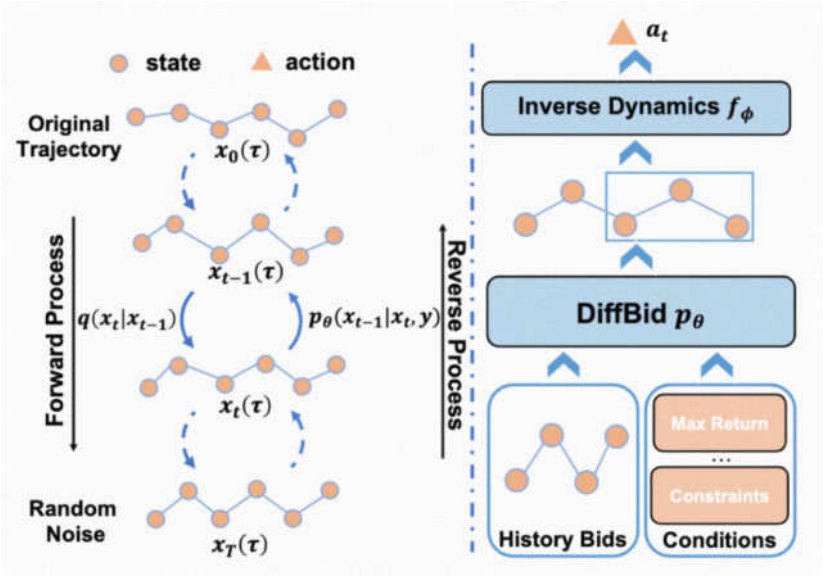


图 11　AIGB 的训练和预测算法

AIGB 基于当前的投放状态信息以及策略生成条件输出未来的投放策略，相比于以往的 RL 策略输出单步 action，AIGB 可以被理解为在规划的基础上进行决策，最大程度地避免分布偏移和策略退化问题，从而更适合长序列决策场景。这一优点有利于在实践中进一步减小出价间隔，提升策略的快速反馈能力。与此同时，基于规划的出价策略也具备更好的可解释性，能够帮助我们更好地进行离线策略评估，方便专家经验与模型深度融合。另外，我们也还在进一步探索，是否可以把竞价领域知识融入到大模型中并帮助出价决策。从「动作判别式」决策 到「轨迹生成式」决策 ，朝着生成式 Bidding 的新时代大踏步迈进！

2.2 副线：百花齐放，更全面的出价决策技术

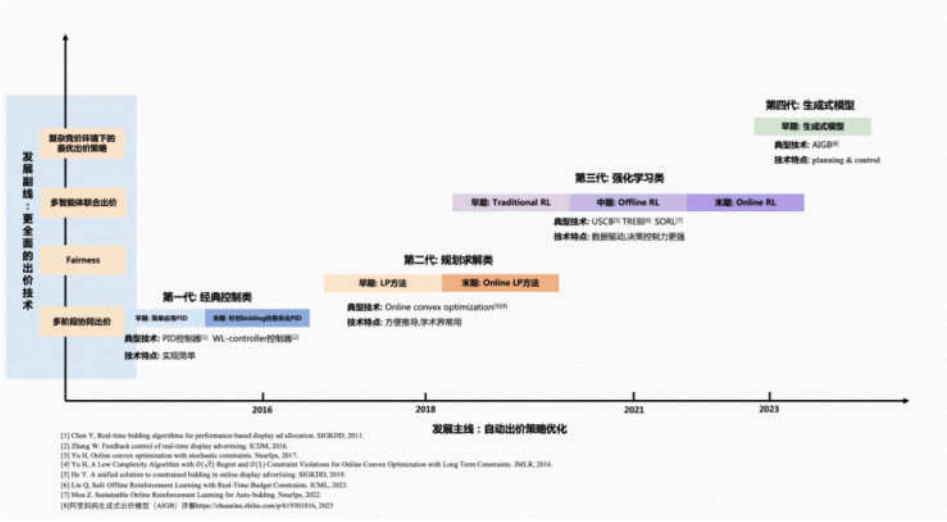


图 12　发展副线：更全面的业务实际场景的特性优化

除了更强的决策能力外，在实际场景中还会针对业务特点做更多的优化，这里介绍 3 个典型的研究技术点：

复杂的竞价环境下的最优出价策略

出价形式化建模依赖对竞价环境的假设，不同的假设下推导出来的出价公式是不同的。以 MaxReturn 计划为例，出价形式为 $bid = \alpha * pcvr$，其中 α 为一个粗粒度与请求无关的参数，在简单竞价环境下 (GSP 单坑下) 这种出价形式是理论最优的。而在实际工业界竞价环境是非常复杂的：多坑、带保留价的机制或其他复杂机制，当前的出价策略并非最优。学术界和工业界针对这一问题提出了不少方法，大概分为 2 类：1）对竞价环境做进一步的假设（比如多坑）推导出闭式解，并进行求解 [18]；2）基于数据驱动的方法，在基础价格上结合当前流量的信息（如 Winning Price）等进行微调，比如 Bid Shading 类方法 [17][19] 。

多智能体联合出价

在线广告本质上是一个多智能体竞价系统。通常情况下每一个自动出价智能体求解一个独立的优化问题，而将其他智能体出价的影响隐式地建模为环境的一部分。这种建模方式忽略了在线广告的动态博弈，即最终的拍卖结果取决于所有智能体的出价，且任一智能体的策略的改变会影响到其他所有智能体的策略。因此若不做协调，则所有

智能体会处于一个无约束状态，进而降低系统的效率。典型的工作包括 [7][11][12] 都是针对线上环境的多智能体问题进行求解，面对线上智能体个数众多（百万级），通过广告主进行聚类等方式，把问题规模降低到可求解的程度。

Fairness

不同行业的广告主在广告投放时面临的竞价环境也是不同的，当前广泛采用的统一出价策略可能使得不同广告主的投放效果存在较大的差异，尤其是对小广告主来说，训练效果会受到大广告主的影响，即“Fairness”问题。典型的工作包括 [16] 将传统的统一出价策略拓展为多个能够感知上下文的策略族，其中每个策略对应一类特定的广告主聚类。这个方法中首先设计了广告计划画像网络用于建模动态的广告投放环境。之后，通过聚类技术将差异化的广告主分为多个类并为每一类广告主设计一个特定的具有上下文感知能力的自动出价策略，从而实现为每个广告主匹配特定的个性化策略。

多阶段协同出价

为平衡行业在线广告的优化性能和响应时间，在线工业场景经常会采用两阶段级联架构。在这种架构下，自动出价策略不仅需要在精竞阶段（第二阶段）进行传统的竞拍，还必须在粗竞阶段（第一阶段）参与竞争才能进入精竞阶段。现有的工作主要集中在精竞阶段的拍卖设计和自动出价策略上，而对粗竞阶段的拍卖机制和自动出价策略研究还不够充分，这部分最主要的挑战在于粗竞阶段的广告量级会比精竞阶段多了近百倍，且自动出价依赖的流量价值预估（如 PCVR）比精竞阶段准度差，因此如何设计更大规模且能够应对不确定性预估值下的出价策略是这个方向主要研究的问题，而且还需要研究两阶段下的拍卖机制设计以引导自动出价正确报价。在这个方向上，我们依赖强大的工程基建能力上线了全链路自动出价策略，显著提升了广告主的投放效果；并设计了适用于两阶段的拍卖机制 [33]。

3. 拍卖机制设计也是一个决策问题

拍卖机制是对竞争性资源的一种高效的市场化分配方式，具有良好博弈性质的拍卖机制在互联网广告场景下可以引导广告主的有序竞争，从而保证竞价生态的稳定和健康。经典拍卖机制如 GSP、VCG 由于其良好的博弈性质以及易于实现的特点使得其在 2002 年前后开始被互联网广告大规模的使用。



图 13　在线广告的拍卖机制的示意图

十几年过去，互联网广告环境已经发生了巨大的改变，与经典静态拍卖机制的假设相比，现在的广告主营销目标多元、策略行为复杂，且机制的优化目标不再是单一的收入或者社会福利，需要将媒体、广告主、广告平台的利益考虑在内统一优化。而在一个智能化的广告系统中，拍卖机制需要根据系统中参与方的行为变化而调整自己的策略行为，即拍卖机制设计也是一个决策问题。因此如何结合互联网海量数据的优势去设计更符合广告主行为模式并贴近业务需求的智能拍卖机制迫在眉睫。

从经济学视角看，最优广告拍卖设计可以看作一个优化决策问题：最大化综合目标（收入、用户体验等），同时需要满足经济学性质保证，最典型的是激励相容性（Incentive Compatibility, IC）和个体理性（Individual Rationality, IR）的约束。IC 要求广告主真实报价总是能最大化其自身效用，而 IR 要求广告主付费不超过其对广告点击的真实估值，这样该机制就可以优化出稳定的效果。

$$\max_{\mathcal{M}} \quad \mathbb{E}_{\mathbf{b}\sim\mathcal{D}}[w_1 \times f_1(\mathbf{b};\mathcal{M}) + \cdots + w_L \times f_L(\mathbf{b};\mathcal{M})] \qquad \text{多目标优化(RPM,CTR,GMV)*}$$

$$\text{s.t.} \quad \textit{Game Equilibrium Constraints,} \qquad \text{经济学性质保证，如博弈均衡}$$

优化拍卖机制需要解决如下问题：

- **机制性质如何满足**：需要一种简洁的数学形式表达机制需要满足的博弈性质，并将其融入到机制的优化过程中。
- **如何面向实际后验效果优化**：工业界中很多优化目标指标难以得到精确解析形式（例如成交额、商品收藏加购量等），如何通过真实反馈的方式优化机制也是需要考虑的。

3.1 主线：飘然凡尘，从只远观到深度优化的拍卖机制

从经典的拍卖机制开始，如何通过数据化 & 智能化提升拍卖机制的效果是发展主线，我们参考了业界大量的公开的正式文献，并结合阿里妈妈自身的技术发展，勾勒出拍卖机制的发展演进脉络。

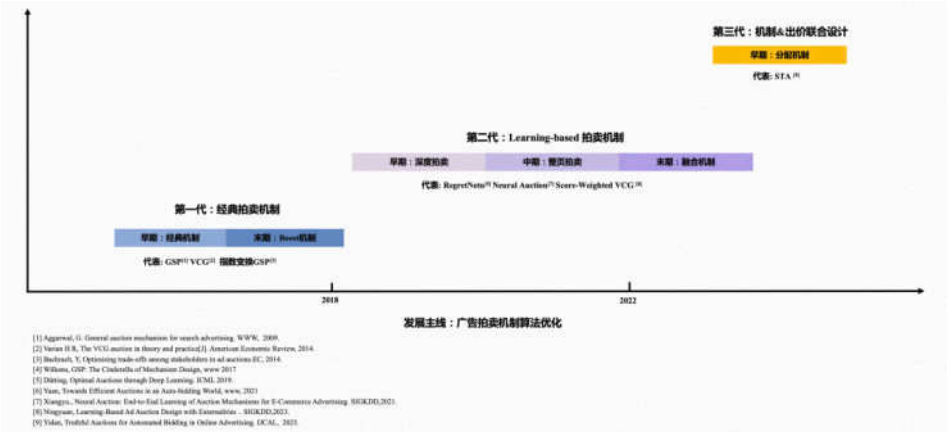


图 14　广告拍卖机制的发展主线：深度优化

整体而言可以划分为 3 个阶段：

第一代：经典拍卖机制

- 经典的 GSP[23]、VCG[24] 在互联网场景大规模落地后，针对场景特点的优化主要集中在两方面：1）提升平台收入，最典型的是 Squashing[25] 和保留价；2）多目标优化能力，通过在排序公式中引入更多的项来优化多目标，最典型的是 Ugsp。这些机制的分配和扣费形式相对清晰，所以关于他们的激励性质也大量被研究。

第二代：Learning-based 拍卖机制

- 随着深度学习 & 强化学习的蓬勃发展，大家开始探索将深度学习 / 强化学习引入到拍卖机制设计中，学术界典型的工作包括 RegretNet[26]、RDM[41] 等，阿里妈妈结合工业界的场景特点，先后设计出 Deep GSP[31]、Neural Auction[32]、Two-Stage Auction[33] 等机制，这些机制都借助了深度网络强大的学习能力，提升拍卖机制的优化效果。

第三代：拍卖机制 & 自动出价联合设计

- 随着自动出价能力的广泛应用，广告主竞价方式相较于之前有了大幅度的改变，广告主向平台提交高层次的优化目标和约束条件，然后由出价代理代表广告主在每次广告拍卖中做出详细的出价决策。对于广告主来说，平台需要把出价和拍卖机制看成一个整体联合设计，典型的工作包括 [36]。

为了让大家有更好的理解，我们以阿里妈妈的实践为基础，重点讲述下智能拍卖机制

在工业界的落地。

3.1.1 一相逢便胜却无数：当拍卖机制遇到智能化

惊艳登场：可 Learning 的拍卖机制

自 2019 年开始，学术界开始将深度学习 & 强化学习引入到机制设计中，如 RegretNet[26]、RDM[41] 等，他们通过引入深度网络强大的学习能力，提升拍卖机制的优化效果，为拍卖机制的发展开辟了一条新的道路。遗憾的是，这些工作都做了很强的理论假设如广告主个数固定等，没有看到在工业界大规模落地的实践。因此，我们开始思考，是否能够针对以上问题设计新型的面向多目标优化的广告拍卖机制，并能够结合工业界海量数据的优势，通过深度网络的强大学习能力来解决广告系统场景下的多目标优化问题。

我们提出一种基于深度神经网络的拍卖机制 Deep GSP[31]。Deep GSP 延续 GSP 的二价扣费机制，并通过深度网络提升其分配能力。不同于经典的广告拍卖机制，其能够通过深度网络的学习实现任意给定目标的优化，整个优化过程使用深度强化学习中确定性策略梯度算法实现。

我们对 Deep GSP 的模式进行了思考：其采用 GSP-Style 的机制设计模式，通过深度网络为每个广告计算出一个分数，排序后决定分配和扣费结果。训练时基于最终效果为参与竞价的每一条广告样本分配奖赏并采用强化学习的方法驱动模型参数更新。从机制的角度，求解最优分配问题是一个全局视角的组合优化问题，而 Deep GSP 是建模在广告粒度，如何把整体的效果分摊到每个广告上，即信用分配问题，会对训练产生很大的影响。但排序是一个不可微的操作，在模型训练的时候无法直接像监督学习那样通过样本标签计算的 loss 反向梯度传导优化模型参数。因此我们又提出了一种新的拍卖机制 Neural Auction[32]，以一种可微的计算形式来表达”排序”算子，从而能够与梯度下降训练方法结合，实现端到端优化，

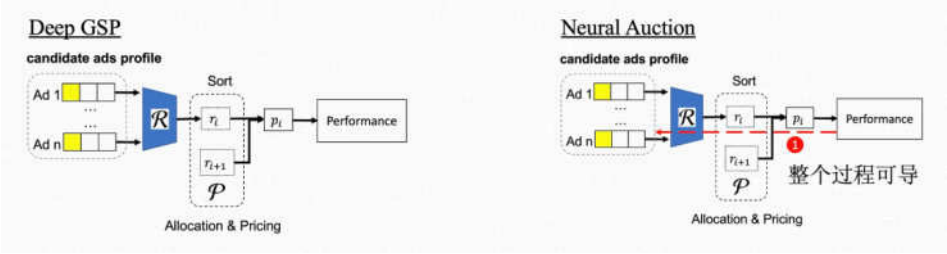


图 15　工业界 Learning-based 拍卖机制 2 个典型工作：Deep GSP 和 Neural Auction

值得注意的是，我们的工作也夯实了工业界智能拍卖机制(Learning-based Mechanism Design)方向，并得到了业界的广泛关注，其中所学术沉淀被国际会议Meta Reviewer和引用者使用开创新方向(“contributes a new perspective to the literature”)和首次(“the first attempts”)等方式评价。

持续发力：整页拍卖(考虑外部性)机制

广告拍卖机制的效果依赖于广告展示商品点击率(CTR)的精确预估，但在实际场景中，商品展示点击率会受到相互之间的外部性影响。这一现象在近年来开始受到学术界和工业界的广泛关注。然而，传统的广告拍卖通常简化或忽略了外部性。例如，广泛使用的GSP拍卖机制基于可分离CTR模型[37]，假定广告的点击率只由广告内容和位置决定，而忽略了其他商品的影响。因此传统的广告拍卖机制在考虑外部性时不再适用。

但考虑外部性影响对于最优广告拍卖的设计带来了许多挑战。由于广告的点击率受到上下文中其他商品的影响，即使对分配进行微小修改，也可能导致广告拍卖的预期收入发生复杂的变化。一般而言，对于外部性结构不作具体假设时，计算具有最大社会福利的分配方案是NP困难的。因此，如何设计高效实用的分配算法是一个非平凡的问题。另一方面，由于外部性影响的存在，拍卖机制更难控制每个广告主得到的效用，因此IC和IR等约束更难满足。

我们的工作[28]提出一个数据驱动的广告拍卖框架，以在考虑外部性的情况下实现收入最大化，同时确保满足IC和IR约束。结合理论分析提出Score-Weighted VCG框架，将最优拍卖机制的设计拆解为一个单调得分函数的学习和一个加权福利最大化算法的设计。基于这一框架又提出一个实用的实现方案，利用数据驱动的模型实现最优拍卖机制。通过完备的理论证明了该框架在各种感知外部性的点击率模型下都能产出满足激励兼容和个体理性的近似最优广告拍卖。

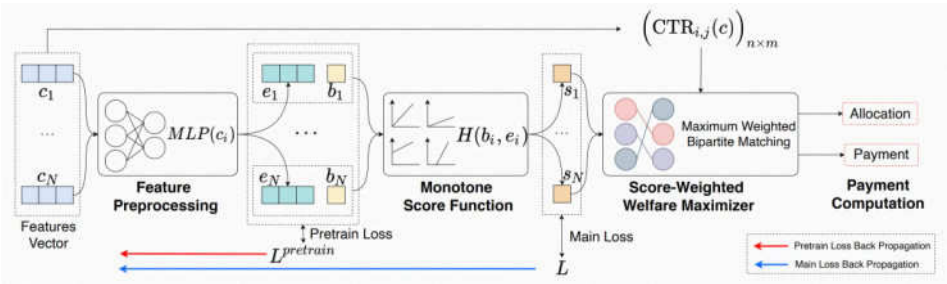


图16　Score-Weighted VCG：考虑外部性的整页拍卖机制

一片蓝海：融合机制设计

融合阶段是工业界一个非常关键的过程。在搜索和信息流等场景中，广告结果与自然结果分别由广告系统和推荐系统产生，融合机制对候选的广告和自然结果进行合并、筛选、排列，决定最终向用户展示的商品列表。



图 17　融合阶段是工业界系统中一个非常关键的过程

同时融合也是一个机制设计问题。广告结果和自然结果的分配不再是独立的，通过综合考虑广告和自然结果排列方式来优化用户体验和平台收入。另外，一个商品可能同时作为广告结果和自然结果的候选出现，这是因为广告系统和推荐系统都倾向于选择与用户偏好或搜索关键词较为匹配的商品。在此情形下，通常不允许将一个商品作为广告和自然结果同时展示给用户，导致对于广告结果和自然结果的分配不再是独立的，这也会导致广告主对广告的付费动机出现激励问题，因此必须重新审视广告与自然结果融合时的机制设计问题。

定坑可以理解为最经典的混排机制，自然结果优化用户体验，广告结果采用传统的机制如 GSP 来优化平台收入。混排通过经典的线性加权把多目标优化问题转换成一个单目标（用户体验和广告平台加权和）的优化问题。所有商品都按给定的排序公式进行打分，按分数从大到小逐个放置到所有坑位里面，并用 uGSP 进行扣费。但因外部性的普遍存在，该方式通常无法得到最优解。

业界普遍在探索的是广告和自然整页优化方式，基于组合优化思想来解该多目标优化问题，通常隐式或者显式地对外部性进行建模，目前妈妈和业界都有一些典型的优化

工作 [38][39]，在机制性质上还有很多的研究空间。

3.1.2 浑然一体：自动出价和拍卖机制的联合设计

随着自动出价产品的广泛应用，现在广告主参竞的方式相较于之前有了大幅度的改变：广告主向平台提交其高层次的优化目标和约束条件，然后由机器学习算法驱动的出价代理代表广告主在每次广告拍卖中做出详细的出价决策。通过自动出价工具，广告主从全局角度针对其经济约束优化其整体广告目标。对于广告主来说，自动出价和拍卖机制整体才是平台真正的机制。

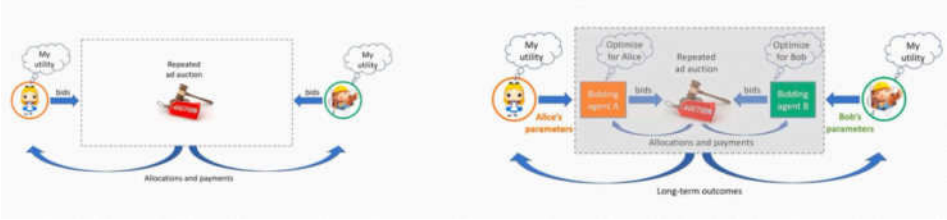


图 18　在自动出价体系下，广告主与广告平台的博弈关系已发生根本改变

在自动出价的新广告范式中，我们需要重新审视经典的拍卖机制模型是否仍然适用。由于可以获取有关广告主与用户之间互动的历史数据，平台可以估计用户的潜在行为（如点击和转化），这些行为可以被视为广告主对物品的估值。在自动出价中，广告主的私有信息实际上是其在整个广告投放过程的约束条件。这些与经典拍卖截然不同的新特点需要对应的新的广告拍卖模型，以激励广告主真实地上报其高层次的私有约束。

我们的工作 [36] 提出了一类基于排序函数的激励兼容机制，关键思想是采用提前确定的排序函数为每个广告主进行排序，并将阈值 ROI 设计为赢得足够多的竞价机会以消耗完预算的最大 ROI。在该机制中，给定广告主上报的预算和 ROI，首先基于排序函数计算不同广告主对于每个物品的虚拟出价。只要这些排序函数在 ROI 上是单调递减的，保证最终的拍卖机制是满足 DSIC 与 IR 的。接下来，将每个物品分配给排序分数最高的广告主，并根据第二高的排序函数计算赢得此物品所需要的 ROI。为了保证约束的 IC，我们使用前面提到的基本规则来计算关键 ROI，即赢得足够多的物品以消耗完预算的最大 ROI，其中使用关键 ROI 作为实际 ROI 来计算支付。这是一个对此类问题的初步尝试，未来还需要进一步深入思考。

3.2 副线：多样的广告主行为建模

广告主行为建模是拍卖机制设计的基础，现有的关于 VCG 和 GSP 的分析主要建立在拟线性效用模型上，也被称为效用最大化广告主（Utility Maximizer，UM），即广告主的目标是优化其分配的价值和扣费之间的差值。雅虎公司的研究人员 Wilkens、Cavallo 和 Niazadeh 为广告主提出了另一个模型，称为价值最大化广告主（Value Maximizer，VM），该模型将分配的价值作为广告主的首要目标，将扣费作为其次的目标，只有当价值相同时才偏好扣费更少的结果。这些设定都接近于单轮拍卖形式下广告主的行为模式，但在广告主已经开始使用自动竞价（Auto-bidding）工具，利用自动竞价工具，广告主只需要设置高层次的约束条件，并由出价代理进行竞价，这与传统的机制存在非常大的差异。因此，核心问题是使用不同的机制，在广告主与代理间的交互完成后，会得到怎样的博弈结果？什么 机制对平台方或社会福利更好这些都是要回答的问题。



图 19　广告主行为建模的研究方向

4. 结语

雄关漫道真如铁，而今迈步从头越。历经阿里妈妈技术同学们坚持不懈的努力，在自动出价决策技术上，从推动经典强化学习类算法在工业界大规模落地，到持续革新提出 Offline RL-based Bidding、Online RL-based Bidding 等适应工业界特点的新算法，再到提出 AIGB 迈入生成式 Bidding 的新时代；在拍卖机制设计上，从只远观的高深领域，到可 Learning 的决策问题，再与工业界深入结合的 Two-Stage Auction、整页拍卖、融合机制等，以及未来的 Auto-bidding 和拍卖机制的联合优

化。一路走来，我们持续推动业界广告决策智能技术的发展，并秉承开放共赢，把我们的工作以学术化沉淀的方式实现对学术界研究的反哺。希望大家多多交流，共赴星辰大海。

关于我们

核心关键词：超核心业务、大规模 RL 工业界落地、决策智能大模型、技术引领业界、团队氛围好！

「智能广告平台」基于海量数据，优化阿里广告技术体系，驱动业务增长，并推动技术持续走在行业前沿：精准建模以提升商业化效率，创新广告售卖机制和商业化模式以打开商业化天花板，研发最先进的出价算法帮助商家获得极致的广 告投放效果和体验，设计和升级算法架构以支撑国内顶级规模的广告业务稳健 & 高效迭代等。

超大业务体量和丰富商业化场景，赋能我们在深度学习、强化学习、机制设计、投放策略、顶层业务 / 技术上的视野和判断极速成长并沉淀丰厚；超一线站位也让我们在“挖掘有价值 & 有挑战新问题，驱动产品技术能力创新等”方面有得天独厚优势。欢迎聪明靠谱小伙伴加入（社招、校招、实习生、高校合作、访问学者等）。

简历投递邮箱：alimama_tech@service.alibaba.com

参考文献

[1] Chen Y, Berkhin P, Anderson B, et al. Real-time bidding algorithms for performance-based display ad allocation[C]//Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining. 2011: 1307-1315.

[2] Zhang W, Rong Y, Wang J, et al. Feedback control of real-time display advertising[C]//Proceedings of the Ninth ACM International Conference on Web Search and Data Mining. 2016: 407-416.

[3] Yu H, Neely M J. A Low Complexity Algorithm with Regret and Constraint Violations for Online Convex Optimization with Long Term Constraints[J]. arXiv preprint arXiv:1604.02218, 2016.

[4] Yu H, Neely M, Wei X. Online convex optimization with stochastic constraints[J]. Advances in Neural Information Processing Systems, 2017, 30.

[5] Zhao J, Qiu G, Guan Z, et al. Deep reinforcement learning for sponsored search real-time bidding[C]//Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining. 2018: 1021-1030.

[6] Cai H, Ren K, Zhang W, et al. Real-time bidding by reinforcement learning in display advertising[C]//Proceedings of the tenth ACM international conference on web search and data mining. 2017: 661-670.

[7] Jin J, Song C, Li H, et al. Real-time bidding with multi-agent reinforcement learning in display advertising[C]//Proceedings of the 27th ACM international conference on information and knowledge management. 2018: 2193-2201.

[8] Wu D, Chen X, Yang X, et al. Budget constrained bidding by model-free reinforcement learning in display advertising[C]//Proceedings of the 27th ACM International Conference on Information and Knowledge Management. 2018: 1443-1451.

[9] He Y, Chen X, Wu D, et al. A unified solution to constrained bidding in online display advertising[C]//Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining. 2021: 2993-3001.

[10] Yang X, Li Y, Wang H, et al. Bid optimization by multivariable control in display advertising[C]//Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining. 2019: 1966-1974.

[11] Guan Z, Wu H, Cao Q, et al. Multi-agent cooperative bidding games for multi-objective optimization in e-commercial sponsored search[C]//Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining. 2021: 2899-2909.

[12] Wen C, Xu M, Zhang Z, et al. A cooperative-competitive multi-agent framework for auto-bidding in online advertising[C]//Proceedings of the Fifteenth ACM International Conference on Web Search and Data Mining. 2022: 1129-1139.

[13] Mou Z, Huo Y, Bai R, et al. Sustainable Online Reinforcement Learning for Auto-bidding[J]. Advances in Neural Information Processing Systems, 2022, 35: 2651-2663.

[14] 阿里妈妈生成式出价模型（AIGB）详解 https://zhuanlan.zhihu.com/p/619301816, 2023

[15] Lin Q, Tang B, Wu Z, et al. Safe Offline Reinforcement Learning with Real-Time Budget Constraints[J]. arXiv preprint arXiv:2306.00603, 2023.

[16] Zhang H, Niu L, Zheng Z, et al. A Personalized Automated Bidding Framework for Fairness-aware Online Advertising[C]//Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining. 2023: 5544-5553.

[17] Gong Z, Niu L, Zhao Y, et al. MEBS: Multi-task End-to-end Bid Shading for Multi-slot Display Advertising[C]//Proceedings of the 32nd ACM International Conference on Information and Knowledge Management. 2023: 4588-4594.

[18] Ou, W., Chen, B., Liu, W., Dai, X., Zhang, W., Xia, W., Li, X., Tang, R., & Yu, Y. (2023). Optimal Real-Time Bidding Strategy for Position Auctions in Online Advertising. Proceedings of the 32nd ACM International Conference on Information and Knowledge Management.

[19] Gligorijevic, D., Zhou, T., Shetty, B., Kitts, B., Pan, S., Pan, J., & Flores, A. (2020). Bid Shading in The Brave New World of First-Price Auctions. Proceedings of the 29th ACM International Conference on Information & Knowledge Management.

[20] Zhang, W., Kitts, B., Han, Y., Zhou, Z., Mao, T., He, H., Pan, S., Flores, A., Gultekin, S., & Weissman, T. (2021). MEOW: A Space-Efficient Nonparametric Bid Shading Algorithm. Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining.

[21] Kumar, A., Zhou, A., Tucker, G., & Levine, S. (2020). Conservative Q-Learning for Offline Reinforcement Learning. ArXiv, abs/2006.04779.
[22] Kostrikov, I., Nair, A., & Levine, S. (2021). Offline Reinforcement Learning with Implicit Q-Learning. ArXiv, abs/2110.06169.
[23] Aggarwal, G., Muthukrishnan, S., Pál, D., & Pál, M. (2008). General auction mechanism for search advertising. ArXiv, abs/0807.1297.
[24] Varian, H.R., & Harris, C. (2014). The VCG Auction in Theory and Practice. The American Economic Review, 104, 442-445.
[25] Bachrach, Y., Ceppi, S., Kash, I.A., Key, P.B., & Kurokawa, D. (2014). Optimising trade-offs among stakeholders in ad auctions. Proceedings of the fifteenth ACM conference on Economics and computation.
[26] Dütting, P., Feng, Z., Narasimhan, H., & Parkes, D.C. (2017). Optimal auctions through deep learning. Communications of the ACM, 64, 109 – 116.
[27] Deng, Y., Mao, J., Mirrokni, V.S., & Zuo, S. (2021). Towards Efficient Auctions in an Auto-bidding World. Proceedings of the Web Conference 2021.
[28] Li, N., Ma, Y., Zhao, Y., Duan, Z., Chen, Y., Zhang, Z., Xu, J., Zheng, B., & Deng, X. (2023). Learning-Based Ad Auction Design with Externalities: The Framework and A Matching-Based Approach. Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining.
[29] Xing, Y.Y., Zhang, Z., Zheng, Z., Yu, C., Xu, J., Wu, F., & Chen, G. (2023). Truthful Auctions for Automated Bidding in Online Advertising. International Joint Conference on Artificial Intelligence.
[30] Wilkens, C.A., Cavallo, R., & Niazadeh, R. (2017). GSP: The Cinderella of Mechanism Design. Proceedings of the 26th International Conference on World Wide Web.
[31] Zhang, Z., Liu, X., Zheng, Z., Zhang, C., Xu, M., Pan, J., Yu, C., Wu, F., Xu, J., & Gai, K. (2020). Optimizing Multiple Performance Metrics with Deep GSP Auctions for E-commerce Advertising. Proceedings of the 14th ACM International Conference on Web Search and Data Mining.
[32] Liu, X., Yu, C., Zhang, Z., Zheng, Z., Rong, Y., Lv, H., Huo, D., Wang, Y., Chen, D., Xu, J., Wu, F., Chen, G., & Zhu, X. (2021). Neural Auction: End-to-End Learning of Auction Mechanisms for E-Commerce Advertising. Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining.
[33] Wang, Y., Liu, X., Zheng, Z., Zhang, Z., Xu, M., Yu, C., & Wu, F. (2021). On Designing a Two-stage Auction for Online Advertising. Proceedings of the ACM Web Conference 2022.
[34] Liu, Y., Chen, D., Zheng, Z., Zhang, Z., Yu, C., Wu, F., & Chen, G. (2023). Boosting Advertising Space: Designing Ad Auctions for Augment Advertising. Proceedings of the Sixteenth ACM International Conference on Web Search and Data Mining.
[35] Lv, H., Zhang, Z., Zheng, Z., Liu, J., Yu, C., Liu, L., Cui, L., & Wu, F. (2022). Utility Maximizer or Value Maximizer: Mechanism Design for Mixed Bidders in Online Advertising. AAAI Conference on Artificial Intelligence.

[36] Xing, Y., Zhang, Z., Zheng, Z., Yu, C., Xu, J., Wu, F., & Chen, G. (2023). Designing Ad Auctions with Private Constraints for Automated Bidding. ArXiv, abs/2301.13020.

[37] Varian H R. Position auctions[J]. international Journal of industrial Organization, 2007, 25(6): 1163-1178.

[38] Zhao, X., Gu, C., Zhang, H., Yang, X., Liu, X., Tang, J., & Liu, H. (2019). DEAR: Deep Reinforcement Learning for Online Advertising Impression in Recommender Systems. AAAI Conference on Artificial Intelligence.

[39] Chen, D., Yan, Q., Chen, C., Zheng, Z., Liu, Y., Ma, Z., Yu, C., Xu, J., & Zheng, B. (2022). Hierarchically Constrained Adaptive Ad Exposure in Feeds. Proceedings of the 31st ACM International Conference on Information & Knowledge Management.

[40] Liao, G.R., Wang, Z., Wu, X., Shi, X., Zhang, C., Wang, Y., Wang, X., & Wang, D. (2021). Cross DQN: Cross Deep Q Network for Ads Allocation in Feed. Proceedings of the ACM Web Conference 2022.

[41] Shen, W., Peng, B., Liu, H., Zhang, M., Qian, R., Hong, Y., Guo, Z., Ding, Z., Lu, P., & Tang, P. (2020). Reinforcement Mechanism Design: With Applications to Dynamic Pricing in Sponsored Search Auctions. *AAAI Conference on Artificial Intelligence.*

Bidding 模型训练新范式：阿里妈妈生成式出价模型（AIGB）详解

作者：银耀、子述、妙临

导读

今天以 ChatGPT 为代表的生成式大模型让科技行业重新兴奋起来，也为广告营销注入了新的想象力。生成式大模型几乎一定会带来用户与互联网产品交互模式的改变，进而颠覆广告营销模式。广告技术人，你们准备好了吗？阿里妈妈技术已提前在该方向布局，并推出了新的广告营销智能技术体系，今天将揭露出其神秘面纱的一角，窥探背后的思考和实践。

摘要

出价产品智能化成为行业趋势，极简产品背后则是强大的自动出价的支撑，其技术不断演进走过了 3 个大的阶段：PID 控制、RL-based Bidding、SORL(Sustainable Online RL)，那么下一步代际性技术升级是什么？今天以 ChatGPT 为代表的生成式大模型以汹涌澎湃之势到来，几乎一定会颠覆广告营销模式，一方面，新的用户交互模式会孕育新的商业机会，给自动出价的产品带来巨大改变；另一方面，新的技术理念和技术范式也会给自动出价算法带来革命性的升级。阿里妈妈技术团队提前布局，以智能营销决策大模型 AIGA（AI Generated Action）为核心重塑了广告智能营销的技术体系，并衍生出以 AIGB（AI Generated Bidding）为代表的各种领域技能模型。**AIGB 是一种基于生成式模型构造的出价模型优化方案**，与以往解决序列决策问题的强化学习视角不同，其将策略建模为条件生成模型，从而消除了以往强化学习视角下的复杂性问题。具体实现上，将出价、优化目标和约束等具备相关性的指标视为一个联合概率分布，并以优化目标和约束项为条件，生成相应出价策略的条件分布。训练时将历史次优投放轨迹数据作为训练样本，以最大似然估计的方式拟合轨迹数据中的分布特征；推断时基于约束和优化目标，以符合分布规律的方式输出出价策略。本文提出的方案可避免传统 RL 方案中的分布偏移和策略退化问题，又具备满足不同出价类型和不同约束的灵活性。通过 AIGB 的技术研究和线上实践，我们愈发地感受到新的技术浪潮正在朝我们奔来，AIGB 只是这一切的开始 ...

一、背景

1.1 出价产品智能化成为行业趋势

广告平台吸引广告主持续投放的核心在于给广告主带来更大的投放价值，出价产品的智能化已成为行业趋势并加以重点建设的能力 (如图 1)。以阿里妈妈为代表的互联网广告平台不断地探索流量的多元化价值，并设计更能贴近营销本质的自动出价产品，广告主只需要简单的设置就能清晰的表达出营销诉求。极简产品背后则是强大的出价策略支撑，广告主出价策略从海量数据中挖掘更好的营销模式，提升广告主对特定价值的优化能力，赋能广告主投放。

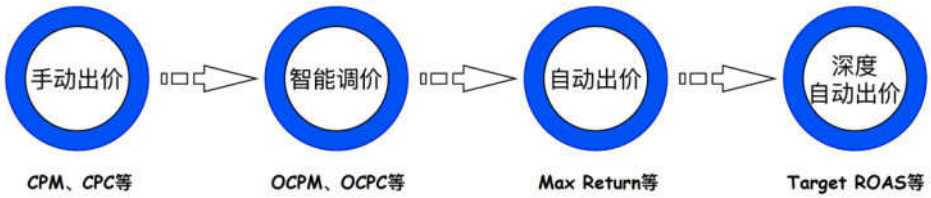


图 1　出价产品的演进趋势，智能化逐步成为互联网广告产品的标配

1.2 自动出价技术的不断演进

阿里妈妈技术团队多年来致力于极致的优化自动出价策略，帮助广告主获得最好的投放效果，其自动出价策略的技术演进可以大体分为三个大的阶段，具体如下图。

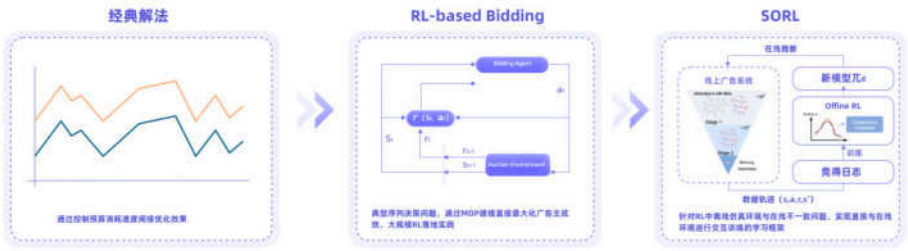


图 2　典型的自动出价技术演进路线，从预算消耗控制 ->RL-based Bidding->SORL，下一步代际性升级是什么?

第一阶段：预算消耗控制，通过控制预算的消耗速度尽可能平滑来优化效果，一般通过经典的控制算法，如 PID 等。在假设竞价环境中流量价值分布均匀的情况下，这种方法能够达到比较好的效果。

第二阶段：RL-based Bidding，现实环境中的竞价环境是非常复杂且动态变化的，只控制预算无法满足更多样的出价计划的进一步优化。AlphaGo 的惊艳表现，展现了强化学习的力量，而自动出价是一个非常典型的序列决策问题，在预算周期内，前面花的好不好会影响到后面的出价决策，而这正是强化学习的强项，因此第二阶段我们用了基于强化学习的 Bidding。Simulation basedbidding 的一些工作 [1] 奠定了我们在广告主报价领域的领先地位。

第三阶段：SORL，它的特点是针对强化学习中离线仿真环境与在线环境不一致。我们直接在在线环境中进行可交互的学习，这是工程设计和算法设计联合的例子。SORL[2] 上线之后，很大程度上解决了强化学习强依赖于仿真平台的问题。

今天以 ChatGPT 为代表的生成式大模型让科技行业重新兴奋起来，也为广告营销注入了新的想象力。生成式大模型几乎一定会带来用户与互联网产品交互模式的改变，例如，多模态交互式对话方式会取代搜索引擎的地位，以广告位拍卖为基础的互联网广告的逻辑也会发生改变。一方面，新的用户交互模式会孕育新的商业机会，给自动出价的产品带来颠覆的改变；另一方面，新的技术理念和技术范式也会给自动出价算法带来革命性的升级。

如今，革命性升级已经到来！

二、相关工作

2.1 自动出价建模

考虑到广告目标、预算和 M 个 KPI 约束，计划的诉求可以通过（LP1）表示为统一的带约束竞价问题。

$$\max_{xi} \sum_i v_i x_i$$
$$s.t. \sum_i c_i x_i \le B$$
$$\frac{\sum_i c_{ij} x_i}{\sum_i p_{ij} x_i} \le k_j, \forall j$$
$$x_i \le 1, \forall i$$
$$x_i \ge 0, \forall i$$

如果已经知道流量集合的全部信息，包括能够触达的每条流量 i 的流量价值 v_i 和成本 c_i 等，那么可以通过解决线性规划问题（LP1）来获得最优解 x_i。然而，在实际应用

中，我们需要在流量集合未知的情况下进行实时竞价。由于在线广告池的动态变化以及每天访问用户的随机性，很难通过准确的预测来构建流量集合。因此，常规的线性规划解决方法并不完全适用。所以在实际应用中，通过对上述出价公式的一些变换，构造一个最优出价公式，将原问题转化为求解最优参数的问题，从而大大降低了在线情况下求解此问题的难度。

最优的出价公式为：

$$b_i^* = w_0^* v_i - \sum_j w_j^* (q_{i,j}(1 - 1_{CR_j}) - k_j p_{ij})$$

其中，$q_{i,j}$ 是常数项，w^* 是参数，其范围为：$w_0^* > 0$。如果约束 j 是 CR，则 $w_j^* \in [0, 1]$; 如果约束 j 是 NCR，则 $w_j^* > 0$。证明过程详见论文 [1]。

最优出价公式共包含 m+1 个核心参数 wk, $k \in [0, ..., M]$，公式中其余项为在线流量竞价时可获得的流量信息。由于最优出价公式存在，对于具有预算约束和 M 个 KPI 约束、且希望最大化赢得流量的总价值的问题，最优解可以通过找到 M+1 个最优参数并根据公式进行出价，而不是分别为每个流量寻找最优出价。理想情况下，通过求解最优参数 w_j^*，即能直接获得每个广告计划的最优出价。我们可以通过 PID 或者 RL 来逼近真实环境中的最优参数。

2.2 生成式模型

生成式模型近年来得到了迅速的发展，在图像生成、文本生成、计算机视觉等领域取得了重大突破，并催生出了近期大热的 ChatGPT 等。生成式模型主要从数据分布的角度去理解数据，并通过拟合训练数据集中的样本分布来进行特征提取，最终生成符合数据集分布的新样本。目前常用的生成式模型包括 Transformer[3]、Diffusion Model[4] 等。Transformer 主要基于自注意力机制，能够对样本中跨时序和分层信息进行提取和关联，擅长处理长序列和高维特征数据，如图像、文本和对话等。而 Diffusion Model 则将数据生成看作一个分阶段去噪的过程，将生成任务分解为多个步骤，逐步加入越来越多的信息，从而生成目标分布中的样本。这一过程与人类进行绘画过程较为相似，由此可见，Diffusion Model 擅长处理图像生成等任务。

依靠生成式模型强大的信息生成能力，我们也可以引入生成式模型将序列决策问题建模为一个序列动作生成问题。模型通过拟合历史轨迹数据中的行为模式，达到策略输出的目标。Decision Transformer(DT)[5] 和 Decision Diffuser(DD)[6] 分别将

Transformer 以及 Diffusion Model 应用于序列决策，在通用数据集中，相比主流的 RL 方法 [7,8] 取得了较好的效果提升。这一结果为我们的 Bidding 建模提供了一个可用的迭代方案。

三、AIGB（AI Generated Bidding）

3.1 智能营销技术体系的重塑

早在生成式模型如 ChatGPT 惊艳之前，阿里妈妈技术团队就已经开始尝试用生成式和大模型重塑智能营销的技术体系，并持续投入相应的团队和资源，设计了一套全新的智能营销技术体系。



图 3　以智能营销决策大模型为核心的智能营销技术体系

其中，在营销层，革新了以往功能繁多操作麻烦的 BP，给广告主带来一种新的对话式交互体验，广告主只需要通过简单的自然语言的描述，即可实现全部的营销流程，大大简化了广告主的操作和学习成本。而这些都依赖于强大的智能营销决策大模型 AIGA（AI Generated Action），以及衍生出来的各种领域技能模型，典型的领域技能模型 AIGB（AI Generated Bidding）是专门服务于自动出价算法的模型。这些模型的训练基于阿里集团自研的高性能硬件以及相应的框架。

3.2 AIGB 建模方案

AIGB 是一种基于生成式模型构造的出价模型优化方案。与以往解决序列决策问题的

强化学习视角不同，AIGB 将策略建模为条件生成模型，从而消除了以往强化学习视角下的复杂性问题。我们进一步考虑额外的条件变量，展示了将出价策略建模为条件生成模型的优势。在训练过程中，对约束进行条件化，使得推断时的行为可以同时满足多个约束组合。我们的研究结果表明，使用条件生成式模型来解决出价问题中的序列决策问题是一个好的选择。



图 4　图左历史投放轨迹中，颜色深浅代表计划 return 的不同。右图为 AIGB 模型根据不同需求生成的新策略。整个模型看作一个分布处理 pipeline，输入历史非最优但存在有效信息的广告投放轨迹，输出符合优化目标的新策略。

从生成式模型的角度来看，我们可以将出价、优化目标和约束等具备相关性的指标视为一个联合概率分布，从而将出价问题转化为条件分布生成问题。这意味着我们可以以优化目标和约束项为条件，生成相应出价策略的条件分布。图 4 直观地展示了生成式出价（AIGB）模型的流程：在训练阶段，模型将历史次优投放轨迹数据作为训练样本，以最大似然估计的方式拟合轨迹数据中的分布特征。这使得模型能够自动学习出价策略、状态间转移概率、优化目标和约束项之间的相关性。在线上推断阶段，生成式模型可以基于约束和优化目标，以符合分布规律的方式输出出价策略。总的来说，**生成式模型的优势**在于：

- **训练阶段**，条件生成式模型通过最大似然估计进行训练，可以最大程度地避免分布偏移和策略退化问题。
- **推断阶段**，条件生成式模型可以根据不同的出价类型生成不同的出价轨迹，以实现不同约束项的满足。

3.2.1 模型结构：

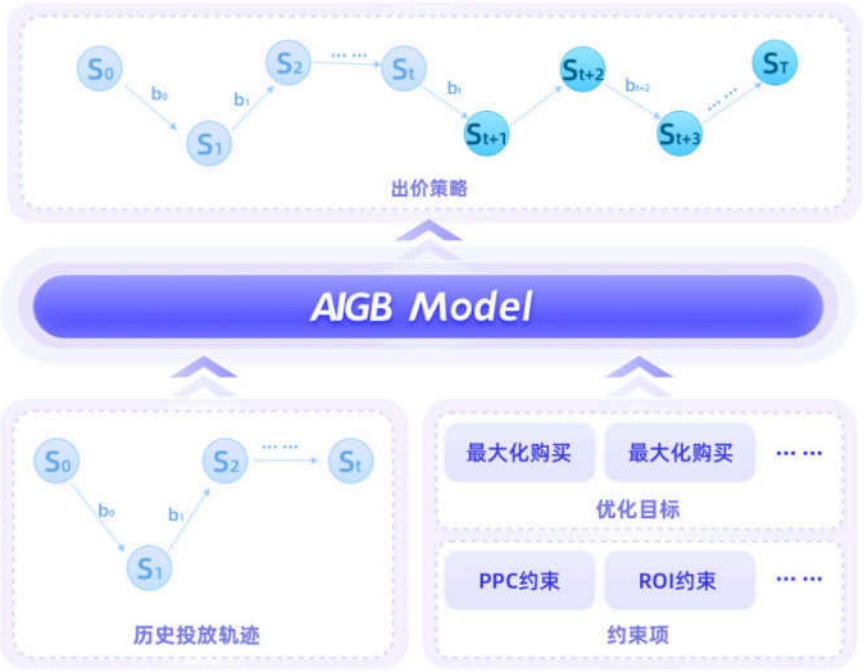


图 5　AIGB 结构

如图 5，给定当前轨迹信息 $x_t^{-1}(\tau) := (s_0, b_0, s_1, \ldots, s_t)$ 和策略生成条件 $y(\tau)$，AIGB 模型可以逐个生成未来的出价策略：

$$p_\theta(x_t(\tau)|x_t^{-1}, y)$$

其中出价策略 $x_t(\tau) := (s_t^*, b_t^*, s_{t+1}^*, \ldots, s_T^*)$ 是由未来的最优状态和与之对应的最优出价组成的序列。生成条件 $y(\tau)$ 包括了优化目标（购买量最大化、点击量最大化）以及约束项（PPC、ROI、投放平滑性）等。p_θ 被用来估计条件概率分布。模型基于当前的投放状态信息以及策略生成条件输出未来的投放策略，相比于以往的 RL 策略仅仅黑盒输出单步 action，AIGB 策略可以被理解为在规划的基础上进行决策，更擅长处理长序列问题。这一优点有利于我们在实践中进一步减小出价间隔，提升策略的快速反馈能力。与此同时，基于规划的出价策略也具备更好的可解释性，能够帮助我们更好地进行离线策略评估，方便专家经验与模型深度融合。

3.2.2 训练方法：

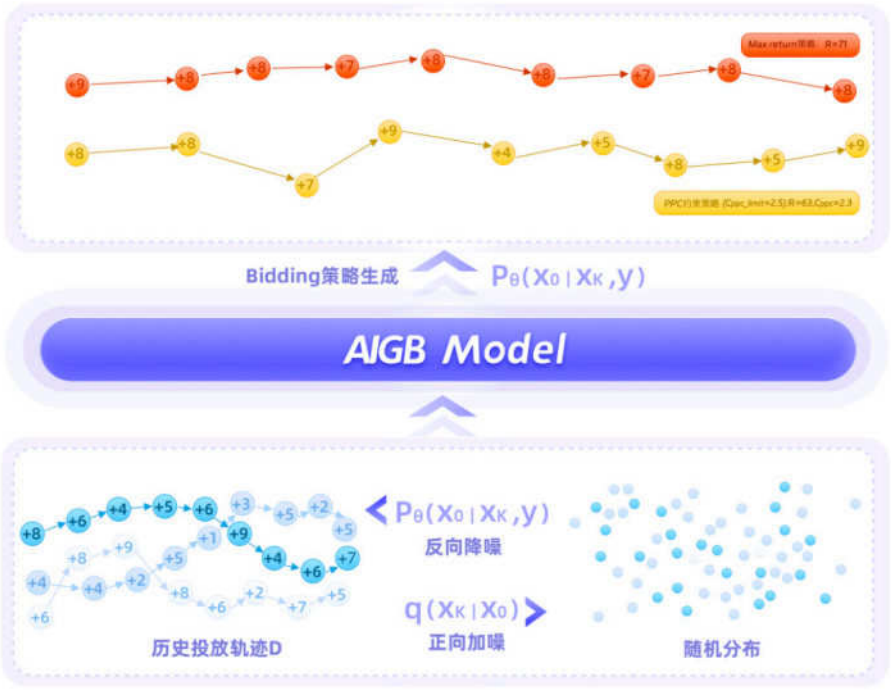


图 6　AIGB 模型训练过程

AIGB 模型通过最大似然估计历史数据集 D 中轨迹 $x_t^{-1}(\tau) := (s_0, b_0, s_1, \ldots, s_t)$ 和策略生成条件 $y(\tau)$ 所对应的轨迹信息进行训练，从而最大限度拟合历史轨迹的分布信息：

$$\max_{\theta} \mathbb{E}_{\tau \sim D}[log\, p_\theta(x(\tau)) | x^{-1}(\tau), y(\tau)]$$

拟合历史分布的过程可以通过引入 Diffusion Model 或 Transformer 等生成式模型来完成。以我们真实使用的扩散模型为例，我们将序列决策问题看作一个条件扩散过程，包括正向过程 $q(x_{k+1}(\tau)|x_k(\tau))$ 和反向过程 $p_\theta(x_{k-1}(\tau)|x_k(\tau), y(\tau))$。整个训练过程如图 6 所示，k 表示正向过程的迭代步，在正向过程，高斯噪声 $x_K(\tau)$ 转化为历史投放轨迹分布 $x_0(\tau)$；反向过程则表示从 $x_0(\tau)$ 转变为 $x_K(\tau)$ 的过程。每一次 $x_k(\tau)$ 到 $x_{k-1}(\tau)$ 的转换均通过加入含有一定信息的高斯扰动实现。除此之外，在反向过程中，我们还希望能够表达 $y(\tau)$ 与 的相关性，因此可以引入 DD 模型中使用的 Classifier-free 方法，利用

$$(\epsilon_\theta(x_k(\tau), x^{-1}(\tau), y(\tau), k) - \epsilon_\theta(x_k(\tau), x^{-1}(\tau), \varnothing, k))$$

提取数据集中与 $y(\tau)$ 相关度最高的部分。其中 ϵ_θ 为噪声模型，通过神经网络生成每一个时间步所增加的噪声。k 步所对应的高斯扰动可以表示为：

$$\hat{\epsilon} := \epsilon_\theta(x_k(\tau), \varnothing, k) + \omega \sum_{i=1}^{n} (\epsilon_\theta(x_k(\tau), x^{-1}(\tau), y^i(\tau), k) - \epsilon_\theta(x_k(\tau), x^{-1}(\tau), \varnothing, k))$$

其中 i 表示不同的目标或者约束，ω 用来调节 $y(\tau)$ 的权重。Classifier-free 方法可以较为优雅地处理多种优化目标和约束条件，避免以往 RL 训练过程中由于约束信号稀疏而效果下降的问题。我们将这一基于扩散模型进行出价建模的方法称为 **Decision Diffuser based Generative Bidding（DDGB）**。DDPG 的 Planning 具体过程如下：

Algorithm 1 Decision Diffuser based Generative Bidding

1: 输入噪声模型ϵ_θ，序列长度T、生成条件y
2: **for** 时刻$t = 0...T$ **do**
3: 对于状态s_t,初始化$x_k \sim \mathcal{N}(0, \alpha I)$
4: **for** 迭代步$k = K...1$ **do**
5: $\hat{\epsilon} \leftarrow \epsilon_\theta(x_k(\tau), \emptyset, k) - \omega(\epsilon_\theta(x_k(\tau), x^{-1}(\tau), y(\tau), k) - \epsilon_\theta(x_k(\tau), x^{-1}(\tau), \emptyset, k))$
6: $x_{k-1} = \frac{1}{\sqrt{\alpha_t}}(x_k - \frac{1-\alpha_t}{\sqrt{1-\alpha_t}}\hat{\epsilon}) + \sigma_t z$
7: **end for**
8: **end for**

四、总结及未来展望

AIGB 方案可以带来诸多优势，包括解决困扰 RL Bidding 在离线不一致问题，更好地训练多约束出价模型，更好的可解释性以及更为顺畅的与专家经验的结合能力等，这些优点可以帮助我们进一步提升模型迭代效率和效果上限。可以看出，生成式模型驱动的 AIGB 已经在以完全不同的方式重构自动出价的技术体系。但是，这仅仅是一个开始。阿里妈妈沉淀了亿级广告投放轨迹数据，是业界为数不多具备超大规模决策类数据资源储备的平台。这些海量数据资源可以成为营销决策大模型训练的有力保证，从而推动 AIGA 技术的发展。与此同时，用户和互联网产品的交互方式也将发生深刻的变化。重塑广告营销模式的机会之门已经在变化之中逐步显现，我们需要做的就是通过持续不断地探索和尝试来迎接变化。期待后续有机会与大家分享和交流我们的进展与实践。

参考文献

[1] He Y, Chen X, Wu D, et al. A unified solution to constrained bidding in online display advertising[C]//Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining. 2021: 2993-3001.

[2] Mou Z, Huo Y, Bai R, et al. Sustainable Online Reinforcement Learning for Auto-bidding[J]. arXiv preprint arXiv:2210.07006, 2022.

[3] Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need[J]. Advances in neural information processing systems, 2017, 30.

[4] Ho J, Jain A, Abbeel P. Denoising diffusion probabilistic models[J]. Advances in Neural Information Processing Systems, 2020, 33: 6840-6851.

[5] Chen L, Lu K, Rajeswaran A, et al. Decision transformer: Reinforcement learning via sequence modeling[J]. Advances in neural information processing systems, 2021, 34: 15084-15097.

[6] Ajay A, Du Y, Gupta A, et al. Is Conditional Generative Modeling all you need for Decision-Making?[J]. arXiv preprint arXiv:2211.15657, 2022.

[7] Kumar A, Zhou A, Tucker G, et al. Conservative q-learning for offline reinforcement learning[J]. Advances in Neural Information Processing Systems, 2020, 33: 1179-1191.

[8] Kostrikov I, Nair A, Levine S. Offline reinforcement learning with implicit q-learning[J]. arXiv preprint arXiv:2110.06169, 2021.

万字长文，漫谈广告技术中的拍卖机制设计（经典篇）

作者：石士

搜索、推荐和广告在过去几年互联网蓬勃发展的浪潮中起到了核心助推引擎的作用，三者技术发展也是互相借鉴和相辅相成，有很多共通之处也有不少差异的地方。本文从广告视角出发，重点介绍广告技术与搜推技术最本质的差异点——拍卖机制的原理和实践。作为最具广告特色的技术模块，拍卖机制理解起来往往较为晦涩。一方面因为这个技术领域是基于博弈论思维，概念较多且定理推导复杂，和已经标准化的基于数据驱动的机器学习思维截然不同；另一方面因为理论假设过于严格，和实际差距较大，使得论证过程无论多么完善，实践落地仍然需要考量诸多复杂因素。本文尝试将拍卖机制的几个经典问题做一个脉络性梳理，概念和论证过程可自行查阅这里不再赘述，但概念之间的演进关系会重点阐述，希望可以勾勒清楚技术全貌，有助于大家系统性地认识计算广告领域的拍卖机制设计。

本文会按照几个小话题顺次展开介绍：

- 广告领域的拍卖机制简化而言是一个什么拍卖问题，背后最基本的博弈论概念是什么；最理想的拍卖机制长什么样，需要满足哪些条件；理想照进现实，哪些条件可以妥协松弛使得真正落地的拍卖机制即使没有非常理想但依然运行良好；本文所说的经典的拍卖机制设计范畴是什么?
- 社会福利最大化的拍卖机制如何设计才能使得市场蛋糕被有效扩大；点击率的不同假设这一关键变量如何影响拍卖机制的性质；机制设计需要做哪些与之对应的调整与升级?
- 平台营收最大化的拍卖机制如何设计才能使得市场蛋糕被最优地分配；以保留价设计为代表的这一关键技术是如何影响拍卖机制的性质；机制设计背后的假设需要做哪些与之对应的调整?
- 经典的拍卖机制的基本框架长什么样；多个优化目标在这个框架下是如何平衡；面向广告业务新形势下的 AutoBidding 模式，拍卖机制该如何重新思考面向未来?

1. 初识广告拍卖机制和相关博弈论基础知识

1.1 为什么说拍卖机制是广告与搜推最本质的差异点

先从本文开头提到的“为什么拍卖机制是广告与搜推最本质的差异之处”说起。以电商场景为例，搜索推荐涉及到免费资源位的高效分配问题，广告涉及到付费资源位的高效分配问题。以最基础的单目标价值最大化为例，搜推的资源位分配按照 GMV 期望最大化排序，广告的资源位分配按照 CPM 期望最大化排序，其中 $pCTR$ 和 $pCVR$ 是模型预估分，bid 是广告主报价（如果是系统自动出价，则 bid 是 $pCVR$ 的相关函数）。可以看出广告领域中广告主可以通过调整自身的 bid 报价策略使得自身的竞争力发生改变，从而影响资源位的分配结果，然而搜推领域的分配结果商家没有办法直接干预，纯粹由平台决策。所以从分配结果来看，搜推的主角是平台，但是广告的主角是广告主。

更为重要的是，广告除了资源位分配环节以外还有广告主扣费环节，以 GSP 机制为例，赢得第一个资源位的广告主需要付费 ，广告主又会根据实际付费情况核算营销表现是否符合预期，从而进一步影响下一轮报价策略。如此往复，广告主个体竞价策略变化的相互博弈最终形成了广告系统整体的收敛分配结果，这个博弈收敛的过程也是搜推领域不需要考虑的。所以广告的拍卖机制包括资源位**分配**和广告主**扣费**两个环节，如何设计能够促使广告主的竞价博弈收敛结果是符合平台引导预期的拍卖机制是重点也是难点。

1.2 广告拍卖机制相关的博弈论基础知识

提到博弈过程和收敛结果，就不得不引出博弈论的相关知识，因为博弈论是一门独立的学科，广告的拍卖机制仅仅是博弈论的一个应用案例，所以下文讲述的侧重点是博弈论在广告拍卖机制中的应用。何谓机制？机制就是设计者想方设法让参与者做设计者想让他们做的事情，手段就是利用参与者的各自偏好，引入博弈环境，使得博弈收敛后的均衡结果符合设计者初衷。注意，参与者必须是智能体，体现在理性和有能力权衡偏好得失，且智能体的偏好往往是**私有信息 Private Value**（有些地方称作 **Type 类型**）、外界不可知，所以好的机制设计需要具备偏好诱导能力，提供某种激励方式使得智能体在博弈环境中的真实表达是他的最优策略，这样单个个体的行为结果可预期，整体博弈收敛结果可以有导向性。总结来说，机制设计有几个要点：

- 智能体有偏好需求：广告主就是智能体，他的营销偏好是理性且私有的，常

见偏好模式有**效用最大化（Utility Maximizer）**和**价值最大化（Value Maximizer）**，机制设计之初就需要先确定智能体的偏好模式，后续才能有针对性地设计偏好诱导方式，下文会结合具体技术再详细阐述。

- 设计者有引导目标：广告平台就是设计者，他的引导目标是明确的，常见目标有**社会福利最大化**和**平台营收最大化**，即引导的博弈均衡结果（包括分配结果和扣费结果）是有设计初衷的。下文会按照两条迭代主线展开介绍。
- 偏好诱导激励相容：**激励相容（Incentive Compatibility）**就是鼓励竞拍者讲真话，使得竞拍者目标和平台目标可以同向发力，有两个标准：1）**优势策略激励相容（Dominant-Strategy Incentive Compatibility，简称 DSIC）**，不管其他智能体如何报告自己的私有信息，如实报告（即讲真话）是每个智能体的最优反应，所谓最优反应就是如果不这么做就会有损失；2）**贝叶斯纳什激励相容（Bayes-Nash Incentive Compatibility，简称 BIC）**，如果其他智能体是如实报告的，那么你的最优反应也是如实报告。
- 博弈结果存在均衡：博弈结果可以有一个均衡，也可以有多个均衡，关键不能是非均衡，起码有确定性的**纳什均衡 Nash Equilibrium** 存在。激励相容引导的均衡可以是较弱的**贝叶斯纳什均衡 Bayesian Nash Equilibrium**，也可以是严格的**优势策略均衡 Dominant Strategy Equilibrium**（可进一步细分三个版本，强、弱和极弱等，这里不再进一步介绍）。

以最常用的囚徒困境为例，介绍一下纳什均衡的应用。有两个智能体囚徒，分别都有两个策略：抵赖 NC 和招供 C，表格里的数字表示囚徒在不同的环境下选择不同的策略可以获得的效用 utility，即利益偏好，这里数字表示判刑多少年。可见 (C, C)->(−5, −5) 是**纳什均衡 Nash Equilibrium，纳什均衡的意思是任何智能体单方面偏离自己的均衡策略均无利可图，当下策略是其他智能体均衡策略的最优反应。**(C, C)->(−5, −5) 状态表明不会有哪个囚徒会单方面改变自己的策略，因为只要对方不动，自己改变都会让自己利益受损，判刑年数更长。另外需要注意，纳什均衡仅保证参与人不会单方面偏离，但不能保证其他人或者大家共同偏离，而且纳什均衡结果不一定是整体收益最大化，显然该例子如果两个囚徒选择共同偏离、合谋采取 (NC, NC) 策略，(−2, −2) 可以使得整体收益最大。

智能体1/智能体2	策略NC	策略C
策略NC	−2，−2	−10，−1
策略C	−1，−10	−5，−5

综上该小节内容，广告拍卖机制涉及到的博弈论要研究的内容就是新设计的机制能否达到纳什均衡；达到的纳什均衡属于什么强度；均衡唯一还是均衡多值；收敛性能如何；收敛结果能否达到预期目标最大化。

1.3 理想化的拍卖机制和理想照进现实的松弛策略

前文已经介绍基于博弈论的机制设计原则，那么最理想的广告拍卖机制需要满足哪些性质？主要有 3 个：

- 高动机保证，**优势策略激励相容（DSIC）**即如实报价是优势策略，注意均衡不仅存在，而且是以最严格的**优势策略均衡（DSE）**的形式存在；
- 高效果保证：均衡结果满足引导预期，社会福利最大化或平台营收最大化；
- 高效率保证：多项式时间（一般近似线性）内完成分配和扣费两个计算过程。

如何实现理想化的拍卖机制？大体思路可以用一句话概括：先定**分配规则（Allocation Rule）**，再定**扣费规则（Payment Rule）**。展开来说：

- 先假设所有竞拍者如实报价，则如何设计分配规则使得上述**高效果保证**和**高效率保证**都成立？
- 再假设得到上述分配结果之后，则如何设计扣费规则使得**高动机保证**成立？

Myerson 引理可以将上述实现理想化拍卖机制的抽象思路具象化，介绍 Myerson 引理之前需要先了解 3 个概念：

- **直接显示机制（direct-revelation mechanism）**：要求竞价者如实报价的机制，但是该机制能否达到均衡，更别说能否达到优势策略均衡，不在要求范畴；
- **可实施的分配规则**：对于一个分配规则 x，如果存在一个扣费规则 p，使得直接显示机制 (x, p) 是 DISC，那么就称这个分配规则 x 是可实施的；
- **单调的分配规则**：当其他竞价者报价不变，当前竞价者的分配函数是报价的单调非减函数，即报价越高，广告主可以获得更高位置更高点击率。

Myerson 引理基于上述概念提供了具体可操作路径：

- 一个分配规则是可实施，当且仅当它是单调的；
- 如果分配规则是单调的，那么存在唯一的扣费规则，使得**直接显示机制**是 DSIC 的，且这个扣费规则有明确的表达式。

Myerson 引理前者说明可实施的分配规则和单调的分配规则是等价的，言下之意就是设计分配规则非常清晰，只要满足单调性即可；后者说明一旦分配规则确定了，扣费规则也是唯一的，且有明确的解析式，这个解析式推导过程可以详见论文，具体形式会在下文详细介绍。按照 Myerson 引理进行机制设计，理想化的广告拍卖机制就可以应运而生。

然而理想照进现实，当拍卖环境变得复杂，需要权衡的因素变多时，理想化的三个性质很难同时得到满足，此时有选择性地适当松弛很有必要。这里给出经典的松弛思路：

- **高效率保证是最基本要求，通常不做松弛。** 原因是计算广告领域往往会面对大规模高并发请求，低时延的高性能计算能力是服务可靠性的基本保障；
- **高效果保证是第一个允许松弛的性质，主要涉及到分配规则。** 因为最优分配问题本质是一个背包问题，而背包问题是 NP 困难问题，分配规则不可能在多项式时间内被实现，所以往往采用启发式的贪心算法，也就是业界常用的排序策略。这样分配问题就转化为了排序问题，例如本文开头 Rankscore 计算公式的设计，可以获得高效率的近似最优解。
- **高动机保证是第二个允许松弛的性质，主要涉及到扣费规则。** Myerson 引理证明当分配规则确定，扣费规则就是唯一的、且有明确的表达式，假设该表达式是 a。可是实际环境非常复杂，或因为计算效率问题，或因为业务约束问题，导致实际采用的扣费规则没有用 a，却用了 b，那么 DSIC 就会被破坏。但是非 DSIC 机制就一定很糟糕吗？答案非也。前文提到，**优势策略均衡**是最理想的均衡结果，机制设计的底限是要有均衡的预期，有均衡预期就有足够假设来预测智能体的行为，那么机制结果就能有效推演。所以非 DSIC 机制也可以有不错的均衡性质，它们在实际中很常见。例如广告拍卖机制中，假设广告主偏好模式是 **Utility Maximizer**，平台优化目标是社会福利最大化，那么只有 **VCG 机制**是 DSIC，**GSP 机制**不是 DSIC，但是也不妨碍 GSP 机制盛行，详细内容下文会重点介绍。

1.4 广告拍卖机制的类型划分与经典机制设计范畴

拍卖机制设计前提是需要对拍卖形式有一定规约，广告拍卖形式有很多不同的视角划分方式：

- 按照拍卖品个数划分，分为**单物品拍卖**和**多物品拍卖**。单物品拍卖理论研究完

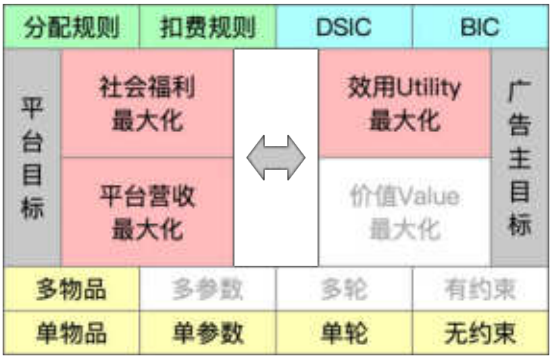
备，机制设计空间较小，可直接应用于实践，而多物品拍卖机制设计空间大，需要对理论做很多假设才能平衡好效果与性能。广告往往有多个付费资源位在一次请求下同时拍卖，所以广告拍卖形式属于多物品拍卖，而且是多个非相同物品拍卖，位置越好点击率越高，对应的价值就越大。

- 按照竞拍者报价个数划分，分为**单参数拍卖**和**多参数拍卖**。报价个数对应竞拍者的估值对象个数，如果广告主只需要报价流量价值，那么就是单参数拍卖；如果广告主需要同时上报多个估值例如不同资源位的流量价值，那么就是多参数拍卖。和上一个划分方式面临的问题一样，参数越多，机制设计空间就越大，无论对理论设计还是计算性能都有很大挑战。
- 按照竞拍轮数来划分，分为**单轮拍卖**和**多轮拍卖**。轮数越多，竞拍者的私有信息 Type 会越来越成为公有信息，使得竞拍者能有更多的信息做决策从而调整报价策略。虽然多轮拍卖更加契合实际广告拍卖场景，广告主一笔预算往往会参与很多轮的请求报价，但是多轮的机制设计比单轮复杂很多，一般简化为单轮拍卖形式进行分析讨论。
- 按照广告主优化目标是否带约束来划分，分为**无约束拍卖**和**有约束拍卖**。有约束是指在广告主追求营销目标最大化的同时会带有指标约束，例如预算约束或者 ROI 约束等，这样广告主报价策略会从短期收益最大化调整为长期收益最大化，这对机制性质也会带来不小挑战。

可见，上述划分标准每一类的后者更加接近广告拍卖实情，但是相比前者会让整个机制设计难度系数大幅提升，所以本文将广告的经典机制设计范畴规约在“异质多物品、单参数、单轮和无约束”拍卖形式下（简称多物品拍卖），这也是目前工业界实践最为成熟的版本之一。其中多物品的拍卖形式没有再进一步简化了，但是下文讲述逻辑依然会从单物品拍卖开始说起，这样便于理解逐步过渡到多物品拍卖，近似理想的拍卖机制该如何调整与设计。

另外，前文提到的广告主优化目标即偏好需求的划分标准（Utility 最大化 vs Value 最大化）和广告平台优化目标的划分标准（社会福利最大化和平台营收最大化）每一项都是广告拍卖实情，属于可选项，不存在升级的说法，但是广告主偏好建模 Value 最大化是目前较新的需求类型，本文暂且没有归类到经典机制设计范畴，后续会作为新业态下的机制设计进行介绍。所以下文的讲述逻辑会基于广告主 Utility 最大化即利润最大化的前提，分别介绍面向社会福利最大化和平台营收最大化的拍卖机制该如何设计。

经典广告拍卖机制设计范畴



2. 社会福利最大化的有效机制

2.1. 广告主和平台优化目标的各自确立

本文以点击付费广告类型为例（可以推广至任意出价类型广告产品），广告主对每个点击流量内心有明确的私有估值 v，对外报价 b 采取策略是 v 的函数即 $b = f(v)$，可以是 v 的如实报价，也可以对 v 有所隐瞒，关键看哪个策略对自己是有利的。机制设计首先需要确定竞拍者的偏好模型，本文采用经典的效用 Utility 最大化，即广告主追求利润最大化。如果广告主竞得该次流量对应的系统扣费为 p，则效用函数是简单拟线性函数 $u = v - p$，利润 = 价值 − 成本。理想的拍卖机制就是要诱导广告主的报价 b 是私有估值 v 的真实反映，即 $b = v$，使得博弈收敛均衡下广告主追求目标效用 u 是最大的。

另一方面，平台目标视角面向**社会福利（Social Welfare）最大化**是指资源位的分配一定要物尽其用，分配给最需要的竞拍者，即分配给最高真实估价的那些广告主，同时扣费机制方面需要诱导广告主如实报价是最优策略。于是机制（包含分配和扣费）的设计首先需要确定分配规则是面向社会福利最大化，即 $SW = \max \sum_{i=1}^{n} v_i x_i$，其中 n 表示竞拍者个数，v_i 是广告主 i 对于点击流量的私有估值（一个点击值多少钱），$x_i = [0, 1]$ 表示广告主 i 赢得当前请求下的付费资源位的点击率（介于 0 和 1 之间）。在确保分配规则是单调（广告主提高报价 b，有机会获得点击率更高的资源位）的基础上，再设计扣费规则使得广告主如实报价是最优策略，即效用 u 可以最大化。

通常，社会福利最大化的机制是对资源最有效率的配置方式，所以该类型的机制也叫有效机制（efficient mechanism）。

2.2. 单物品拍卖的有效机制设计

先从最简单的单物品拍卖场景说起，假设只有一个资源位需要拍卖，有多个广告主同时参与竞拍，Second-price auction（简称 **SP 机制**，也叫 Vickrey auction）是社会福利最大化的 DSIC 机制。**SP 机制**首先确定分配规则，将资源位分配给报价最高的广告主 b_i，使得资源分配效率最高；然后确立扣费规则为二价扣费，除广告主 i 以外的最高报价，令 $B = \max_{j \neq i} b_j$，即扣费 $p_i = B$，使得广告主如实报价 $b_i = v_i$ 可以让自身效用 $u_i = v_i - B$ 最大化，从而让 SP 机制具备优势策略激励相容 DSIC 的性质。

接着简单论证 SP 机制如何保证 DSIC 性质。广告主 i 报价只有两种结果：当 $b_i < B$，则广告主 i 输掉这场竞拍，效用 $u_i = 0$；当 $b_i \geq B$，则广告主 i 赢得这场竞拍，支付价格为 B，效用 $u_i = v_i - B$。然后我们再针对广告主的私有估值进行分情况讨论：当 $v_i < B$，最大效用 $\max\{0, v_i - B\} = 0$，显然广告主选择如实报价，输掉这场竞拍效用最大；当 $v_i \geq B$，最大效用 $\max\{0, v_i - B\} = v_i - B$，显然广告主选择如实报价，赢得这场竞拍效用最大。证明完毕，平台优化目标和广告主优化目标均在优势策略激励相容的性质下得以满足。

2.3. 多物品拍卖的 VCG 机制和 GSP 机制比较

真实广告拍卖场景往往是多个资源位同时竞拍，而且不同资源位点击率的差异使得多物品非同质，单物品拍卖下的有效机制设计无法直接应用过来。我们先从最简单的假设说起，点击率仅和资源位本身有关，不考虑广告本身质量，这也是最早期广告技术的真实落地方式，而且对于机制性质的分析也不失一般性。假设一共有 k 个广告位，点击率分别是 $\alpha_1 \geq \alpha_2 \geq \ldots \geq \alpha_k$，一共有 n 个广告主，报价从高到低排序分别是 $b_1 \geq b_2 \geq \ldots \geq b_n$，一般 $n > k$。如果将广告主 i 分配到广告位 j，则其效用收益是 $\alpha_j(v_i - p_i)$，v_i 是广告主对点击流量的估值，p_i 是广告主的单位点击支付。

前文提到单调的分配规则往往较为容易制定，我们按照报价的大小从高到低排序，第高报价的广告主被分配至第好的广告位，这样平台目标即社会福利最大化就可以得以满足（分配问题贪心启发式转化为排序问题）。难点是扣费规则该如何设计，可以保证 DSIC 且实现广告主目标 Utility 最大化。Myerson 引理论证，满足该性质的唯一支付公式是 VCG 扣费方式，多物品拍卖下的 Vickrey-Clarke-Groves auction（简称 **VCG 机制**）是单物品拍卖下的 Vickrey auction（也称 **SP 机制**）有效升级。单次点击 VCG 扣费公式：，其中 $\alpha_{k+1} = 0$，注意这里分母 α_i 出现是因为按照点击

扣费。VCG 扣费基本原理就是该竞拍者赢得广告位后，给其他广告主带来的收益损失是该竞拍者需要支付的费用。

下图是来自《博弈论二十讲》课件中的图示，个人认为是对 VCG 机制的 DSIC 性质最为直观和形象的表述了，所以就原封不动摘抄过来。为了和本文的参数定义对齐，这里做一下简单的解释：图中 z 表示估值分布；v 是真实估值；b 是报价；$x(z)$ 表示分配结果（对应不同坑位的点击率），仅考虑位置点击率的情况下就是对应本文的 α_i 定义；$p(v)$ 表示扣费结果，红色面积区域是展现视角下的期望点击扣费，所以对应本文的 $p_i\alpha_i$。根据 Utility= 期望价值 − 期望扣费 的定义，用绿色面积来表示 Utility 大小，比较 (g)(h)(i) 三张图的绿色面积，可见当 $b = v$ 即如实报价，广告主效用最大。

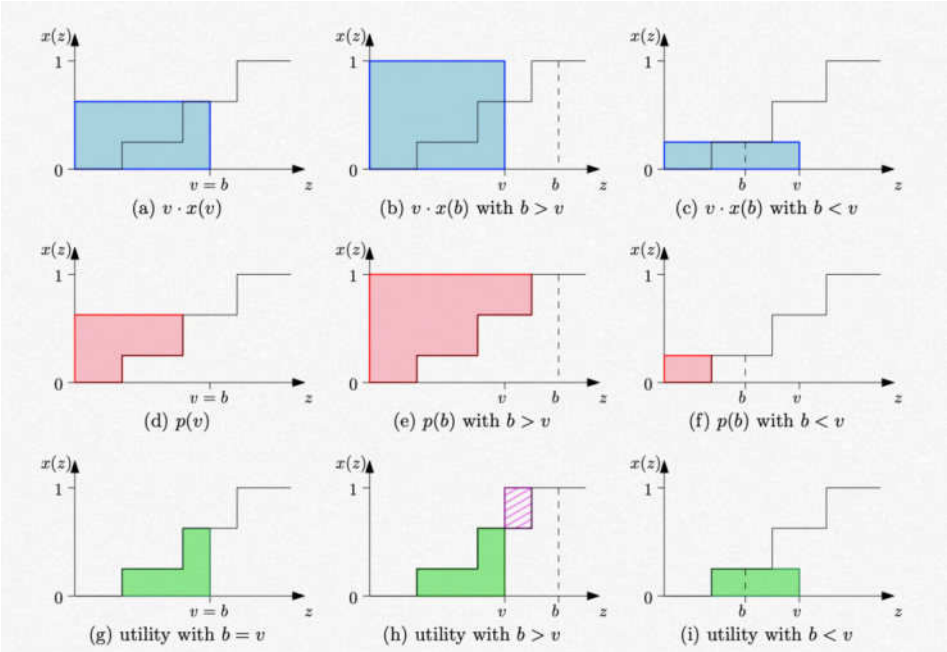
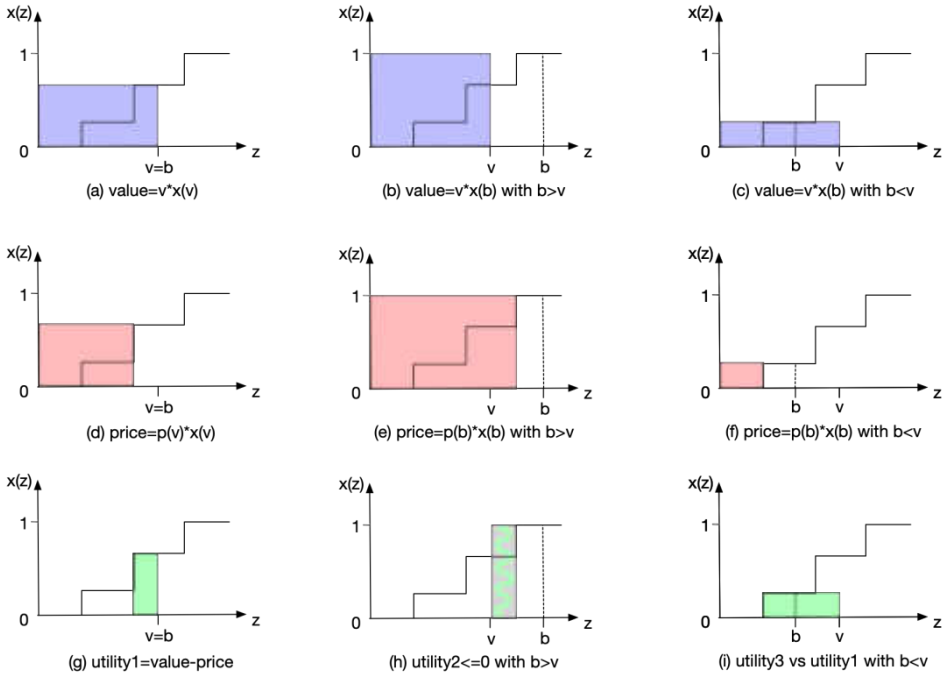


Figure 2: Proof by picture that the payment rule in (6), coupled with the given monotone and piecewise constant allocation rule, yields a DSIC mechanism. The three columns consider the cases of truthful bidding, overbidding, and underbidding, respectively. The three rows show the surplus $v \cdot x(b)$, the payment $p(b)$, and the utility $v \cdot x(b) - p(b)$, respectively. In (h), the solid region represents positive utility and the lined region represents negative utility.

VCG 机制固然可以满足 DSIC 性质，使得广告主目标 Utility 最大化，但是其运行效率较低且给广告主解释成本较高。综合考虑诸多因素，实际系统常常选用 **GSP 机制（Generalized second-price auction）**，分配规则和 VCG 机制一致，扣费规则有差异但更简单，单次点击 GSP 扣费公式：$p_i = b_{i+1}$。GSP 机制虽然看似是 SP 机

制的广义延伸，但不是 DSIC，如实报价不能让广告主效用最大化。举例：有两个广告位点击率如下，$\alpha_1 = 0.5$ 和 $\alpha_2 = 0.4$；有三个广告主对点击流量的私有估值如下，$v_1 = 200$、$v_2 = 180$ 和 $v_3 = 100$。假设另外两个广告主均如实报价，若广告主 1 也如实报价 $b_1 = v_1$，则可以赢得第一个广告位，根据 GSP 机制扣费 $p_1 = b_2 = 180$，他的效用收益为。但是如果其隐瞒报价 $b_1 = 110 \neq v_1$，虽然只能赢得第 2 个广告位，根据 GSP 机制扣费 $p_1 = b_3 = 100$，但是其效用反而更大。所以 GSP 机制下广告主有隐瞒价值出低价的动机（under-bidding incentive）。

下图是参考《博弈论二十讲》关于 VCG 性质图示的方法，相同定义方式下画了 GSP 机制的性质表现。同样紫色面积表示期望价值，红色面积表示 GSP 机制下的期望扣费，绿色面积表示期望效用，可见：1）GSP 机制不是 DSIC，即如实报价 $b = v$ 的绿色面积不一定是最大的；2）高报价 $b > v$ 的绿色面积是负数，表示高报价没有正向收益；3）低报价 $b < v$ 绿色面积和如实报价的绿色面积两项比较孰高孰低不一定，存在获利空间，所以 GSP 机制下广告主有 under-bidding incentive。



虽然 GSP 机制在广告主 Utility 最大化的目标下并不是 DSIC 的，但是它也有不错的均衡性质，包括两个视角的均衡：Pure Nash Equilibrium under Full-Information Setting 和 Bayesian Nash Equilibrium under Partial-Information Setting。以

前者为例，假设多轮竞价交互之后广告主的估值成为公有信息，GSP 机制达到的一种均衡结果是广告主会认为保留当前位置是最优策略，不愿意与比他顺位高的人换位置，不然利润受损，这种均衡叫 locally envy-free equilibrium（也称 symmetric Nash equilibrium），对应广告主的报价策略不再是如实报价，详细可见文献。

GSP 机制除了有不错的均衡性质以外，还有其他优势：1）计算高效、对客解释成本低、扩展性好；2）最新研究论证在广告主 Value 最大化的目标下 GSP 机制又是 DSIC 的，这点会在文章最后提及；3）GSP 存在多个均衡，其中包含一个均衡结果和 VCG 等价，因为扣费大于 VCG，所以平台营收会比 VCG 更高。后续技术演进都是在 GSP 机制框架下进行。

2.4. 点击率因素如何影响 GSP 机制升级

前文提到的 GSP 机制是最基础的版本也是互联网公司落地最早的一个版本，但是点击率仅与资源位有关的假设显然和实际情况相差甚远，对于点击付费的广告类型，离谱的点击率假设会让资源位的配置效率大打折扣。为此，引入广告质量本身的点击率 β 是非常自然而然的想法。关于资源位点击率 α 和广告质量点击率 β 的关系，主要有两个大版本的升级。

基础假设是整体点击率关于位置因素可分离，即 $CTR = \alpha\beta$，GSP 机制升级为 **wGSP**（weighted GSP，weighted 体现在广告质量分的引入）。这样平台优化目标变成 ，n 是广告主个数，k 是资源位个数；广告主优化目标变成 $u_i = \beta_i\alpha_j(v_i - p_i)$，表示广告 i 分配至资源位 j 效用函数；分配规则是按照 βb 进行排序，第 i 大的广告分配至第 i 好的资源位；扣费规则是 $p_i = \frac{\beta_{i+1}b_{i+1}}{\beta_i}$。如此升级可以论证机制性质保持不变，但因为点击率和真实情况更为接近，SW 表现更好。实际真实点击率无法提前预知，往往需要有点击率预估的能力。wGSP 机制要求除了具备预估粗粒度的位置点击率以外，系统还要有能力预估细粒度的广告点击率。

升级假设是整体点击率关于位置因素不可分离，即位置点击率和广告质量点击率无法解耦，这个假设与现实更加接近，两方面原因：1）不同广告在不同位置上的点击率影响各异；2）不同广告在不同上下文影响下的点击率各异。前文提到的 VCG 机制满足 DSIC 性质的隐含假设是位置因素可分离，如果遇到位置因素不可分离则满足 DSIC 性质的机制就会演变成 Laddered Auction，详细内容参见文献。同样对面这种情况，GSP 机制也需要升级，**iGSP**（iterated GSP）可以最大限度保证 existence of efficient equilibria，但也仅限 2 个资源位同时拍卖，如果遇到 3 个及以上就仍然是

一个 open question，不过还算乐观的情况是移动端的资源位往往较少。iGSP 的分配规则是既然位置点击率不可分离，那就逐个坑位顺次拍卖。先根据经验选择一个拍卖序，可以假定从高到低，针对当前资源位 j 考虑位置因素和上下文因素重新计算每个广告主的点击率 γ_{ij}，获得当前资源位下的广告新排序；接着执行扣费规则，对首位广告主进行二价扣费，$p_{1j}=\frac{\gamma_{2j}b_2}{\gamma_{1j}}$；然后将首位竞得广告主从队列内去除，相同逻辑继续执行下一个资源位拍卖流程；如此遍历迭代，直至资源位竞拍完。以上通过点击率假设和预估能力的不断完善，使得 SW 更优，资源配置效率越来越高，市场蛋糕不断做大，接下去作为平台视角，营收最大化也是目标之一，如何合理分配蛋糕也是重要课题。

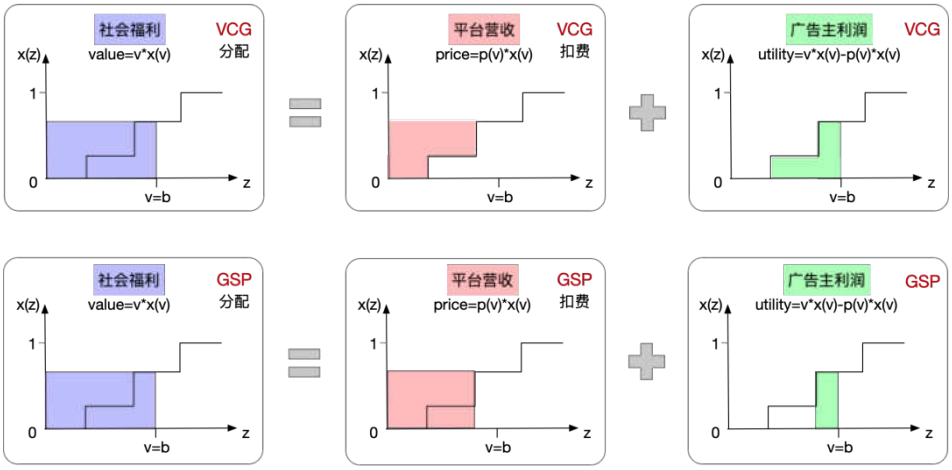
点击广告的拍卖机制类型	VCG	GSP	wGSP	iGSP
平台目标SW最大化	$\max\sum_{i=1}^{n}\sum_{j=1}^{k}v_i\alpha_j$	$\max\sum_{i=1}^{n}\sum_{j=1}^{k}v_i\alpha_j$	$\max\sum_{i=1}^{n}\sum_{j=1}^{k}v_i\beta_i\alpha_j$	$\max\sum_{i=1}^{n}\sum_{j=1}^{k}v_i\gamma_{ij}$
广告主目标Utility最大化	$\max\alpha_j(v_i-p_i)$	$\max\alpha_j(v_i-p_i)$	$\max\beta_i\alpha_j(v_i-p_i)$	$\max\gamma_{ij}(v_i-p_i)$
分配规则	$b_1\geq b_2\geq\ldots\geq b_n$ $\alpha_1\geq\alpha_2\geq\ldots\geq\alpha_k$ $n>k$，从高到低顺次摆放	$b_1\geq b_2\geq\ldots\geq b_n$ $\alpha_1\geq\alpha_2\geq\ldots\geq\alpha_k$ $n>k$，从高到低顺次摆放	$\beta_1b_1\geq\beta_2b_2\geq\ldots\geq\beta_nb_n$ $\alpha_1\geq\alpha_2\geq\ldots\geq\alpha_k$ $n>k$，从高到低顺次摆放	1. 确定一个拍卖序，资源位拍卖顺序可以打乱，先验做到best to worst 2. 遍历拍卖序中的每个资源位，当前资源位 j 下重新计算点击率，获得新序 $\gamma_{1j}b_1\geq\gamma_{2j}b_2\geq\ldots$ 3. 对排名首位广告主 $p_{1j}=\frac{\gamma_{2j}b_2}{\gamma_{1j}}$ 4. 将首位竞得广告主从队列内去除 5. 如上遍历迭代完每个资源位拍卖
扣费规则	$p_i=\sum_{j=i}^{k}b_{j+1}\frac{\alpha_j-\alpha_{j+1}}{\alpha_i}$	$p_i=b_{i+1}$	$p_i=\frac{\beta_{i+1}b_{i+1}}{\beta_i}$	
激励相容性质	DSIC	非DSIC，但有 locally envy-free equilibrium	非DSIC，但有 locally envy-free equilibrium	非DSIC，2个资源位尚能保证 existence of efficient equilibria； 3个及以上仍属open question
广告主报价策略	$b_i=v_i$	under-bidding incentive $b_i\leq v_i$	under-bidding incentive $b_i\leq v_i$	\
点击率假设	仅考虑位置点击率	仅考虑位置点击率	位置点击率和广告质量点击率可分离	位置点击率和广告质量点击率不可分离

3. 平台营收最大化的最优机制

3.1. 通过对比 VCG 和 GSP 来看平台营收空间

前文介绍了面向社会福利最大化的机制该如何设计，即把资源位**有效地**分配给高估值的广告主，使得整体价值最大化。整体价值通过扣费规则一分为二，一边是对广告主的扣费形成平台营收，另一边是广告主留下的效用即利润，即 Value=Utility+Price。以下图 VCG 和 GSP 机制对比为例，假设广告主都选择如实报价，因为两者分配规则相同所以社会福利对应的面积大小也相同，但是因为两者扣费规则不同所以两者的平台营收和广告主利润就会有差异。从图示面积一目了然，相比 VCG，GSP 的平台

营收更高，同时与之对应的广告主利润就会更低，所以从平台营收视角来说 GSP 机制更受欢迎。



这就是一个分蛋糕的过程，如果平台收益变多了，那么广告主收益就会变少。当然，分蛋糕也不是一个简单随意修改扣费规则的事情，是不是找个小暗门扣费多一点，平台营收就能多一些，然后就解决问题了？其实不然，扣费规则和激励相容性质息息相关，前文提到即使松弛了 DSIC 性质，但是均衡结果是最起码的要求，否则从长期博弈结果来看短期的平台营收都不可持续。换个角度解释这个现象，GSP 机制不是 DSIC 的，广告主有隐瞒估值低报价的动机，所以长期来看会随着广告主低报价整体社会福利不如 VCG，使得虽然在分蛋糕方面 GSP 有优势但在做大蛋糕方面存在不利因素，此消彼长，长期看平台营收 GSP 一定会打折扣。GSP 有不错的均衡性质尚且如此，更何况其他粗暴魔改扣费规则的机制了，所以事情看来没有那么简单。

相较社会福利最大化（简称 SW）通过分配规则就可以直接达到平台优化目标的运作逻辑不同，平台营收最大化（简称 Rev）则需要结合分配和扣费规则一起才能实现（因为只有知道了扣费金额才能知道平台营收），这对机制设计带来不小挑战，以点击广告为例，ppc 是表示单次点击扣费，SW 类似于 $\max ctr * bid$，Rev 类似于 $\max ctr * ppc$。SW 的扣费规则只需要承担激励相容要求即可，但是 Rev 的扣费规则除了激励相容的部分还要承担平台目标，而且之前分配规则和扣费规则都是解耦地有先后地分别设计，此时两者却要耦合在一块儿共同围绕相同目标一起设计。**最优机制（optimal mechanism）**的出现就是来解决这个问题，它可以使 Rev 机制的设计范式像 SW 一样清晰明了，还能保证 DSIC 性质。

3.2. 单物品拍卖的最优机制设计

先从最简单的单物品拍卖说起，并且进一步简化假设只有一个竞拍者，该设定因为没有其他竞拍者和他竞争，所以如果平台不做点其他的设置，竞拍者可以瞒报自己估值 v，以一个很低的报价竞得这个物品，导致即使平台按照一价扣费，营收也会大大受损。为了解决这个问题，平台可以设置一个**保留价** r，类似虚拟竞拍者角色。如果竞拍者报价 b 高于保留价，则竞得物品，平台按照二价扣费，即收取保留价费用；如果竞拍者报价低于保留价，则物品流拍。这样最核心的问题就变成了保留价该设置多少才能使得平台营收最大化？显然，如果保留价 r 刚好比竞拍者私有估值 v 小一点点，即 $r \lesssim v$，竞拍者又如实报价，这样物品可以竞得，并且平台收取的费用和竞拍者私有估值相当，也就是说平台赚取了估值中的绝大部分，竞拍者仅保留微乎其微的利润 $(v-r) \gtrsim 0$，但毕竟利润还是正的，竞拍者也只能勉强接受。

上述保留价的取巧设计，可以使得平台营收最大化，前提需要猜准竞拍者的私有估值，毕竟估值是私有信息，平台无从准确知晓。可以借助贝叶斯分析方法，假设该估值服从某个分布 F，则竞拍者如实报价可以竞得物品的概率是 $1-F(r)$，那么平台的营收期望就是 $E(Rev)=r\cdot(1-F(r))$。举例来说，如果 F 是 $[0,1]$ 均匀分布，则 $F(r)=r$，$\max r\cdot(1-r)$ 通过求导梯度为 0 的方式得到 $r=1/2$，则 $E(Rev)=1/4$。同样计算方法继续求解稍微复杂一点的两个竞拍者场景：如果采用 DSIC 的 **SP 机制**，假设两个竞拍者的私有估值独立且服从 $[0,1]$ 均匀分布，此时平台期望营收为 $1/3$；如果采用 **SP 机制**配合**保留价**，上述假设基础上增设保留价 $r=1/2$，如实报价最高者竞得物品并支付第二名报价和保留价中更高的价格，此时平台期望营收为 $5/12$。

那么有没有其他办法可以获得更高的平台收益？换一个保留价，或者换一个拍卖形式？ **Myerson 机制**就是可以获得最大化期望收益的最优机制。首先定义虚拟估值，它和私有估值以及分布有关；然后可以论证在满足 DSIC 的机制空间上对期望收益最大化，可以等同于在同一个空间对期望**虚拟福利最大化**，这样原本关心的收益最大化问题就转变为了分配最优化问题，与前文 2.1. 章节中 $SW=\max\sum_{i=1}^{n}v_ix_i$ 相比仅仅是将私有估值 v_i 转换为对应的虚拟估值 $\varphi_i(v_i)$，其他分配规则的细节以及扣费规则都与**社会福利最大化**一模一样；最后只需要保证 $\varphi_i(v_i)$ 是单调的，即私有估值的分布 $F_i(v_i)$ 是**正则分布**，许多常见分布基本都是正则分布，例如 normal、lognormal、uniform and exponential distributions 等；非正则分布包括多峰分布和长尾分布，分配的单调性是机制 DSIC 性质的前提。论证过程可详见文献。

可以发现虚拟估值 $\varphi_i(v_i)$ 可能为负，在分配的过程中如果遇到 $\varphi_i(v_i)$<0 的竞拍者会被剥夺竞拍资格，如果所有竞拍者的虚拟估值均为负，则流拍，所以 $\varphi_i^{-1}(0)$ 即虚拟估值为 0 的逆函数起到了**个性化保留价**的功能。如果假设所有竞拍者的私有估值服从**独立但不同分布**，则不仅会有各自不同的保留价，还会遇到如实报价情况下报价高，但虚拟估值不一定高，导致没有竞得更好的资源位，也好理解，此时虚拟估值代表了个性化自己和自己对比的报价意愿强烈程度；如果假设所有竞拍者的私有估值服从**独立同分布**，因为估值分布相同则虚拟估值函数 φ 也相同，又因为估值分布 F 服从正则分布，则 φ 严格递增，那么虚拟估值最高的竞拍者就是私有估值最高的竞拍者，这样虚拟福利最大化机制就和带有保留价 $\varphi_i^{-1}(0)$ 的二价机制相同，回答上文提到的两个竞拍者估值服从独立同分布 $[0,1]$ 均匀分布的例子，$\varphi_i^{-1}(0)=1/2$ 确实是最优保留价。以下是单物品拍卖的几个机制对比：

单物品拍卖DSIC机制类型	Second Price机制	Second Price+保留价机制	Myerson机制
平台优化目标	社会福利最大化	平台营收最大化	平台营收最大化 等同于虚拟福利最大化
分配规则	$b_1>b_2$ then b_1 竞得	if $b_1<r\&b_2<r$ then 流拍；else $b_1>\max\{b_2,r\}$ then b_1 竞得； 其中 $r=\varphi^{-1}(0)$	if $\varphi_1(b_1)<0\&\varphi_2(b_2)<0$ then 流拍；else $\varphi_1(b_1)>\max\{\varphi_2(b_2),0\}$ then b_1 竞得；
扣费规则	$p_1=b_2$	$p_1=\max\{b_2,r\}$	$p_1=\varphi_1^{-1}(\max\{\varphi_2(b_2),0\})$
广告主私有估值分布假设	\	独立同分布 $F_i(v_i)$ 是正则分布	独立但不同分布 $F_i(v_i)$ 是正则分布

3.2. 多物品拍卖的经典保留价方案与 Squashing 方案

虽然 **Myerson 机制**可以完美解决单物品拍卖场景下的平台营收最大化问题，但是面对多物品拍卖场景该机制性质的理想化状态就很难保持，更大方面来说目前整个多物品拍卖领域关于平台营收最大化的最优机制设计问题依然是一个开放问题。回到广告多坑位拍卖场景，经典的机制设计会围绕 GSP 机制做针对性改造，将 Myerson 机制的思路和 GSP 机制的流程相融合，在保持良好机制性质的前提下持续优化平台营收，获得近似最优结果。接下来会介绍 3 种基于 wGSP 机制改造的提升平台营收的常见方法。

第一种是 **mGSP（myerson GSP）机制**，想法很直接，就是将 Myerson 保留价

技术直接应用到 wGSP 机制上。和单物品拍卖的设计思路类似，Myerson 保留价最核心的虚拟估值依赖广告主的私有估值分布，系统可以酌情根据独立同分布的颗粒度进行假设的调整，例如客户之间完全独立不同分布、相同搜索词的客户之间独立同分布、整体流量客户独立同分布等。因为广告系统是多轮拍卖系统，假设广告主是如实报价，则私有估值分布可以根据广告主历史报价进行不同粒度的分布拟合，从而得到虚拟估值。有了虚拟估值和保留价，在分配规则方面有两种类型可做尝试：Eager 模式，先过滤后排序，即先根据保留价过滤没有竞争力的广告主，再按照虚拟估值排序；Lazy 模式，先排序后过滤。Eager 模式对最高报价无法胜出保留价的情形更友好，Lazy 模式对第 2 高报价无法胜出保留价的情形更友好，多数情况下 Eager 模式在提营收方面会比 Lazy 更有优势。

第二种是 **aGSP（anchoring GSP）机制**，是对 **mGSP 机制**一种松弛方式，业务落地过程中虚拟估值的求逆计算和私有估值的正则分布要求等均会让机制实现不够简洁且解释性较差，略牺牲效果的情况下精简机制设计也是可接受的迭代方向。首先保留价的计算方法和 mGSP 机制相同，一般保留价的颗粒度会选择数据积累较为充分的粒度。其次主要差异点就是将虚拟估值由简化为 $\varphi_i(v_i) = v_i - r_i$，如果假设私有估值分布是均匀分布，化简的表达式就非常接近，求逆也方便，而且业务含义与 myerson 思路很吻合，报价竞争力更多看重每个广告主超出个性化保留价的那部分。

第三种是 **rGSP（reserve GSP）机制**，是对 **aGSP 机制**更加进一步的松弛。因为有文献提出，为了解决私有估值和虚拟估值趋势不一致问题，虚拟估值可以仅用于个性化保留价的过滤，分配和扣费依然按照私有估值进行，这样虽然对营收有折损但是机制实现更加简洁且易解释。

另外一类不是在报价因子上做文章，而是在 wGSP 的广告质量点击率上做文章，通过对广告质量点击率进行挤压，也能达到提升营收的效果，这类机制叫做 **sGSP（squashing GSP）机制**。分配规则是按照 $\beta_i^s b_i$ 进行排序，其中 s 就是挤压因子，扣费规则按照。该挤压因子能够体现三方面的结论：1）Efficiency 视角：当 $s = 1$ 是社会福利最大化及 Efficiency 最大化，凡是 $s \neq 1$ 的调节都提升平台的短期收入，从而影响广告主价值；2）Relevance 视角：只要 s 往大调节，整体点击率就会提升，相关性就会变好；3）Revenue 视角，当相邻广告质量随排序递减即 $\frac{\beta_{i+1}}{\beta_i} < 1$，在不改变排序的情况下调小 s，p_i 变大，利好营收增加，当相邻广告质量随排序递增即 $\frac{\beta_{i+1}}{\beta_i} > 1$，在不改变排序的情况下调大 s，p_i 变大，利好营收增加。这一方案为了进一步挖掘营收空间，可以升级为分位置调控挤压因子等，具体详见文献。

多物品拍卖提升平台营收机制方案	sGSP	rGSP	aGSP	mGSP
分配规则	$\beta_1^* b_1 \geq \beta_2^* b_2 \geq \ldots \geq \beta_n^* b_n$ $\alpha_1 \geq \alpha_2 \geq \ldots \geq \alpha_k$ $n > k$, 从高到低顺次摆放	$Let\ S = \{i : b_i > r_i\}$ $\beta_1 b_1 \geq \beta_2 b_2 \geq \ldots$ $\alpha_1 \geq \alpha_2 \geq \ldots$ 从高到低顺次摆放	$Let\ S = \{i : b_i > r_i\}$ $\beta_1(b_1 - r_1) \geq \beta_2(b_2 - r_2) \geq \ldots$ $\alpha_1 \geq \alpha_2 \geq \ldots$ 从高到低顺次摆放	$Let\ S = \{i : \varphi_i(b_i) > 0\}$ $\beta_1\varphi_1(b_1) \geq \beta_2\varphi_2(b_2) \geq \ldots$ $\alpha_1 \geq \alpha_2 \geq \ldots$ 从高到低顺次摆放
扣费规则	$p_i = (\frac{\beta_{i+1}}{\beta_i})^* b_{i+1}$	$p_i = \frac{\beta_{i+1}}{\beta_i} \max\{r_i, b_{i+1}\}$	$p_i = \frac{\beta_{i+1}}{\beta_i} \cdot b_{i+1} + \frac{\beta_i r_i - \beta_{i+1} r_{i+1}}{\beta_i}$	$p_i = \varphi_i^{-1}(\frac{\beta_{i+1}}{\beta_i} \cdot \varphi_{i+1}(b_{i+1}))$
优化项	作用在广告质量点击率 β	作用在报价 b ，保留价 r 的构造可以根据个性化程度酌情选择颗粒度大小，保留价的嵌入程度也酌情选择。		

4. 经典广告拍卖基本框架和预告进阶篇

4.1. 经典的广告拍卖基本框架

前文介绍了机制性质不算完美但还算良好的面向平台营收的拍卖机制，但也明确强调平台营收最大化是一个分蛋糕的过程，它其实是和广告主零和博弈相互争夺利润，一味提升平台营收短期来看没有问题，但长期来说一定会影响广告主的预算投入，从而影响社会福利最大化即做大蛋糕的结果。所以我们需要在两者之间做平衡，把握好做大蛋糕和分好蛋糕的节奏。同时，用户体验也是广告系统需要兼顾的重要考量，和免费资源位相比，付费资源位或多或少会牺牲用户体验换取社会福利，如果不加以约束，长期以往用户会失去对付费资源位的关注度，从而平台会彻底丧失流量变现的机会。

综上，以一个开放生态的视角看待广告系统，广告主和用户都会用脚投票，进行不同平台之间的选择，所以平衡好三者之间的利益述求就极其关键。那么经典的广告拍卖基本框架就是一个带约束的最优化问题：max 社会福利最大化；s.t. 平台营收约束、用户体验约束（电商场景可以是点击率、转化率和相关性等）。然后通过拉格朗日对偶法将分配问题转变为多目标排序问题，同时扣费规则明确为关键最小扣费，即广告主仅需要支付保持住当前广告位的最小费用。这样一套广告拍卖机制可以在业务发展的不同阶段择机调整适配，能够促进广告生态健康发展。

4.2. 广告拍卖进阶版展望

本文从机制设计原理出发，不断加强前提假设，改造机制内容，逐渐完善广告拍卖的基本框架，至此经典范畴的内容介绍完毕。但是随着广告业务和技术的不断发展，新趋势孕育新机会，主要有三个方向的持续升级：

- **业务方面，广告主优化目标的改变：**广告主的营销述求不再以利润 Utility 最大化为第一优化目标，而是以价值 Value 最大化为主要优化目标；扣费不再是广

告主的敏感成本项，而是一个必须要花完的预算，核心关注这笔预算花完能否最大化价值，当然了如果获得相同价值的前提下可以考虑少付费。那么一旦广告主偏好模型发生变化，现有机制就需要重新审视，哪些机制的 DSIC 性质会发生改变，是否需要设计新机制来满足新要求?

- **广告主方面，Autobidding 模式的兴起：**越来越多的 DSP 开始智能化承接广告主的营销述求，此时广告主不再对流量进行估值报价，而是以更加高阶的或者直观的优化目标形式（最大化进店等）出现，伴以业务约束要求（预算和 ROI 约束等），然后由智能报价系统进行在线实时地对每个流量进行自动报价。面对使用占比越来越高的 Autobidding 模式，拍卖机制该做哪些假设的调整，新机制该如何设计才能满足良好性质实现平台优化目标?
- **技术方面，数据驱动的机制设**计：数据驱动的深度学习技术已经逐渐成为广告系统其他算法模块（召回和预估等）的标配，而以平台收入最大化为目标的最优机制设计目前仍然有较大的优化空间，经典的设计方案需要依赖较强的领域知识，通过先验缩小优化范围，获得的近似最优解质量不高。我们该如何为机制设计进行定制化改造，满足机制性质的同时借助数据驱动的能力使得最优值的搜索过程更加智能，而且标准化的迭代流程又可以进一步加快技术升级，从而打开优化天花板?
- **生态方面，生成式模型颠覆广告营销模式：**以 ChatGPT 为代表的生成式大模型让科技行业重新兴奋起来，也为广告营销注入了新的想象力。生成式大模型几乎一定会带来用户与互联网产品交互模式的改变，例如，多模态交互式对话方式会取代搜索引擎的地位，以广告位拍卖为基础的互联网广告的逻辑也会发生改变。一方面，新的用户交互模式会孕育新的商业机会，给自动出价的产品带来颠覆的改变；另一方面，新的技术理念和技术范式也会给自动出价算法带来革命性的升级。在如此汹涌磅礴技术浪潮到来之际，我们该如何革新广告营销的技术体系?

以上四个方向的技术演进如今如火如荼，希望未来有机会可以以进阶篇的形式分享给大家我们的思考与实践 ~

5. 参考文献

[1] Ranking and Tradeoffs in Sponsored Search Auctions
[2] Optimal reserve prices in weighted GSP auctions
[3] Revenue Optimization in the Generalized Second-Price Auction

[4] Learning Algorithms for Second-Price Auctions with Reserve
[5] Reserve Prices in Internet Advertising Auctions- A Field Experiment
[6] A Field Guide to Personalized Reserve Prices
[7] Position auctions
[8] Truthful Auctions for Pricing Search Keywords
[9] GSP with General Independent Click-Through-Rates
[10] Bidding to the Top- VCG and Equilibria of Position-Based Auctions
[11] Truthful Outcomes from Non-Truthful Position Auctions
[12] Revenue Monotone Mechanisms for Online Advertising
[13] Sponsored Search Auctions with Rich Ads
[14] Revenue Analysis of a Family of Ranking Rules for Keyword Auctions[15] Multi-Score Position Auctions
[16] Internet Advertising and the Generalized Second-Price Auction- Selling Billions of Dollars Worth of Keywords
[17] Simplified Mechanisms with Applications to Sponsored Search and Package Auctions
[18] General Auction Mechanism for Search Advertising
[19] Bayes - Nash equilibria of the generalized second-price auction
[20] Sponsored Search Auctions- Recent Advances and Future Directions
[21] On Revenue in the Generalized Second Price Auction
[22] Optimal Auction Design and Equilibrium Selection in Sponsored Search Auctions
[23] OPTIMAL AUCTION DESIGN
[24] Optimal Multi-Object Auctions
[25] Simple versus Optimal Mechanisms
[26] Auction Mechanism for Optimally Trading Off Revenue and Efficiency
[27] Efficiency-Revenue Trade-offs in Auctions
[28] Optimising Trade-offs Among Stakeholders in Ad Auctions
[29]《斯坦福算法博弈论二十讲》

加入我们：

阿里妈妈搜索广告算法团队，诚招日常实习生和暑期实习生，欢迎对广告算法感兴趣的同学（最好有 NLP、CV 和信息检索背景）投递简历。

邮箱： alimama_tech@service.alibaba.com

PerBid：在线广告个性化自动出价框架

作者：幸樵、律因、妙临

导读

出价产品智能化成为行业发展趋势，自动出价（Auto-bidding）已成为互联网广告主营销的主流，大商家数据量大往往在统一模型中更占优势，为提升中小商家效果，我们开启探索“千店千模”之路。

摘要

随着深度学习技术的发展，广告平台推出了多样化的自动出价服务，协助广告主实现智能决策。然而由于不同的广告主往往处于具有极强异质性的广告投放环境中，因此当前广泛采用的“使用统一自动出价策略服务全体广告主”的方法可能使得不同广告主的广告计划的投放效果存在较大的差异。在本文中，我们提出了个性化自动出价框架 PerBid（Personalized Automated Bidding Framework），将传统的仅有单一策略（agent）的自动出价策略拓展为多个能够感知上下文（context-aware）的策略（agent），其中每个策略对应一类特定的广告主聚类（Advertiser Cluster）。具体来说，我们首先设计了广告计划画像网络（Ad Campaign Profiling Network）用于建模动态的广告投放环境。之后，我们通过聚类技术将差异化的广告主分为多个类并为每一类广告主设计一个特定的具有上下文感知能力的自动出价策略（agent），从而实现为每个广告主匹配特定的个性化策略。

论 文： A Personalized Automated Bidding Framework for Fairness-aware Online Advertising

链接（点击↓阅读原文）： https://dl.acm.org/doi/pdf/10.1145/3580305.3599765

一、背景

随着电子商务的快速发展，在线广告成为了大量品牌和商户推广产品的主要渠道。为了更好地服务广告主，在线广告平台推出了各类广告策略服务，提供了基于机器学习的算法来辅助广告主实现广告投放过程中的智能决策。其中一个最具代表性的例子就是基于强化学习（Renforcement Learning，RL）的自动出价策略。然而由于每个

广告主自身数据的稀疏性以及平台计算资源的限制，当前的自动出价策略往往是以一种统一的方式生成的，也就是平台收集所有广告主广告计划的数据，利用这些数据训练生成一个统一（unified）的自动出价策略，然后再将这个策略应用到所有广告主的广告计划上。然而这样统一的自动出价策略对于每个单一个体广告主和广告计划来说并不是最优的。通过离线数据分析，我们发现了问题背后的两个潜在原因：一方面，不同广告主在不同时间的广告投放过程中面临差异化的广告投放环境，包括用户流量的分布情况、流量价格的差异等等。当前统一的自动出价策略未能深入的感知不同广告主所面临环境的差异性，因此导致不同广告主在使用同样的自动出价策略时所获取的广告效果具有较大的差异。另一方面，不同类型广告主和广告计划在线上平台所占据的比例有很大的差异，且其在利用基于 RL 的广告投放策略进行投放的过程中面对的广告投放状态（state）也具有很大的不同，导致了 RL 训练的过程中存在不平衡的状态探索（state exploration），使得策略在服务那些状态未能被完全探索的广告主时的效果不如那些状态被充分探索的广告主。因此在本工作中，我们研究如何设计个性化的基于 RL 的自动出价策略，使其能够充分观测差异化的环境，并能够高效地探索不同的广告投放状态，进一步提升所有广告主的效果。设计该自动出价策略主要面临两大挑战。首先，广告投放环境随着时间会发生快速的变化，同时决定环境的因素非常多样，包括投放时间、目标人群、其他广告主的策略等等，使得难以使用传统的建模方法对其进行建模，需要设计高效的方式来获取广告投放环境的上下文信息。其次，即使对广告投放环境进行了合理的建模，直接将其加入基于 RL 的广告投放策略当中会大幅度扩张广告投放状态的状态空间，从而进一步加剧数据的稀疏性和不平衡的状态探索，使得难以高效地生成能够感知上下文的自动出价策略。为了解决上述挑战，我们提出了个性化自动出价框架 PerBid。该框架不再仅生成单一的自动出价策略，而是通过生成一系列能够感知上下文的候选自动出价策略以应对不同类型的广告计划及差异化的广告投放环境，并通过匹配不同的广告主 / 广告计划和特定的个性化出价策略进一步提升个体广告主的投放效果并缓解效果差异现象。具体来说，PerBid 首先通过广告计划画像网络来表征和建模当前广告主 / 广告计划所处的投放环境，并基于对环境的建模和表征进一步设计具有上下文感知能力的自动出价策略。之后，我们将广告主 / 广告计划根据其所处广告投放环境的差异通过聚类技术划分入若干个聚类，并为每一个聚类训练生成一个特定的自动出价策略。最后，对于一个广告主 / 广告计划，我们根据历史信息为其匹配最合适的候选策略，并通过本地适配进一步提升其效果。大量的在离线实验验证了该方法的效果。

二、前置知识

2.1 自动出价策略

当前的在线广告投放过程可以被建模在线规划问题，具体如下：

$$\begin{aligned} \max_{\mathbf{x}} & \sum_{i=1}^{n} x_i \times v_i, \\ s.t \quad & \sum_{i=1}^{n} x_i \times cost_i \leq Budget, \\ & \frac{\sum_{i=1}^{n} x_i \times cost_i}{\sum_{i=1}^{n} x_i \times ctr_i} \leq PPC, \\ & x_i \in \{0,1\}, \forall i \in [1,n], \end{aligned}$$

其中 n 为用户流量的数量；$v_i, cost_i, ctr_i$ 分别为用户流量 i 的预估流量价值（例如预估转化）、流量价格以及预估点击率；x_i 代表是否获取该用户流量；$Budget, PPC$ 则是广告主预设的预算约束和 Pay-Per-Click 约束。我们的目标是通过对每一条用户流量设置合理的出价 bid_i 来确定对应的 x_i，从而在满足各项约束的前提下最大化广告主的目标。根据 [1] 中的结果，我们在 CPC 扣费规则下的 GSP 广告拍卖中可以通过设置如下形式的出价策略以获取最优结果：

$$bid_i^* = \left(\alpha^* \times \frac{\frac{v_i}{ctr_i}}{average(\frac{v_i}{ctr_i})} + \beta^* \right) \times PPC$$

其中 $\alpha^* = \frac{average(\frac{v_i}{ctr_i})}{(\lambda_1 + \lambda_2) \times PPC}$，$\beta^* = \frac{\lambda_2}{(\lambda_1 * + \lambda_2)}$ *为出价参数*，λ_1^*, λ_2^* 分别为预算约束和 PPC 约束对应的最优对偶变量。

由于最优对偶变量需要通过所有潜在用户流量的信息才能在离线环境下计算获得，因此当前平台设计了基于反馈控制 [1] 或是强化学习 [2, 3] 的自动出价策略，通过对出价参数进行在线实时调控从而优化广告投放效果。我们在此工作中聚焦于基于 RL 的自动出价策略，并将这一出价参数调控过程建模为一个马尔可夫决策过程（Markov Decision Process）。该马尔可夫过程可以被建模为一个四元组 $\mathcal{S}, \mathcal{A}, \mathcal{R}, \mathcal{T}$，其中状态 $s \in \mathcal{S}$ 表征实时的广告投放状态，主要涵盖了当前的实时广告投放情况（包括剩余投放时间、剩余预算、预算花费速度、实时 PPC、累计 PPC 以及当前的出价参数等信息），$a \in \mathcal{A}$ 代表对出价参数的实时调控。基于 RL 的自动出价 agent 在时段结束时

根据其自身的策略 π 以及实时获取的广告投放状态 s_t 采取参数调控动作 a_t 对出价参数进行调控（$\{\alpha_{t+1}, \beta_{t+1}\} = \{\alpha_t, \beta_t\} + a_t$），然后广告投放状态会根据环境动态（Environment Dynamic）$\mathcal{T}: (s_t, a_t) \rightarrow (s_{t+1}, r_t)$ 转移到 $s_{t+1} \in \mathcal{S}$ 同时获取 $r_t \in \mathcal{R}$。其中 r_t 代表了在 $t+1$ 时间段内利用调控后的出价参数 $\{\alpha_{t+1}, \beta_{t+1}\}$ 进行投放后所获取的流量价值。根据 [3] 中提出的方法，我们可以将在 s_t 状态下采取 a_t 的长期期望总收益定义为 $G(s_t, a_t) = \frac{R_t + R_{-t}}{R^*} - P$，其中 R_t 为直到 t 时段结束时已获取的流量价值，R_{-t} 是在 $t+1$ 时段直到投放结束固定出价参数为 $\{\alpha_{t+1}, \beta_{t+1}\}$ 所能获取的流量价值，R^* 为使用离线最优参数所能获取的流量价值，P 为违反 PPC 约束的惩罚项。在 RL agent 的训练过程中，我们将逐步优化策略 π，使其能够采取最优动作以最大化长期期望总收益，即 $\pi(s) = \arg\max_a G(s, a)$。

2.2 自动出价的效果评估

本文主要研究采用统一的自动出价策略服务大量差异化广告主时不同广告主 / 广告计划之间存在的广告投放效果的效果差异现象。通过在线上收集不同广告计划的投放数据，我们首先构建了对应的离线数据集。在这一离线数据集中，我们按照传统方法 [3] 训练生成了统一的 RL 自动出价策略，并将该策略应用到离线数据集中的测试广告计划上。通过数据分析，我们可以发现该策略在不同的广告主 / 广告计划上的表现具有很大差异，如图 1 所示，针对不同广告计划的广告投放效果分布具有非常明显的长尾效应：后 10% 的广告计划相较于前 10% 的广告计划仅能获得 76% 的广告投放收益。

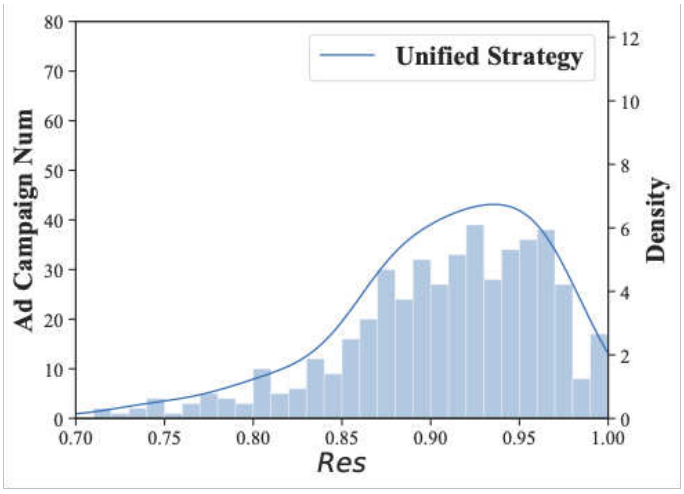


图 1　统一 RL 自动出价策略的广告投放效果分布

同时，我们可以发现不同类型的广告计划由于其在平台中占据的比例不同，其效果也有存在很大的差异。当前的平台中我们可以根据离线最优 β^* 的值将所有的广告计划划分为三类，包括预算敏感型（$\beta^* = 0$）、PPC 敏感型（$\beta^* = 1$）、以及综合型（$\beta^* \in (0,1)$）。其中预算敏感型的广告计划在离线数据集和实际的线上平台中占据最大的比例，而另两类的规模则相对较小。对应的，统一生成的 RL 自动出价策略在应对这三类广告计划时效果也存在很大差异，如表 1 第一行结果所示，在离线数据集中，综合型广告计划的平均广告投放效果相较于预算敏感型广告计划下降了 4.82%，同时 PPC 敏感型广告计划的 PPC 约束超限率相较于预算敏感型提升了 12.27%。在线上实际应用的过程中这一差异则被进一步放大，如图 2 所示，预算敏感型广告主的平均广告投放效果和 PPC 约束的达成情况相较于其他两类都有非常明显的优势。

Types of Ad Campaigns	Budget-sensitive (59.61%)	PPC-sensitive (20.20%)	Mixture (20.19%)
Baseline	0.9199	0.9025	0.8756
Consistent	0.9186	0.9443	0.8996
Inconsistent	0.5376	0.7813	0.8599

表 1　不同类型广告计划的广告投放效果在不同策略训练设置下的比较

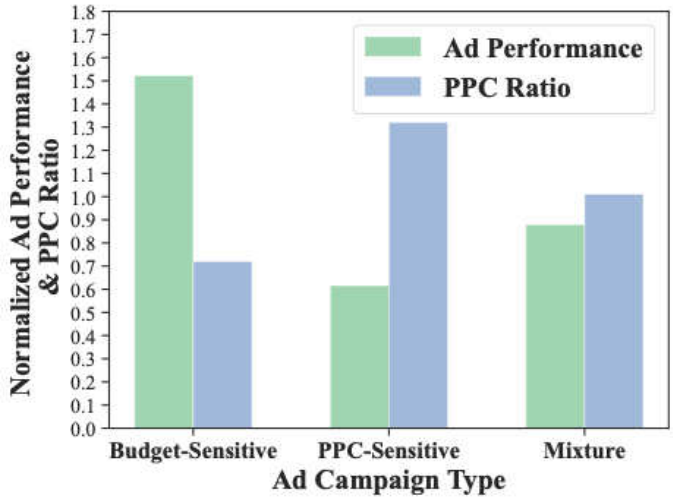


图 2　不同类型广告计划线上广告投放效果和 PPC 控制情况

通过对广告计划投放数据和 RL 自动出价策略生成过程的分析，我们发现了两个造成这个问题的主要原因：一方面，不同广告计划所处的广告投放环境，尤其是用户流量

的规模和质量的分布情况，具有极强的异质性。不同广告计划的用户流量平均质量之间的差异可以达到 15 倍，而用户流量规模的差异更是可以达到 28 倍，使得自动出价 agent 难以生成一个对所有广告计划都能取得优秀效果的统一策略。另一方面，在自动出价策略生成的过程中我们随机从所有历史广告计划中采集数据进行策略训练，而不同类型的广告计划的占比以及其在广告投放过程中所经历的广告投放状态具有非常明显的差异。如图 3 所示，预算敏感型的广告主在广告投放过程中经历的状态更注重预算的平稳花费（图 3 蓝色部分），而 PPC 敏感型的广告主则更注重 PPC 的控制（图 3 绿色部分），因此采用当前的统一策略生成方式会导致不平衡的状态探索和策略训练，导致不同类型广告主间效果的差异。

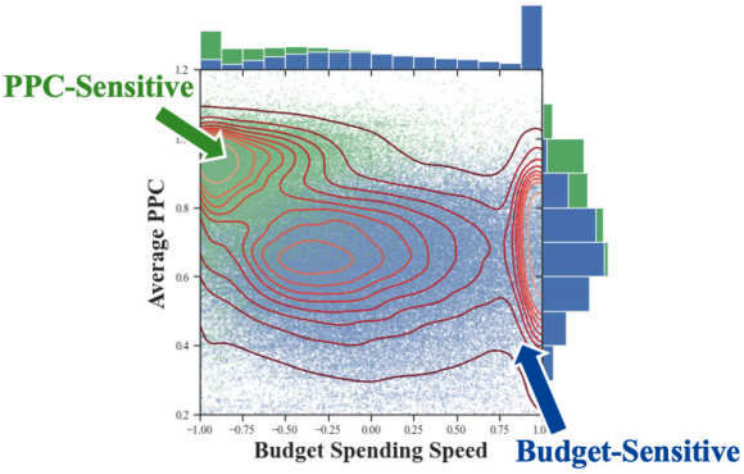


图 3　不同类型广告计划的广告效果在不同策略训练设置下的比较

我们在表 1 的第二行和第三行进一步通过离线实验探讨了这两个要素对于广告投放效果的影响。在第二行中我们展示了如果使用同一类型的广告计划作为训练数据，并将其应用到相同类型的广告计划时的效果；在第三行中我们展示了如果训练时使用的广告计划和测试时的广告计划不同时的结果。从中我们可以发现当训练时的广告投放环境和测试时的环境接近，且训练过程中所关注的广告投放状态和实际遇到的广告投放状态较为一致时，生成的投放策略的效果会有较大提升；当训练时的广告投放环境与实际环境有较大差异，且对广告投放状态的探索与实际遭遇的状态有较大偏差时，该策略的广告投放效果会产生断崖式的下降。因此基于以上的观察，本工作的目标就是针对这两个因素设计个性化自动出价框架，为每个广告主 / 计划提供个性化的具有感知环境上下文能力的自动出价策略从而解决广告投放过程中的效果差异问题。

三、个性化自动出价框架

在本工作中，我们设计了个性化自动出价框架 PerBid，通过设计一系列具有环境上下文感知能力的自动出价策略构成候选策略集并为每一个广告计划匹配最合适的策略从而在保障整体策略效果的同时缓解效果差异问题。图 4 展示了 PerBid 的主要流程，该框架首先利用一个广告计划画像网络来表征实时广告投放环境并生成广告计划画像。根据该画像我们能够进一步为特定的环境设计能够感知上下文的自动出价策略。由于单一策略无法高效应对所有的差异化环境，因此我们根据不同广告计划所处的环境对广告计划进行聚类，将处于相似环境的广告计划分入同一聚类中，并为每一个聚类生成一个特定的策略从而生成候选策略集。对于一个新到达的广告计划，我们根据其历史数据为其匹配最合适的候选策略，并通过策略本地适配进一步提升策略能力。

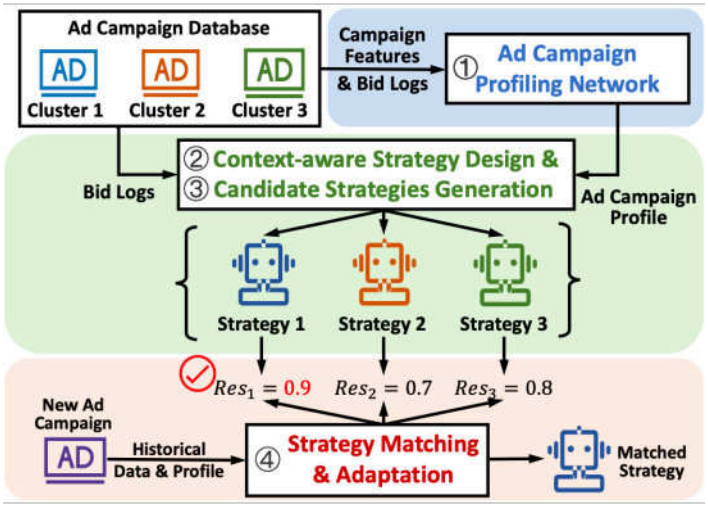


图 4　PerBid 主要流程

3.1 广告计划画像（Ad Campaign Profiling）

我们通过广告计划画像网络来表征动态的广告投放环境。该画像网络的设计如图 5 所示。在广告计划画像的过程中我们同时考虑了计划层面的静态特征和竞价层面的动态特征，并利用一个广告投放效果分类任务来进一步训练整个广告计划画像网络。具体来说，计划层面特征从宏观的角度描述了当前广告计划所处广告投放环境，包括了预先设置的约束的情况以及其他 ID 类特征（例如计划 ID、目标人群 ID、广告投放时段、广告投放渠道等）。我们利用 Data Embedding 技术提取这些这些高层次静态特

征，从而帮助画像网络粗略但是快速得分辨环境的特性。对于竞价层面的动态特征，我们首先针对每个时间段积累的 bid log 进行 Feature Encoding 从而为每个时间段生成特征向量 E_t，其中包含了该时间段内的用户流量规模、用户流量价格以及用户流量质量（性价比）的分布的相关信息。为了表征广告投放环境随时间的变化规律，我们进一步利用 GRU 循环神经网络来提取不同时间段之间的时序演进关系。在 T 时间段结束时，其输入为 $1-T$ 时间段的特征向量 $\{E_1,\ldots,E_T\}$，而输出则为隐藏层参数 $h(T)$。通过结合计划层面特征和竞价层面特征，我们可以生成最终的广告计划画像以表征其所处的广告投放环境。

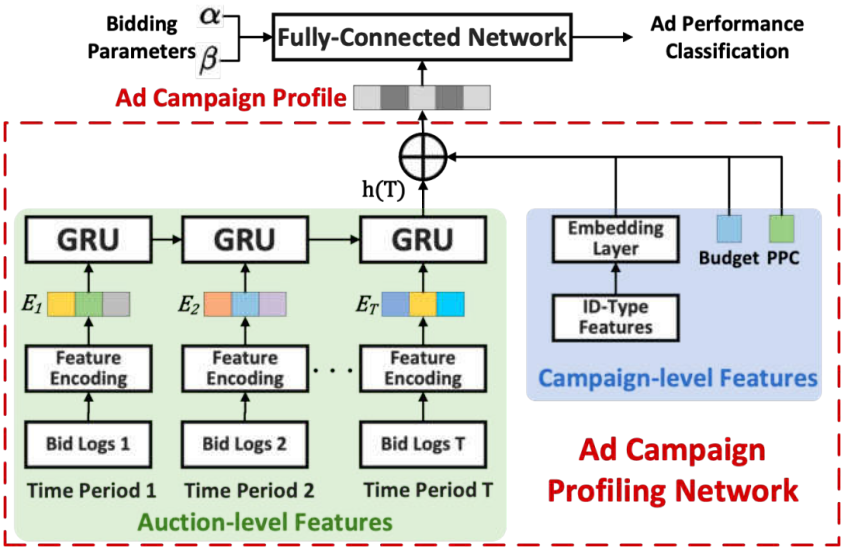


图 5　广告计划画像网络

3.2 上下文感知出价策略（Context-aware Bidding Strategy）

基于生成的广告计划画像，我们可以进一步设计可感知环境上下文的自动出价策略。在 [3] 中提出的 USCB 的基础上，我们保留了动作 a 和长期期望收益 G 的设计，并将先前对广告投放环境的表征融入广告投放状态 s 的设计中，从而希望出价策略在保障泛化性能的基础上高效观测实时投放环境的信息。为了实现对环境的感知，我们首先将广告计划画像中竞价层面的特征 $h(t)$ 作为环境上下文特征直接加入广告投放状态 s_t。考虑到广告计划的规模非常庞大，直接将高层次且较为稀疏的计划层面特征加入状态 s_t 会大幅度提升状态空间的大小，从而加剧数据稀疏性和对不同状态的不平衡的探索和训练。因此我们并不直接将计划层面特征加入状态，而是将其用于改进对

实时投放情况的表征，希望使策略能够在理解环境上下文信息的基础上实现更准确的实时投放情况表达。具体来说，s_t 可以被表达为 $s_t = \{status_t, context_t\}$，其中 $context_t = h(t)$ 代表从广告计划画像的竞价层面特征中直接获取的环境上下文信息。$status_t = \{remain_time_t, remain_budget_t, spend_speed_t, average_ppc_t, ppc_t, \alpha_t, \beta_t\}$ 则代表改进后的实时投放情况表征，其具体定义如下：

$$
\begin{aligned}
&remain_time_t = \frac{T-t}{T},\ ppc_t = \frac{Cost_t}{Click_t \times PPC},\\
&remain_budget_t = \frac{Budget - \sum_{i=1}^{t} Cost_i}{Budget},\\
&average_ppc_t = \frac{\sum_{i=1}^{t} Cost_i}{\sum_{i=1}^{t} Click_i \times PPC},\\
&spend_speed_t = \min\left(\frac{\frac{\sum_{i=t+1}^{T} w_i}{w_t} \times \frac{Cost_t}{Budget}}{remain_budget_t} - 1, 1\right).
\end{aligned}
$$

其中 $Cost_t$ 和 $Click_t$ 分别是 t 时间段内所花费的预算和获得的广告点击；预设的约束值 $Budget$，PPC，以及广告计划的持续时间 T 被用于归一化 $remain_time$ 和 $remain_budget$，并将 ppc 和 $average_ppc$ 映射到一个相对值。我们进一步利用不同广告计划有关用户流量规模在不同时段分布的知识（即个性化的权重向量 $\mathbf{w}$）来修正 $spend_speed$，使其能够感知用户流量规模随着时间的演变。

3.3 候选策略生成器（Candidate Strategies Generation）

由于单一策略无法高效应对所有的环境，因此我们进一步生成一系列不同的候选策略以应对差异化的环境。我们通过聚类的手段将所有的广告计划分入数个聚类中去，每个聚类代表一类处在类似环境中的广告计划，然后为每个聚类生成一个特定的自动出价策略（agent）以应对不同的广告投放环境。这个候选策略集的生成过程可以被划分为三个步骤，包括聚类初始化、自动出价 agent 训练以及广告计划的重分配。具体的算法流程如算法 1 所示，我们首先根据预设的规则（例如广告主预设 PPC 的值等）初始化所有广告计划对应的聚类（行 1–3）；随后，在训练步骤中我们对每一个聚类进行并行的出价 agent 训练。在每一个训练轮中，我们在该聚类对应的广告计划中随机选择一个并对其广告投放状态进行探索（行 7–12），并将对应的观测数据存储入对应聚类的经验回放池中（行 13）。在每一个训练轮探索结束后，我们从经验回放池中采样一系列历史经验，并利用基于 actor-critic 的强化学习训练算法对出价 agent 进行

更新（行 14–15）。由于不同的聚类拥有相同的训练和探索资源，而每个聚类所覆盖的广告计划的规模不同，因此覆盖规模较小的聚类能够针对罕见广告投放环境下的广告投放状态进行集中的探索和训练，避免了策略生成过程中的不平衡。因为预设的聚类初始化方法往往只能获取次优的广告计划分类，因此在训练步骤结束后，我们进一步根据不同广告计划在不同聚类对应出价 agent 下的表现进行广告计划的重分配，将广告计划动态得重新划分到最合适（投放效果最好）的聚类中去。通过重复训练步骤和广告计划重分配步骤，我们可以逐渐生成若干个对应不同类型广告投放环境的稳定的广告计划聚类，并生成对应的高质量候选投放策略（agent）。

Algorithm 1: Candidate Strategies Generation.

Input: Ad Campaigns $\mathbb{C} = \{c_1, ..., c_N\}$; Automated Bidding Agents $\mathbb{A} = \{agent_1, ..., agent_M\}$; Replay Buffers $\mathbb{B} = \{B_1, B_2, ..., B_M\}$; Clusters $\mathbb{S} = \{S_1, ..., S_M\}$;

1 Initialize $agent_j$ with policy network π_j and value network Q_j for each cluster S_j; `// Initialization step.`

2 **for** *Campaign* $c_i \in \mathbb{C}$ **do**

3 $\quad j \leftarrow Init_Cluster(c_i)$; $S_j \leftarrow S_j \cup c_i$;

4 **while** *Not Converge* **do**

5 $\quad$ **for** *Cluster* $S_j \in \mathbb{S}$ **do**

6 $\quad\quad$ **for** *ep* **from** 1 **to** *Episode* **do** `// Training Step.`

7 $\quad\quad\quad$ Randomly select the campaign $c_i \in S_j$;

8 $\quad\quad\quad$ Initialize bidding parameters $\{\alpha_1, \beta_1\}$;

9 $\quad\quad\quad$ **for** t **from** 1 **to** $T_i - 1$ **do**

10 $\quad\quad\quad\quad$ Obtain state s_t with parameters $\{\alpha_t, \beta_t\}$;

11 $\quad\quad\quad\quad$ Explore state s_t by taking $a_t \leftarrow \pi_j(s_t) + \epsilon$, $\{\alpha_{t+1}, \beta_{t+1}\} \leftarrow \{\alpha_t, \beta_t\} + a_t$;

12 $\quad\quad\quad\quad$ Observe long-term value $G(s_t, a_t)$;

13 $\quad\quad\quad\quad$ Store observation $O(s_t, a_t, G(s_t, a_t))$ to B_j;

14 $\quad\quad\quad$ Sample observations **O** from B_j;

15 $\quad\quad\quad$ Update π_j and Q_j with **O**;

16 $\quad\quad$ **for** *Campaign* $c_i \in S_j$ **do** `// Re-assignment step.`

17 $\quad\quad\quad$ **for** $agent_k \in \mathbb{A}$ **do**

18 $\quad\quad\quad\quad$ $Res_{ik} \leftarrow Auction_Simulation(c_i, agent_k)$;

19 $\quad\quad\quad$ $k^* = \text{argmax}_k\, Res_{ik}$;

20 $\quad\quad\quad$ **if** $Res_{ik^*} - Res_{ij} > thr$ **then**

21 $\quad\quad\quad\quad$ $S_{k^*} \leftarrow S_{k^*} \cup c_i$; $S_j \leftarrow S_j \setminus c_i$;

22 **Return** the set of candidate strategies $\mathbb{A}$;

3.4 策略匹配与自适应（Strategy Matching and Adaptation）

在完成了候选策略生成后，我们利用广告计划的历史数据进行计划和策略之间的匹配以及进一步的本地化适配。在策略匹配的过程中，对于一个待匹配策略广告计划，我们首先采集其之前 D 天的历史数据，并利用 M 个候选策略在这些历史数据上进行离线模拟竞价，从而获得对第 i 天的结果 $\mathbf{Res}^i = (Res_{i1}, \cdots, Res_{iM})$。基于前 D 天的结果，我们通过加权平均可以获得一个历史效果向量 $\mathbf{Res}^{avg} = (Res_1^{avg}, \cdots, Res_M^{avg})$ 代表每个候选策略在前 D 天历史数据上的平均表现。我们选择平均表现最好的候选策略作为匹配的策略。对于缺乏历史数据的冷启动广告计划，我们为其匹配覆盖面最广且在所有广告计划上平均效果最好的出价策略作为默认策略。策略本地化适配和策略生成过程中的训练步骤类似，只是所有的训练数据都来自于同一广告计划的历史数据。由于策略的本地化适配过程同时也可能造成过拟合的问题，因此仅应用于初始效果不佳的广告计划上。

四、实验结果

4.1 离线实验

离线实验数据集采集自阿里巴巴展示广告平台，共包含超过 3500 个广告计划并被划分为训练集、验证集、测试集。我们用如下的效果指标表征自动出价策略在单一广告计划上的投放效果：

$$Res = \min\left(\frac{\sum_{i=0}^{T-1} r_i}{R^*} \times \frac{1}{\max(PPC_ratio, 1)}, 1\right)$$

其中前半部分为归一化后的广告投放结果，而后半部分则为违反 PPC 约束的惩罚项。为了全面地展示 PerBid 在保障总体效果和公平性上的表现，我们同时展示了整体平均效果 $\overline{Res}$ 以及 $\overline{Res}_{0.3}$ 和 $\overline{Res}_{0.1}$ 用于代表后 30% 和 10% 广告计划的平均效果。此外，我们还考虑了两类常用的公平性指标 GGF（Generalized Gini Social Welfare Function）[4] 和 $Gini$（Gini Coefficient）[5]。其中 GGF 的定义如下：

$$GGF_{\mathbf{g}}(\mathbf{Res}^{\uparrow}) = \sum_{i=1}^{N} g_i \times Res_i^{\uparrow}, \quad g_i = \frac{1 - i/N}{\sum_{k=1}^{N}(1 - k/N)},$$

其中 $\mathbf{Res}^{\uparrow}$ 代表将所有广告计划的结果从小往大排列后组成的向量。通过赋予效果差的广告计划更大的权重 g，该指标可以同时考虑公平性和整体效果。$Gini$ 的定义如下：

$$Gini = \frac{\sum_{i=1}^{N}\sum_{j=1}^{N}|Res_i - Res_j|}{2 \times N^2 \times \overline{Res}}.$$

直接表征了不同广告计划投放效果的离散程度，$Gini$ 越大则说明不公平现象越明显。

我们在离线实验中考虑如下几个对照方法：1）M-PID [1]：使用多变量联合反馈控制进行参数调控。2）DRLB [2]：基于强化学习的方法，定义特定状态下采取特定参数调控动作的奖励值为所有出现该状态动作的轨迹的总体广告投放效果的最大值，通过设计独立的 RewardNet 进行奖励值的预估。3）Baseline [3]，当前线上应用的基于 actor-critic 的强化学习算法 USCB。4）Baseline w. Profile：如前文 4.2 描述在基线 USCB 算法的基础上改进得到的可感知环境上下文的自动出价策略。5）Fixed Agents (OPT): 在 Baseline w. Profile 的基础上将单一策略拓展为三个候选策略，其广告计划聚类仅根据预设规则进行划分而不进行广告计划的动态重分配。在策略匹配的过程中视为能够实现 100% 准确率的最优匹配。6) PerBid (OPT)：本文提出的个性化自动出价框架，如前文 4.3 描述生成三个候选策略对应，在策略匹配的过程中视为能够实现 100% 准确率的最优匹配。7) PerBid (Match) ：本文提出的个性化自动出价框架，生成三个候选策略且在策略匹配的过程中使用 4.4 提出的基于历史数据的策略匹配算法（匹配过程仅使用前一天的历史数据）。具体的离线实验结果如下表所示，从中我们可以发现 PerBid 在能够保障策略匹配准确率的情况下相较于其他算法在综合效果和保障公平性方面具有非常明显的效果提升。同时，在使用有限历史数据的情况下 PerBid 也可以达成 74.53% 的匹配准确度并取得不俗的效果。在图 6 中我们进一步展示了各方法的效果分布，可以发现 PerBid 能有效缓解长尾效应提升公平性。在图 7 中我们展示了使用 PerBid 后在不同类型广告计划上获得的平均效果。

Method	$\overline{Res}$	GGF	$Gini$	$\overline{Res}_{0.3}$	$\overline{Res}_{0.1}$
M-PID	0.8918	0.8480	0.0490	0.7954	0.7201
DRLB	0.8941	0.8557	0.0429	0.8075	0.7342
Baseline	0.9033	0.8678	0.0392	0.8242	0.7517
Baseline w. Profile	0.9233	0.8943	0.0313	0.8565	0.7969
Fixed Agents (OPT)	0.9327	0.9049	0.0298	0.8678	0.8095
PerBid (OPT)	**0.9494**	**0.9270**	**0.0235**	**0.8982**	**0.8493**
PerBid (Match)	0.9337	0.9039	0.0318	0.8640	0.8026

表 2　不同方法的广告投放效果和公平性指标

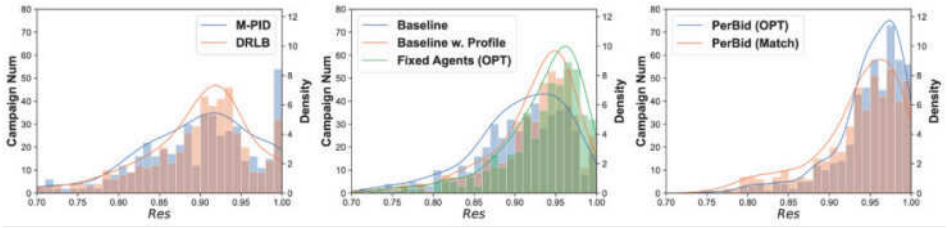


图 6　不同方法的广告投放效果分布

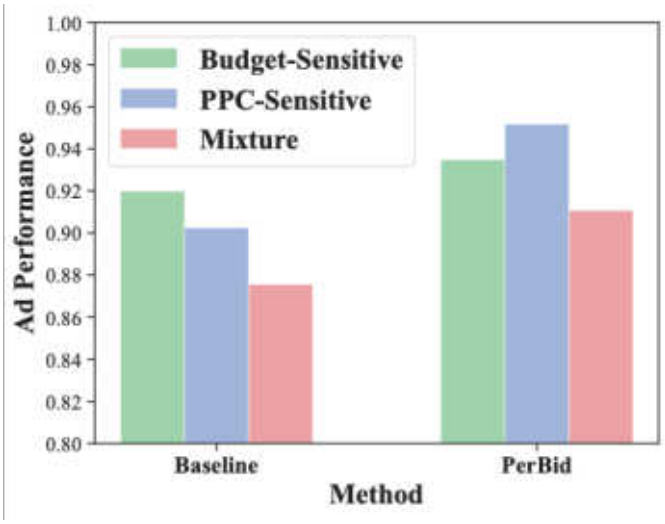


图 7　PerBid 在不同类别广告计划上的效果

4.2 在线实验

我们将提出的框架部署在阿里巴巴展示广告平台中，并将其与线上使用的基础 USCB 算法进行对比。并在一定时间内选取了 1% 的全体广告计划进行了在线 A/B 实验，具体的实验设置如下：1) 为了保障测试公平性，我们对所有的广告计划使用相同的固定权重向量 $\mathbf{w}$ 进行实时投放状态修正，同时省略了环境上下文特征。2) 我们根据广告计划的最优出价参数 β^* 进行聚类划分并生成了三个候选自动出价策略，同时我们单独利用所有的类型的广告计划生成了一个默认策略以应对冷启动广告计划。3) 线上实验中我们使用前七天的历史数据进行策略匹配来选择最优的候选策略。从线上实验的结果中我们可以发现 PerBid 可以提升 $\overline{Res}$ 达 8.02%，提升 GGF 达 8.53%，提升 $\overline{Res}_{0.3}$ 达 10.85%，展现了 PerBid 在提升综合效果和保障公平性两方面的能力。

五、结论

在本文中，我们揭示了使用统一自动出价策略服务全体广告主的效果差异问题，并分析了其主要原因。为了解决广告主之间的效果差异问题，我们提出了一个个性化的自动出价框架 PerBid。在这个框架中，我们首先提出了一个广告计划画像网络来建模广告环境，基于此设计了上下文感知的自动出价策略。然后，我们根据它们的特征将广告计划分组为几个簇，并为每个簇分配特定的策略。最后，我们提出了一个匹配算法来匹配异质的广告计划与最合适的策略。我们在真实世界的数据集上进行了全面的离线实验和在线 A/B 测试，以验证该框架在提高平均性能和解决效果差异问题方面的有效性。

参考文献

[1] Xun Yang, Yasong Li, Hao Wang, Di Wu, Qing Tan, Jian Xu, and Kun Gai. 2019. Bid optimization by multivariable control in display advertising. In Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. ACM, 1966 – 1974.

[2] Di Wu, Xiujun Chen, Xun Yang, Hao Wang, Qing Tan, Xiaoxun Zhang, Jian Xu, and Kun Gai. 2018. Budget constrained bidding by model-free reinforcement learning in display advertising. In Proceedings of the 27th ACM International Conference on Information and Knowledge Management. ACM, 1443 – 1451.

[3] Yue He, Xiujun Chen, Di Wu, Junwei Pan, Qing Tan, Chuan Yu, Jian Xu, and Xiaoqiang Zhu. 2021. A Unified Solution to Constrained Bidding in Online Display Advertising. In Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining. ACM, 2993 – 3001.

[4] John A Weymark. 1981. Generalized Gini inequality indices. Mathematical Social Sciences 1, 4 (1981), 409 – 430.

[5] Sen, Amartya. 1997. On economic inequality. Oxford university press.

Auction Design in the Auto-bidding World 系列一：面向异质目标函数广告主的拍卖机制设计

作者：少平、妙临

导读

传统拍卖机制不存在了！出价产品智能化成为行业发展趋势，自动出价（Auto-bidding）已成为互联网广告主营销的主流，经典效用最大化模型（Utility Maximizer）的假设已经不再能良好地刻画此类广告主，传统拍卖机制在 Auto-bidding 下的激励性质和最优性需要被重新思考。

摘要

数字广告是互联网平台的主要收入来源之一。近年来，很多广告主开始采用自动竞价（Auto-bidding）工具来优化广告效果，这使得拍卖理论中经典的效用最大化模型（Utility Maximizer）不再能够良好地刻画此类广告主。因此，近年来业界研究提出了一个新的模型，称为价值最大化模型（Value Maximizer），用于有投资回报率（ROI）约束的自动竞价广告主。然而，无论是效用最大化还是价值最大化的模型，都只能刻画实际广告平台中的一部分广告主。在效用最大化广告主和价值最大化广告主并存的混合环境中，真实的广告拍卖设计属于多参数机制设计问题，因为广告主可以同时操作他们的广告点击价值和所属的类别两个信息。**本文提出了一种满足真实性（Truthfulness）的拍卖机制，其通过巧妙结合经典的 VCG 和 GSP 机制中的扣费规则来解决多参数机制设计中的复杂性问题。**同时该机制具备良好的机制性质，其社会福利的近似比为 2，我们同时也证明了该近似比的下界为 1.25。特别值得一提的是，我们所设计的拍卖机制是针对效用最大化广告主的 VCG 和针对价值最大化广告主的 GSP 两种拍卖机制的自然推广。基于该项工作整理的论文已被 AAAI 2023 录取，欢迎阅读交流。

论 文：Utility Maximizer or Value Maximizer: Mechanism Design for Mixed Bidders in Online Advertising

下 载：https://arxiv.org/abs/2211.16251

一、背景介绍

在典型的数字广告场景中，平台通过 Vickrey-Clark-Grove（VCG）拍卖机制或广义第二价格（Generalized Second Price，GSP）拍卖机制进行广告位的分配并为广告主计算相应的扣费。在近二十多年的研究中，学术界和工业界已经发现，VCG 是一种能够满足真实性（Truthfulness，也称诚实性或激励兼容性）的机制，而 GSP 虽然是工业界更普遍的选择，却并不满足真实性，不过 GSP 拍卖中存在无嫉妒纳什均衡，而且所有无嫉妒纳什均衡的收益都不比 VCG 低 [1,2]。

现有的关于 VCG 和 GSP 的分析主要建立在拟线性效用模型上，也被称为效用最大化广告主（Utility Maximizer，UM），即广告主的目标是优化其分配的价值和扣费之间的差值。然而，在当前的很多在线广告系统中，广告主已经开始使用自动竞价（Auto-bidding）工具，利用自动竞价工具，广告主只需要设置高层次的约束条件，即在一定的预算下，指定一个投资回报率（Return On Investment，ROI，所分配得到的价值和扣费之间的比例）的阈值约束。当投资回报率约束接近于 1 时，它可以由 UM 模型中的个体理性的性质来刻画。然而，当投资回报率约束变得较大时，经典的 UM 模型无法刻画出广告主在自动竞价中的行为。因此，雅虎公司的研究人员 Wilkens、Cavallo 和 Niazadeh[3] 为广告主提出了另一个模型，称为价值最大化广告主（Value Maximizer，VM），该模型将分配的价值作为广告主的首要目标，将扣费作为其次的目标，只有当价值相同时才偏好扣费更少的结果。他们通过理论分析和实际系统中的观察发现，只要广告主的投资回报率约束稍高（高于 2 或 3），其行为模式就会接近 VM 模型对应的行为模式。这个新模型带来了一个重要的理论结果，即工业界常用的 GSP 拍卖成为一个满足真实性的机制，这为 GSP 在工业界的普遍使用提供了一个新的视角，也使得 VM 模型吸引了学术界和工业界的广泛关注。

然而，无论是 UM 还是 VM 模型，都只能代表一部分广告主，其中，UM 代表投资回报率约束相对较低的广告主，而 VM 则代表投资回报率约束相对较高的广告主。由于广告主的投资回报率约束可能多种多样，那么当 UM 和 VM 这两类广告主同时存在于一个广告系统中时，如何设计一个满足真实性的机制呢？这个问题包括两个层次：1）类别信息是公开的，即一个广告主只能自主上报他的广告点击价值；2）类别信息是私有的，即一个广告主可以同时自主上报他的类别和他的广告点击价值。即使是较简单的第一种情况，也不是一个简单的问题，因为在这种情况下，广告主的策略性行为是未知的，也就是说 VCG 或 GSP 机制都不会满足真实性。而后一种情况更加符合实际场景，因为广告主可以很容易地操纵自己的投资回报率约束，以改变相应的类

别，进而从中获益。这种私有的类别信息给广告拍卖机制设计带来了实质性的挑战，这一问题属于多参数机制设计领域，而这一领域通常面临较难解决的开放性问题。

在为了解决上述问题，我们要为公开类别信息和私有类别信息两种情况设计满足真实性的机制。针对公开类别信息的情况，存在以下满足真实性的机制，且能够保证最佳的社会福利：根据广告主对广告位的出价进行排序并分配坑位，然后按照 VCG 扣费规则向 UM 收费，按照 GSP 扣费规则向 VM 收费。该机制是直观的，但在某种程度上对 VM 是“不公平的”，因为 VM 与 UM 相比，获得相同的分配的前提下，在该机制中可能需要付出更多费用。这意味着该机制在类别信息是私有信息的情况下是不满足真实性的。因此，我们进一步提出了一个针对私有类别信息的混合广告主的机制（Mechanism for bidders with PRivate classes，MPR）。MPR 的关键思想是为每个坑位而不是每个广告主指定这样一个扣费规则：从下方最近的 UM 推导出的 VCG 式扣费值和从下方最近的 VM 推导出的 GSP 式扣费值中取最大值。有了这个扣费规则，MPR 先通过自下而上的方式用所有的 VM 填充广告位，然后迭代地每次给一个 UM 分配一个具有最优效用值的广告位。大量实验证明，这种机制在广告点击价值和类别信息两个维度上都是满足真实性的，同时也保证了社会福利的近似比最高为 2。我们还证明了，没有任何机制可以达到比 1.25 更好的近似比。

本文提出的 **MPR 机制**，主要成果包括两方面：一方面，我们首次研究了 UM 和 VM 并存的混合环境下满足真实性的广告拍卖机制设计，这为 VCG 和 GSP 机制的结合提供了一个有趣的视角：当所有的广告主都是 UM 时，我们提出的两个机制都会简化为 VCG；当所有广告主都是 VM 时，会简化为 GSP。另一方面，针对投资回报率约束的广告主的机制设计是近年来的一个热门研究问题 [4,5]，我们的工作为理解该问题迈出了重要的一步。此外，我们的机制发现，一个满足真实性的机制可能会将更高的坑位分配给具有高投资回报率约束的广告主（即本工作中的 VM），尽管他们的出价可能低于那些具有低投资回报率约束的广告主。

二、问题建模

在经典的广告拍卖模型中，有 K 个广告位，自下而上地索引为 $k \in 1, 2, \ldots, K$。每个广告位 k 具有点击率（Click-Through Rate，CTR）x_k，而且我们假设 $x_K \geq x_{K-1} \geq \ldots \geq x_1 > 0$。我们使用 $k = 0$ 和 $x_0 = 0$ 来表示最低坑位下面的一个虚拟坑位。有一组广告主 $\mathcal{N} = 1, 2, \ldots, n$，索引为 i，其中每个广告主都有一个私有的广告点击价值 v_i。不失一般性地，我们假设对于每一对广告主 i 和 j，有

$n > K$ 和 $v_i \neq v_j$。为了便于表述，我们假设一个拍卖机制 M 确定一个分配结果 $\Pi = \pi_1, \pi_2, \ldots, \pi_K$，并对每个广告主 i 的每次点击收取价格 p_i，其中 π_k 表示被分配到坑位 k 的广告主。我们也用 a_i 表示分配给广告主 i 的广告坑位的索引，即若有 $\pi_k = i$，则有 $a_i = k$。

在我们考虑的场景中，每个广告主可以是一个效用最大化广告主（UM），也可以是价值最大化广告主（VM）。它们的定义如下：

定义 1（效用最大化广告主，UM）：一个效用最大化的广告主的目标是最大化他的效用函数 $u_i = v_i x_{a_i} - p_i x_{a_i}$。

定义 2（价值最大化广告主，VM）：一个价值最大化广告主 i 的策略是最大化他的分配价值 $u_i = v_i x_{a_i}$，同时保证扣费 $p_i \leq v_i$，在价值相同的结果中，更偏好价格较低的结果。

直观地说，UM 遵循在线广告中经典的广告主模型，而 VM 则认为他所获得的分配价值优先于他的扣费，为了简化表达，我们也用“效用”一词表达 VM 的分配价值 u_i。我们用 $\tau_i \in UM, VM$ 来表示广告主 i 的类别，并用 $\theta_i = (v_i, \tau_i)$ 表示他的类型（我们将价值和类别统称为“类型”）。我们假设一个广告主可以虚报他的价值和他的类别，即，一个广告主可以报告他的类型为 $\hat{\theta}_i =< \hat{v}_i, \hat{\tau}_i >$，其中可能出现 $\hat{v}_i \neq v_i$ 或 $\hat{\tau}_i \neq \tau_i$。此外，我们用 θ 表示所有参与者的类型概况，而 θ_{-i} 表示除了 i 以外所有参与者的类型。基于这些符号，我们定义 $p_i\left(\hat{\theta}_i|\theta_i, \theta_{-i}\right)$ 作为广告主 i 的扣费，其中广告主 i 报告他的类型为 $\hat{\theta}_i$，他的真实类型为 θ_i，其他人的类型为 θ_{-i}，此外，我们用 $u_i\left(\hat{\theta}_i|\theta_i, \theta_{-i}\right)$ 表示相应的效用。

一个理想的拍卖机制应该满足三个要求：激励相容性（Incentive Compatibility，IC）、**个体理性**（Individual Rationality，IR）和**稳健性**（Robustness）。

定义 3（激励相容性，IC）：一个机制是激励相容的，当且仅当

$$u_i(\theta_i|\theta_i, \theta_{-i}) \geq u_i\left(\hat{\theta}_i|\theta_i, \theta_{-i}\right), \quad \forall \hat{\theta}_i \neq \theta_i, \theta_{-i}, i \in \mathcal{N}$$

。

定义 4（个体理性，IR）：一个机制是个体理性的，当且仅当

$$p_i\left(\hat{\theta}_i|\hat{\theta}_i, \theta_{-i}\right) \leq \hat{v}_i, \quad \forall \theta_{-i}, i \in \mathcal{N}$$

。

定义 5（稳健性，Robustness）：如果当所有广告主都是 UM 时，结果与 VCG 相同，而当所有广告主都是 VM 时，结果与 GSP 相同，则该机制是稳健的。

换句话说，IC 保证了机制中的所有广告主都没有虚报类型的动机，IR 保证了广告主在诚实出价时不会得到负的效用值，而稳健性保证了机制是现有机制的自然扩展。我们将宽泛地使用“真实性（Truthfulness）”一词来描述一个既满足 IC 又满足 IR 性质的机制。众所周知，VCG 对于 UM 是满足真实性的。近年来，也有工作证明了 GSP 对 VM 是满足真实性的。这些研究都集中在只有一类广告主且类别信息是公开的情况下。而在本工作中，我们的目标是为类别信息私有的混合广告主设计一个满足真实性的机制，同时使拍卖的整体福利最大化。在这里，混合广告主可以是 UM 也可以是 VM。

值得注意的是，经典的社会福利概念，即所有广告主的效用和卖方的收入之和，对于 VM 来说并不是一个可行的定义，因为在社会福利的计算中，广告主和卖方之间的交易金额不能恰好抵消（这一点与 UM 不同）。因此，我们从此前针对预算约束的广告主的机制设计的文献 [4,6] 中借鉴了可变现社会福利（Liquid Social Welfare，LSW）的概念：

定义 6（可变现社会福利，LSW）：一个机制中的分配结果的可变现社会福利 是所有广告主对该分配结果的最大付费意愿之和，即 $Wel(\Pi) = \sum_{k=1}^{K} v_{\pi_k} x_k$。

我们定义可变现社会福利的比较基准为所有可能分配中的最大 LSW，即 $Wel_{OPT} = max_{\Pi} Wel(\Pi)$。此外，由于我们在这项工作中主要关注满足真实性的机制，所以在当上下文清楚时我们将不再区分 θ_i 和 $\hat{\theta}_i$。

三、公开类别信息

首先，我们考虑一个简单的设定，即广告主的类别是公开的。在这种情况下，该问题属于单参数机制设计的范畴，相对比较简单，但对于理解后面章节中我们设计的机制很帮助。针对这种简单情况，我们提出了以下针对公开类别信息的广告拍卖机制（Mechanism for bidders with PUblic classes，MPU）。

机制 1（MPU）:

- **分配：**按价值对广告主进行排名，并相应地从上到下分配坑位。
- **付款：**对于每个分配到坑位 $1 \le a_i \le K$ 的广告主 i，令 j 为下一个坑位的广告主，即 $a_j = a_i - 1$。
  - 如果 i 是一个 UM，那么 $p_i = \frac{1}{x_{a_i}} \sum_{k=0}^{a_j} v_{\pi_k}(x_{k+1} - x_k)$；
  - 如果 i 是一个 VM，那么 $p_i = v_j$。

直观地说，MPU 直接按价值将坑位分配给广告主，而不考虑其类别。此外，UM 的扣费遵循 VCG，VM 的扣费遵循 GSP（注意我们使用自下而上的坑位索引）。我们接下来表明，这个机制是 IC、IR 和稳健的，同时也保证了最佳的 LSW。

定理 1：当广告主的类别是公开信息时，MPU 满足 IC、IR 和稳健性，同时是 LSW 最优的。

参考此前基于 UM 模型和 VM 模型的文献，定理 1 的证明是直接的，因此这里将证明省略。

四、私有类别信息

在上一节中，我们提出了针对公开类别的混合广告主的最优机制。然而，当类别信息是私有的时候，这个问题就变成了一个多参数的机制设计问题，这在一般情况下很难解决。我们可以首先分析 MPU 在私有类别的设定下是否满足 IC 性质。我们可以很容易地观察到，如果一个 VM 把他的类别虚报为 UM，同时如实地报告他的价值，他有可能会享受到更低的扣费同时不改变他的分配。换句话说，MPU 在某种意义上对 VM 是不公平的，而这种不公平可能在类别信息是私有的情况下带来策略性操纵的可能性。

4.1 针对私有类别混合广告主的机制设计

上述分析意味着，在一个私有类别设定下的真实机制中，广告主的扣费应该只依赖于他被分配的位置，而不是他的类别。换句话说，假设一个广告主在两种情况下被分配到一个相同的坑位，而其他人的分配结果也是相同的，那么无论他是 UM 还是 VM，扣费都应该是一样的。这一要求促使我们基于坑位设计扣费规则，而非基于广告主类别设计扣费规则，这一规则应当基于该坑位下方的广告主的类型，同时应当结合 VCG 和 GSP。因此，我们提出将一个坑位的价格设定为以下两个条件的最大值：1）从下方最近的 UM 推导得出的 VCG 式扣费，以及 2）从下方最近的 VM 推导得出的 GSP 式扣费。具体来说，对于一个坑位 $k \geq 1$，令 k 下方最近的 VM 为 i_V，位于 k_V，而最近的下方的 UM 是 i_U，位于 k_U，那么，我们设定获得坑位 k 的扣费为：

$$p^{(k)} = max(\hat{p}_U^{(k)}, v_{i_V}), (1)$$

其中，$p^{(k)} = max(\hat{p}_U^{(k)}, v_{i_V}), (1)$

当没有低于坑位的 VM 或 UM 时，我们将相应的扣费项定为 0。基于这一扣费规则，

我们提出了混合广告主机制（MPR），它是 IC、IR 和稳健的，同时在 LSW 上达到了理想的近似比。

Algorithm 1: MPR

Input: The type profile θ of all bidders, the CTR x_k of all slots.

Output: The allocation and payment outcome for each bidder i.

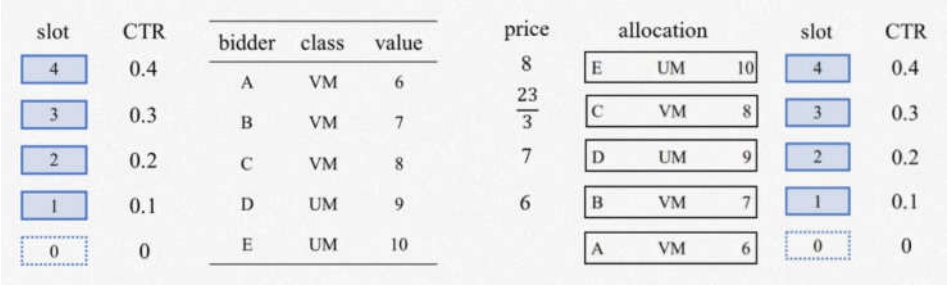
1 $p_i \leftarrow 0, \forall i \in \{1, ..., n\}$;
2 Sort all bidders by their values;
3 Let N be the set of top K bidders by their values, and i be the $(K+1)$th highest one;
4 $\pi_0 \leftarrow i$;
5 $S \leftarrow$ the set of all UMs in N;
6 $T \leftarrow$ the set of all VMs in N;
// Allocate slots to all VMs in T.
7 **if** $|T| > 0$ **then**
8 **for** k *from* 1 *to* $|T|$ **do**
9 $\pi_k \leftarrow$ the VM with the kth lowest value in T;
10 Update the payment $p^{(k)}$ for slots $1 \le k \le |T|+1$ by equation (1);
// Allocate slots to UMs iteratively.
11 **while** $|S| > 0$ **do**
12 $i \leftarrow \arg\min_{j \in S}\{v_j\}$;
13 $\bar{k} \leftarrow K - |S| + 1$;
14 $k^i \leftarrow \arg\max_{1 \le k \le \bar{k}}\{x_k(v_i - p^{(k)})\}$;
15 **if** $k^i \neq \bar{k}$ **then**
16 // Move existing bidders up one slot.
 for k *from* $\bar{k}$ *down to* $k^i + 1$ **do**
17 $\pi_k \leftarrow \pi_{k-1}$;
18 $\pi_{k^i} \leftarrow i$;
19 Update the payment $p^{(k)}$ for slots $k^i + 1 \le k \le \bar{k} + 1$ by equation (1);
20 $S \leftarrow S \backslash \{i\}$;
21 $p_{\pi_k} \leftarrow p^{(k)}, \forall k \in \{1, ..., K\}$;
22 Return π_k and p_i for each slot k and each bidder i.

图一　MRP 机制算法流程

MPR 的完整伪代码在图一中给出。MPR 的关键思想是用所有的 VM 来填充坑位，然后迭代地给一个 UM 分配一个具有最优效用的坑位。在图一中，MPR 首先根据广告主的价值对其进行排序，并获得价值排名前 K 名的广告主的集合 N（第 2-3 行）。然后将价值第 $(K+1)$ 高的广告主分配到索引为 0 的坑位，作为定价的基础

（第 4 行）。我们把 N 中所有的 UM 集合表示为 S，把 N 中所有的 VM 集合表示为 T（第 5–6 行）。接下来，MPR 将 T 中的所有 VM 根据它们的值依次填充到最低坑位（第 7–9 行）。然后，我们基于分配的 VM，按照公式一计算以下坑位的价格 $1 \le k \le |T|+1$（第 10 行）。事实上，由于在这一步中 S 中的 UM 在这一步没有一个被分配到坑位，所以 $\hat{p}_U^{(k)}$ 的项被设置为 0。因此，价格计算可以简化为 GSP 支付规则。在计算每个坑位的价格后，我们接着确定 UM 的分配。当前尚未分配的 UM 中价值最低的广告主 i 被选中，我们为他选择最优坑位 k^i，也就是具有最高效用值的坑位（如果存在并列的情况，则选择较低的那个坑位）（第 12–14 行）。值得注意的是，有两种坑位可能被选择：1) $k^i = \bar{k} = K - |S| + 1$，即在所有被分配的广告主之上的位置（注意在第一轮的时候 $K = |T| + |S|$，且 S 将在这个过程中被持续更新为所有未分配的 UM 的集合）；2) $k^i < \bar{k}$，即一个现有广告主所占据的位置。在前一种选择中，我们只需要将这个位置分配给 k^i 到 i；在后一种选择中，我们首先将所有在位置 k^i 和在该位置以上的广告主上移一个坑位，然后将该坑位 k^i 分配给 i（第 15–18 行）。接下来，价格 k^i 上面的坑位将根据新插入的 UM 的价值进行更新（第 19 行）。然后，该 UM i 被从 S 中删除，然后重复这个过程，直到所有 UM 在 S 中的所有 UM 都被分配了一个位置（第 20 行）。最后，如果一个广告主被分配到了坑位 k，他的点击扣费将是该坑位对应的价格；否则，他的付费将是 0（第 21 行）。

接下来，我们提供一个例子来说明 MPR 的运行过程。我们可以从这个例子中观察到一个有趣的结果：出价较低的 VM 可能比出价较高的 UM 获得更高的坑位。



图二 （左边）例 1 中的示例，（右边）分配结果与价格

例 1：假设有四个位置和五个广告主，他们的 CTR 和类别信息如图二（左边）所示。在 MPR 中，我们首先将具有第五最高价值的广告主，即广告主 A，放在虚拟坑位 0 上。然后，其余的 VM，也就是 B 和 C，被放置在坑位 1 和 2 上。因此，我们可以计算出 1，2，3 号坑位对应的价格，即 $p^{(1)} = 6, p^{(2)} = 7, p^{(3)} = 8$。有了这些价格，

我们就可以得到广告主 D 在坑位 1，2，3 分别对应的效用：0.3，0.4，0.3。然后，最优的坑位，也就是坑位 2 被分配给他，而广告主 C 则被移至坑位 3。接下来，坑位 3 和坑位 4 的价格通过公式一被更新为 $p^{(3)} = \frac{23}{3}$，$p^{(4)} = 8$。接着，我们得到广告主 E 在坑位 1，2，3，4 分别对应的效用：0.4，0.6，0.7 和 0.8。因此，坑位 4 被分配给了广告 E，最终分配结果如图二（右边）所示，广告主点击价格由其获得的坑位的相应价格给出。

4.2 理论性质

在证明 MPR 的理论性质之前，我们首先定义两个坑位的 _ 边际扣费增加值 _ 的概念，以衡量广告主获得更高坑位时的成本表现。基于这个概念，我们提出几个引理来帮助理解 MPR 背后的思想，这些引理将对 IC 和 IR 的证明有帮助。

定义 7： 对于两个坑位 $k' > k$，边际扣费增加值定义为：

$$\Delta(k, k') \triangleq \frac{p_{\pi_{k'}} x_{k'} - p_{\pi_k} x_k}{x_{k'} - x_k}$$

基于边际扣费增加值的定义，我们证明以下引理。由于篇幅原因，我们将证明过程略去。

引理 1： 如果一个 $\mathrm{UM}\, i$ 被分配到一个坑位 $a_i = k$，我们有 $\Delta(k, k+1) = v_i$。

引理 2： 对于两个广告主 i 和 j，如果 $v_i > v_j$ 且 $\tau_i = \tau_j$，即他们都是 UM 或都是 VM，那么我们有 $a_i > a_j$。

引理 3： 对于两个广告主 i 和 j，如果广告主 i 是一个 VM，而 j 是 UM，并且 $a_i < a_j$，那么我们有 $v_i < v_j$。

引理 4： 如果一个 UM i 被分配到一个坑位 $a_i = k$，那么我们有 $\Delta(k, k') \geq v_i, \forall k' > k$。

基于以上这些引理，我们证明以下定理，作为本工作的主要结果。由于篇幅原因，我们将证明过程略去。

定理 2：MPR 是一个稳健的机制。

定理 3：MPR 满足个体理性。

定理 4：MPR 满足激励相容性。

定理 5：MPR 在 LSW 上可以实现不超过 2 的近似比，即：$Wel_{MPR} \geq \frac{1}{2} Wel_{OPT}$。

定理 6：没有任何一种 IC、IR 且稳健的机制可以实现低于 1.25 的近似比。

五、结论

本文中，我们研究了 UM 和 VM 混合的广告环境中的机制设计问题，并提出了两种机制 MPU 和 MPR，分别在类别信息公共和私有的情况下能够保证 IC、IR、稳健性和（近似）最优 LSW。这项工作为后续关于投资回报率约束广告主的研究提供了启发，同时也产生了一些开放性问题。其中最主要的开放性问题是如何缩小近似比的下界和上界之间的差距。我们猜想 MPR 在某种程度上是解决这个问题的最优机制，在后续的研究中我们希望能严格证明这一点。此外，我们将对这一机制进行推广，使之能够适用于多种类型投资回报率约束的广告主。

参考文献

[1] Edelman, B.; Ostrovsky, M.; and Schwarz, M. 2007. Internet advertising and the generalized second-price auction: Selling billions of dollars worth of keywords. American economic review, 97(1): 242 - 259.

[2] Varian, H. R. 2007. Position auctions. International Journal of Industrial Organization, 25(6): 1163 - 1178.

[3] Wilkens, C. A.; Cavallo, R.; and Niazadeh, R. 2017. GSP: the cinderella of mechanism design. In Proceedings of the 26th International Conference on World Wide Web, 25 - 32.

[4] Deng, Y.; Mao, J.; Mirrokni, V.; and Zuo, S. 2021. Towards Efficient Auctions in an Auto-bidding World. In Proceedings of the Web Conference 2021, 3965 - 3973.

[5] Balseiro, S.; Deng, Y.; Mao, J.; Mirrokni, V.; and Zuo, S. 2021. The Landscape of Auto-Bidding Auctions: Value Versus Utility Maximization. In Proceedings of the 22nd ACM Conference on Economics and Computation, 132 - 133.

[6] Dobzinski, S.; and Paes Leme, P. 2014. Efficiency guarantees in auctions with budgets. In International Colloquium on Automata, Languages, and Programming, 392 - 404.

关于我们

阿里妈妈智能广告算法团队，超大业务体量和丰富的商业化场景使我们在深度学习、强化学习、机制设计、自动出价等技术方向极速成长并积累丰富，期待与高校师生及业界同行一起交流探讨，也欢迎加入我们（暑假实习生 / 研究生实习生 / 高校 AIR 计划合作 / 访问学者等）。

联系邮箱： alimama_tech@service.alibaba.com（投递简历请注明 – 智能广告算法团队）

自动出价下机制设计系列（二）：面向私有约束的激励兼容机制设计

作者：真柏、妙临

导读

这篇是阿里妈妈在自动出价下机制设计系列的第二篇工作（第一篇：面向异质目标函数广告主的拍卖机制设计）。在自动出价中，广告主的诉求以及与平台交互的模式发生了明显的变化。其中最值得关注的特点之一，是广告主的决策内容从具体的流量出价变成了对出价算法的设置调整。我们考虑了出价设置中最常见的两个约束，预算与投资回报率（ROI），推导相应的激励兼容条件，即在怎样的拍卖机制下广告主上报真实约束就可以最大化自身的利益，并进一步设计可实现的激励兼容机制。

摘要

自动出价已经成为在线广告拍卖的流行范式。不同于传统的手动竞价，自动出价中的广告主在一个周期内对多次广告拍卖进行累积的广告效果评估，并且拥有私有的经济约束。基于这些新的特点，我们考虑了适用于自动出价的拍卖模型：广告主将预算和投资回报率（ROI）等经济约束作为私有信息，且优化长期累积的广告效果。我们针对这种多维设定，从私有约束的角度推导了激励兼容的条件，并展示了任何可行的分配规则都可等效地简化为关于预算的一系列非递减函数。然而，从这些非递减函数映射而来的分配通常呈现出不规则的形状，这使得很难将机制设计目标闭式表达。为了克服这一设计难题，我们提出了一系列基于个性化排序函数且灵活、易实现的激励兼容自动出价拍卖，借鉴常见的广义第二价格（GSP）拍卖并做进一步设计。我们设计的背后思想是利用个性化排名分数作为分配物品的标准，并基于预算计算关键 ROI，以将预算的约束转化为与 ROI 相同的维度。

本文工作已被 IJCAI 2023 接收，欢迎阅读交流。

论文：Truthful Auctions for Automated Bidding in Online Advertising

作者：Yidan Xing, Zhilin Zhang, Zhenzhe Zheng, Chuan Yu, Jian Xu, Fan Wu, Guihai Chen

下载（点击↓阅读原文）: https://arxiv.org/abs/2301.13020

1. 背景

随着机器学习在在线广告领域取得的成功，广告主们开始转向采用自动出价工具，这为广告主和在线平台之间的互动带来了重大变革。在自动出价服务中，广告主向平台提交其高层次的优化目标和约束条件，然后由机器学习算法驱动的出价代理代表广告主在每次广告拍卖中做出详细的出价决策。通过自动出价工具，广告主能够从全局角度针对其经济约束优化其整体广告目标。在自动出价下，我们需要重新审视拍卖理论中的一个基本问题：在自动出价的新广告范式中，传统的拍卖模型是否仍然适用。由于平台可以获取有关广告主与用户之间互动的历史数据，我们可以估计用户的潜在行为（如点击和转化），这些行为可以被视为广告主对物品的估值。在自动出价中，广告主的私有信息实际上是其在整个广告投放过程的约束条件。这些与传统拍卖截然不同的新特点需要对应的新广告拍卖模型，以激励广告主真实地揭示其高层次的私有约束。

在这项工作中，我们考虑了一种新的自动出价拍卖模型，在这个模型中，广告主提交预算以及投资回报率（ROI）要求作为他们的私有约束，并希望在一定时期的多次拍卖中最大化其赢得流量的累积价值。我们分析了与预算和投资回报率这两个私有约束相关的激励兼容条件。我们证明了，针对这种多维设定的任何可行（激励兼容且个体理性）的拍卖机制都可以等效地表示为一系列以预算为输入的非递减函数。当这些非递减函数被映射为相应的拍卖机制时，呈现出一种新的价值分组现象：不同的预算 – 投资回报率类型被分组以共享相同的累积价值，并且分组模式由一个阈值投资回报率函数（从上述非递减函数转换而来）以及它的单调性确定。由于阈值投资回报率函数可以拥有任意的单调性，预算 – 投资回报率类型的分组形状通常是不规则的，因此我们难以获得拍卖优化目标（如收入和社会福利）的闭式表达式。

为了克服这些困难而设计出易于实现的拍卖，我们提出了一系列允许个性化排名分数的广告拍卖，这个框架可以提供优化各种设计目标的灵活性。我们的拍卖机制采用排名分数作为分配的依据，与广义第二价格（GSP）拍卖的思想相似。为了保证私有约束的真实上报，我们设计了一类关键 ROI，其定义为在不违反预算约束的情况下赢得最多物品的最大 ROI，从而等效地将预算转化为与 ROI 相同的维度，使我们能够在预算和 ROI 间找到相对更严格的约束，并利用这个约束防止虚报。

2. 拍卖模型

基于上述的自动出价广告系统，我们考虑如下的拍卖模型。有 n 个广告主竞争 m 个物品（用户流量），这些物品在一个时间段内按顺序出现在 m 个时间段中，每个时间段只会出现一个物品。在整个工作中，我们将广告主和竞拍者、物品和用户流量视为等价的术语。这些广告主是“价值最大化”的竞拍者，在经济约束没有被违反的前提下，他们的目标是最大化在所有时间段内分配到的物品的累积价值。每个广告主 i 都有预算（$B_i \geq 0$）和投资回报率（ROI）（$R_i > 0$）的约束，这些是私人信息，也可以称为类型 $t_i = (B_i, R_i)$。我们用 $\boldsymbol{t} = (t_i)_{i=1}^n$ 表示所有广告主的类型，其空间为 $\mathcal{T} = \prod_{i=1}^n \mathcal{T}_i$，其中 $\boldsymbol{t} \in \mathcal{T}$ 且 $\mathcal{T}_i = \mathcal{B}_i \times \mathcal{R}_i$。我们用 $\boldsymbol{t}_{-i} = (t_1, \ldots, t_{i-1}, t_{i+1}, \ldots, t_n)$ 和 $\mathcal{T}_{-i} = \prod_{k \neq i}^n \mathcal{T}_k$ 表示广告主 i 之外的其他竞拍者的类型。我们假设广告主对物品的估值是平台已知的公共信息，并用 $v_{i,j} > 0$ 来表示广告主 i 对物品 j 的估值。

在收集了所有竞拍者的预算和 ROI 之后，在线平台采用某个拍卖机制 $(\mathcal{A}, \mathcal{P})$ 来决定广告分配和支付，其中 $\mathcal{A}$ 表示一个（随机化的）分配规则 $\mathcal{A} : \mathcal{T} \to [0,1]^{n \times m}$，而 $\mathcal{P}$ 表示一个（随机化的）支付规则 $\mathcal{P} : \mathcal{T} \to \mathbb{R}^n$。具体来说，如果平台收集到的上报类型是 $\boldsymbol{t}' \in \mathcal{T}$，我们令竞拍者 i 被分配到物品 j 的概率表示为 $a_{i,j}(\boldsymbol{t}')$，而竞拍者 i 的预期支付表示为 $p_i(\boldsymbol{t}')$。对于任何物品 $j \in [m]$，其必须满足的分配约束为 $\sum_{i=1}^n a_{i,j}(\boldsymbol{t}') \leq 1$。竞拍者 i 在这些拍卖中的累积价值可以由此表示为 $v_i(\boldsymbol{t}') = \sum_{j=1}^m v_{i,j} a_{i,j}(\boldsymbol{t}')$，她的实际投资回报率则可以被计算为 $\mathrm{ROI}_i(\boldsymbol{t}') := v_i(\boldsymbol{t}') / p_i(\boldsymbol{t}')$（如果 $p_i(\boldsymbol{t}') = 0$，将其视为 $+\infty$）。

当竞拍者的真实类型为 $t_i = (B_i, R_i)$ 时，上报类型 t_i' 时的效用被定义为：

$$u_i(t_i, \boldsymbol{t}') = \begin{cases} v_i(\boldsymbol{t}'), \text{if } p_i(\boldsymbol{t}') \leq B_i \text{ and } \mathrm{ROI}_i(\boldsymbol{t}') \geq R_i, \\ -\infty, \text{otherwise}, \end{cases}$$

其中 $\boldsymbol{t}' = (t_i', \boldsymbol{t}'_{-i})$。

在这项工作中，我们专注于设计满足预算和投资回报率的占有策略激励兼容（DSIC）和个体理性（IR）性质的直接显示（direct-revelation）拍卖机制。

定义 1：如果拍卖机制满足以下条件，则称该拍卖机制是满足占有策略激励兼容（DSIC）性质的：$\forall i \in [n], t_i, t_i' \in \mathcal{T}_i, \boldsymbol{t}_{-i} \in \mathcal{T}_{-i}$: $u_i(t_i, (t_i, \boldsymbol{t}_{-i})) \geq u_i(t_i, (t_i', \boldsymbol{t}_{-i}))$

定义 2：如果拍卖机制满足以下条件，则称该拍卖机制是满足个体理性（IR）性质的：$\forall \boldsymbol{t} \in \mathcal{T}, i \in [n]$: $p_i(\boldsymbol{t}) \leq B_i$ 且 $\mathrm{ROI}_i(\boldsymbol{t}) \geq R_i$.

当竞拍者拥有经济约束时，我们用流动福利 (liquid welfare) 来替代传统的社会福利，用于衡量所有主体在拍卖中的总收益：

$$\mathrm{LW} = \sum_{i\in[n], u_i(t_i, \boldsymbol{t}')\geq 0} \min\left(\frac{v_i(\boldsymbol{t}')}{R_i}, B_i\right).$$

3. 机制设计的可行域

在这一节中，我们将推导机制设计的可行域，即符合占有策略激励兼容（DSIC）和个体理性（IR）性质的机制的设计范围。由于篇幅限制，我们会省略部分推导的步骤与证明细节，呈现主要定理，感兴趣的读者朋友可查阅原论文。

为了使拍卖中私有约束和私有价值之间的区别更加直观，我们考虑只有一个私有约束（预算）的情况，并固定另一个约束（ROI）为真实值。为了使竞拍者真实地上报预算，我们需要确保竞拍者不会通过报告更小或更大的预算来获得更高的效用。因为报告更小的预算不会导致竞拍者打破她原有的预算约束，因此减少预算应当导致获得的效用等于或小于原先的效用。对于报告更大预算且获得更高价值的竞拍者，平台需要对其进行收费并打破原有的预算约束以防止虚报。

引理 3：只有当拍卖机制满足以下性质时，该拍卖机制在预算上满足占有策略激励兼容（DSIC）性质：$\forall i \in [n],\ R \in \mathcal{R}_i,\ \boldsymbol{t}_{-i} \in \mathcal{T}_{-i}$：(1) $v_i((B,R),\boldsymbol{t}_{-i})$在 B 上单调递增；(2) 若 $v_i((B',R),\boldsymbol{t}_{-i}) > v_i((B,R),\boldsymbol{t}_{-i})$对于某些 $B' > B$ 成立，那么 $p_i((B',R),\boldsymbol{t}_{-i}) > B.$

对应地，我们可以写出当只有 ROI 是私有约束时的 DSIC 条件。

引理 4：只有当拍卖机制满足以下性质时，该拍卖机制在投资回报比上满足占有策略激励兼容（DSIC）性质：$\forall i \in [n]$, $B \in \mathcal{B}_i$, $\boldsymbol{t}_{-i} \in \mathcal{T}_{-i}$：(1) $v_i((B,R),)\boldsymbol{t}_{-i}$ 在 R 上单调递减；(2) 若 $v_i((B,R'),\boldsymbol{t}_{-i}) > v_i((B,R),\boldsymbol{t}_{-i})$，对于某些 $R' < R$ 成立，那么 $ROI_i((B,R'),\boldsymbol{t}_{-i}) < R$。

上述引理中的单调递增 / 递减是非严格的。引理 3 与引理 4 自然地构成了 (B,R) 双参数 DSIC 的必要条件。我们进一步证明了，它们实际上也构成了充要条件，也就是说，当竞拍者无法通过在单参数上的虚报获得更高效益时，也同样无法通过双参数上的同时虚报来获得更高的效益。

定理 5：拍卖机制满足引理 3 与引理 4 是占有策略激励兼容（DSIC）性质的充要条件。

接下来，我们想要在机制设计时对分配和支付规则进行分解。我们找到了一个“万

能”的支付规则，满足如下性质：只要对于特定的分配规则，存在某个支付规则使得拍卖机制是 DSIC 与 IR 的，那么该支付规则就一定可以使其与分配规则组成的拍卖机制保持 DSIC 与 IR。该支付规则在给定的分配与约束下，收取最大的可能支付，其具体形式是：

$$p_i((B_i, R_i), \boldsymbol{t}_{-i}) = \min(v_i((B_i, R_i), \boldsymbol{t}_{-i})/R_i, B_i)$$

然后，将其重新代入定理 5 中，我们发现类型的累积价值与其预算乘以 ROI 的乘积之间的关系严格限制了其相邻类型的累积价值（定理 6）。

定理 6：分配规则可以与某个支付规则组成 DSIC 和 IR 的拍卖机制，当且仅当其满足如下条件：$\forall \boldsymbol{t} \in \mathcal{T}, i \in [n]$：(1) $v_i((B, R), \boldsymbol{t}_{-i})$ 在 B 上单调递增，在 R 上单调递减；(2) 若 $v_i((B_i, R_i), \boldsymbol{t}_{-i}) > \lim_{B \to B_i^-} v_i((B, R_i), \boldsymbol{t}_{-i})$，那么 $v_i((B_i, R_i), \boldsymbol{t}_{-i}) \geq B_i \times R_i$；(3) 若 $v_i((B_i, R_i), \boldsymbol{t}_{-i}) > \lim_{R \to R_i^+} v_i((B_i, R), \boldsymbol{t}_{-i})$，那么 $v_i((B_i, R_i), \boldsymbol{t}_{-i}) \leq B_i \times R_i$。

在定理 6 中，一个重要的观察是，对于每个预算，必定存在一个阈值 ROI，在此之后，即使上报的 ROI 要求降低，分配的累计价值也不会再增加，其定义如下：

$$\mathrm{thr}_i(B, \boldsymbol{t}_{-i}) = \sup_{R \in \mathcal{R}_i} (v_i((B, R), \boldsymbol{t}_{-i}) \geq B \times R)$$

定理 7：

$\forall i \in [n], \boldsymbol{t}_{-i} \in \mathcal{T}_{-i}, B_i \in \mathcal{B}_i, R_i \leq \mathrm{thr}_i(B_i, \boldsymbol{t}_{-i})$：$v_i((B_i, R_i), \boldsymbol{t}_{-\boldsymbol{i}}) = \mathrm{thr}_i(B_i, \boldsymbol{t}_{-i}) B_i$.

进一步，我们发现所有 ROI 高于其预算对应的阈值 ROI 的类型都必须与其同行相邻的类型（相同 ROI，不同预算）具有相同的累积价值。

定理 8：$\forall i \in [n], \boldsymbol{t}_{-i} \in \mathcal{T}_{-i}, B_i \in \mathcal{B}_i \setminus \{0\}, R_i > \mathrm{thr}_i(B_i, \boldsymbol{t}_{-i})$：

$v_i((B_i, R_i), \boldsymbol{t}_{-i}) = \lim_{B \to B_i^-} v_i((B, R_i), \boldsymbol{t}_{-i})$.

结合定理 7 与 8，我们发现阈值 ROI 函数可以唯一定义相应的累积价值函数，只要预算乘以其阈值 ROI 是单调递增的，而且这个映射关系是双射的。为了方便表示，我们定义 $g_i(B_i, \boldsymbol{t}_{-i}) = \mathrm{thr}_i(B_i, \boldsymbol{t}_{-i}) \times B_i$。我们将 $\mathcal{V}$ 表示为在固定 i 和 $\boldsymbol{t}_{-i}$ 的情况下可行的累积价值函数（$v_i : T_i \to \mathbb{R}^+$）的空间，并将 $\mathcal{G}$ 表示为非严格递增函数 $g : \mathbb{R}^+ \to \mathbb{R}^+$ 且满足 $g(0) = 0$ 的空间。

定理 9：存在一个双射的映射 $m : \mathcal{G} \to \mathcal{V}$，其定义为：

$$(m \circ g)(B,R) = \begin{cases} g(B) & \text{if } R \le \dfrac{g(B)}{B}, \\ g\left(\sup\limits_{B' \le B}\{R \le \dfrac{g(B')}{B'}\}\right) & \text{otherwise,} \end{cases}$$

其中我们假设 $\sup \varnothing = 0$。

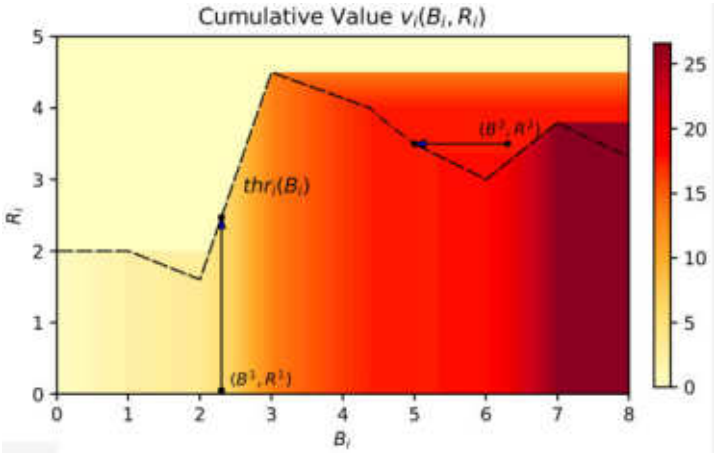


图 1. 从阈值 ROI 函数映射到累积价值的一组图例（固定）

也就是说，给定阈值 ROI 函数后，每种类型都会通过在垂直或水平方向上与阈值 ROI 曲线相交来与某些类型共享相同的累积价值，且所有类型的累积价值都会以这种方式被确定。具体来说，阈值 ROI 曲线上的类型需要拥有与其自身 $B \times R$ 相同的累计价值，阈值 ROI 曲线下方的类型会在垂直方向上向上搜索，而其上方的类型在水平方向上向左搜索，从而与阈值 ROI 曲线相交，并与该交点拥有相同的累积价值。我们可以通过设计阈值 ROI 函数来设计相应的拍卖机制，而每个满足 DSIC 与 IR 的拍卖机制也独特地对应一种阈值 ROI 函数。在这种结构下，由于阈值 ROI 曲线上方的类型通过在水平方向上向左搜索而与阈值 ROI 曲线相交，不同类型会根据阈值 ROI 函数的单调性而被分组，并拥有相同的累积价值，但是，由于阈值 ROI 函数本身可以拥有任意的单调性，在整个可行域中获得设计目标的通用表达式是困难的。

4. 基于排序函数的激励兼容机制设计

基于以上的机制设计可行域分析，我们提出了一类基于排序函数的激励兼容机制，其排序函数灵活可变，且机制易于实现。设计的关键思想是采用提前确定的排序函数为每个物品分别对广告主进行排名，并将阈值 ROI 设计为赢得足够多的物品以消耗完

预算的最大 ROI。这在我们的拍卖中被称为关键 ROI，模拟了广告主在面对竞拍价格时的最优反应。该机制设计的具体流程如下：

Algorithm 1: A Family of Simple Truthful Auctions

Input: Bidder's reported type (B_i, R_i), and non-increasing rank score functions $f_{i,j}$.

Output: Bidder's allocated items A_i and payment p_i.

Initialize $A_i = \{\}$ for $i \in [n]$;
Compute virtual bids $b_{i,j} \leftarrow v_{i,j} \times f_{i,j}(R_i)$;
for *each item* $j \in [m]$ **do**
  Find bidder i_0 with the highest virtual bid $b_{i_0,j}$;
  Record the second highest bid $c_j \leftarrow \max_{i \neq i_0} b_{i,j}$;
  Add item j into A_{i_0} and set $a_{i_0,j} = 1$;
for *each bidder* $i \in [n]$ **do**
  Compute ROI $r_{i,j} = f_{i,j}^{-1}(\frac{c_j}{v_{i,j}})$ required for winning the item $j \in A_i$;
  Find a largest $R_i^c \in \mathcal{R}_i$ s.t. $\sum_{j \in A_i} \mathbb{I}(R_i^c \leq r_{i,j}) \times \frac{v_{i,j}}{R_i^c} \geq B_i$, and let $d \leftarrow$ the difference between left-hand side and right-hand side of the above inequality;
  if $d > 0$ **then**
    Find the items j' with $r_{i,j'} = R_i^c$, and remove the items with value d/R_i^c;
  if $R_i \leq R_i^c$ **then**
    Remove the items with $r_{i,j} < R_i^c$ from A_i;
  $p_i \leftarrow \min\left(\frac{\sum_{j \in A_i} v_{i,j} \cdot a_{i,j}}{R_i}, B_i\right)$;
return (A_i, p_i) for $i \in [n]$.

在该机制中，给定广告主上报的预算和 ROI，我们首先基于排序函数计算不同广告主对于每个物品的虚拟出价。只要这些排序函数在 ROI 上是单调递减的，我们就可以保证最终的拍卖机制是满足 DSIC 与 IR 的。接下来，我们将每个物品分配给排序分数最高的广告主，并根据第二高的排序函数计算她赢得此物品所需要的 ROI。为了保证约束的真实上报，我们使用前面提到的基本规则来计算关键 ROI，即赢得足够多的物品以消耗完预算的最大 ROI，其中我们使用关键 ROI 作为实际 ROI 来计算支付。即使广告主的真实 ROI 低于此关键 ROI，她最终的分配也会根据此关键 ROI 进行调整。最后，我们按照之前找到的“万能”支付规则进行扣费。这样的机制是 IC 的，从直觉上来说是因为我们根据关键 ROI（代表了预算）和上报 ROI 中更紧的一项决定分配，如果广告主通过虚报而获得更好的分配，她就一定有至少一个真实的约束被违反了，反而会导致负效益。

5. 实验结果

我们在不同的自动出价环境中验证了所提出的拍卖机制的性能。在这里仅简要描述实验设定，完整的设定请参照原论文。为了模拟自动出价中广告主的不同类型与对物品的估值，我们令从均匀分布中取值，并考虑如下的两种环境：(1) i.i.d. 环境，即每位广告主的参数均从同样的均匀分布中取值；(2) non-i.i.d. 环境，每位广告主的 v, B, R 可以从特定的高 / 低均匀分布中取值，共 8 种广告主。我们采用了重复一价和二价拍卖，以及基于线性规划的理论最优解作为基准，这三种基准拍卖都不满足 DSIC 性质。我们基于流动福利、拍卖收入和公平性对不同机制进行评估，公平性定义为 $\text{fairness} = \min_{i\in[n]} \min\left(\frac{v_i(\boldsymbol{t}')}{R_i}, B_i\right)$。我们将排序函数设计为 $f_{i,j}(R) = \alpha_{i,j} \times e^{-\beta R}$ 的形式，其中 $\alpha_{i,j}$ 从正态分布 $N(\mu_i, \sigma_i^2)$ 中采样，β, μ_i, σ_i 是提前设置的常数，根据不同的实验环境选取这些参数。

实验的结果如下：

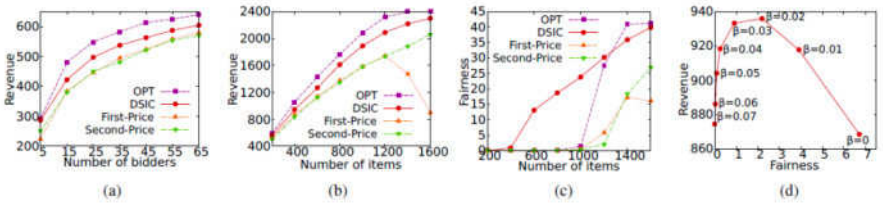


Figure 3: Revenue and fairness in different experiment setting for symmetric bidders: (a) Revenue for different number of i.i.d bidders with 200 items; (b) Revenue for different number of items with 40 i.i.d bidders; (c) Fairness for different number of items with 40 i.i.d bidders; (d) Fairness of our DSIC auction with 40 i.i.d bidders and 400 items using different parameter β in rank score functions;

在 i.i.d. 环境中，可以观察到，相较于重复一价与二价拍卖，我们的方法在收入和流动福利方面表现更好，并且更接近于理论最优解（图 3(a) 与 3(b)）。我们发现，为了实现更高的收入，不可避免地需要进行差别化的分配，而这种公平性和收入之间的权衡可以通过排序函数中的参数进行调整（图 3(c) 与 3(d)）。

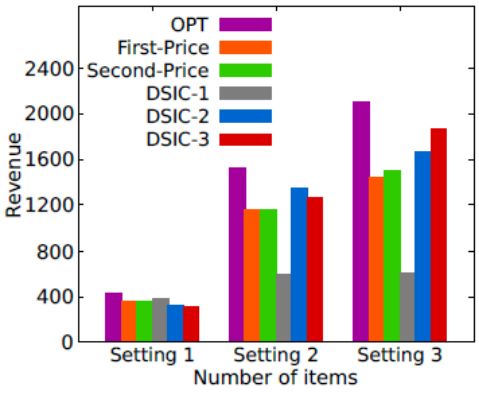


Figure 4: Revenue with 40 mixed bidders in Setting 1: 200 items; Setting 2: 1000 items; Setting 3: 1600 items; with three different groups of rank score functions for our DSIC auction

在 non-i.i.d. 环境中，我们不再呈现与 i.i.d. 环境中类似的趋势，而进一步说明选择适当的排名分数函数的重要性。我们选择了三组分别在各自对应得设定中性能良好的排序函数，并在其他两种设定中测试其性能。结果在图 4 中呈现，其中 DSIC-n 表示在第 n 个设定中表现良好的真实拍卖。DSIC-n 机制只在第 n 个设定中表现良好，其原因可能来自于为了在不同数量的物品下实现高收入而需要不同的分配方法。例如，在物品短缺的情况下（设定 1），DSIC-1 拍卖将大部分物品分配给 v 高、ROI 低的竞标者，导致在设定 2 和 3 中向这些广告主分配过多的物品而导致浪费。

6. 结论

本文中，在自动出价场景下我们考虑了具有私有预算和投资回报率约束的广告主在多物品拍卖中的拍卖机制设计，其中物品的价值是公开的信息。我们刻画了这个新的自动出价拍卖模型中的 DSIC 和 IR 条件，其中涉及到使竞拍者不同类型共享累积效用的新颖分组约束。我们提出了一系列易于实现且可灵活调整的激励兼容拍卖机制，实验结果验证了所提出的拍卖机制的性能表现。

部分参考文献

[Edelman et al., 2007] Benjamin Edelman, Michael Ostrovsky, and Michael Schwarz. Internet advertising and the generalized second-price auction: Selling billions of dollars worth of keywords. American Economic Review, 97(1):242 - 259, March 2007.

[Aggarwal et al., 2019] Gagan Aggarwal, Ashwinkumar Badanidiyuru, and Aranyak Mehta. Autobidding with constraints. In International Conference on Web and Internet

Economics, pages 17 – 30. Springer, 2019.
[Balseiro et al., 2021b] Santiago R Balseiro, Yuan Deng, Jieming Mao, Vahab S Mirrokni, and Song Zuo. The landscape of auto-bidding auctions: Value versus utility maximization. In Proceedings of the 22nd ACM Conference on Economics and Computation, pages 132 – 133, 2021.
[Balseiro et al., 2022] Santiago Balseiro, Yuan Deng, Jieming Mao, Vahab Mirrokni, and Song Zuo. Optimal mechanisms for value maximizers with budget constraints via target clipping. In Proceedings of the ACM Conference on Economics and Computation, page 475. ACM, 2023.

关于我们

阿里妈妈智能广告算法团队，超大业务体量和丰富的商业化场景使我们在深度学习、强化学习、机制设计、自动出价等技术方向极速成长并积累丰富，期待与高校师生及业界同行一起交流探讨，也欢迎加入我们（暑假实习生 / 研究生实习生 / 高校 AIR 计划合作 / 访问学者等）。

联系邮箱： alimama_tech@service.alibaba.com（投递简历请注明 – 智能广告算法团队）

增广拍卖——二跳页下的拍卖机制探索

作者：溯渊、淮竹

1. 引言

本文提出的方案已被 WSDM 2023 接收，论文：Boosting Advertising Space: Designing Ad Auctions for Augment Advertising，

下载：https://dl.acm.org/doi/abs/10.1145/3539597.3570381

信息流产品为了保障用户体验通常会严格限制信息流中的广告曝光占比。稀缺的广告展示坑位难以满足广告主日益增长的广告投放需求，并且有限的广告供给量会进一步加剧广告主之间的竞争，导致点击价格 (ppc) 水位过高，进而降低了中小广告主的投放意愿。为了缓解这种情况，我们设计了将多个广告“打包”成一个二跳页的广告投放机制——**“增广拍卖”**。本文聚焦于大规模在线广告系统的可扩展性的增广广告拍卖机制设计。为了阐述清晰，本文会结合手淘的“微详情”二跳信息流的业务模式阐述增广拍卖的设计方案。值得说明的是，除了微详情页外该机制能够应用到任何其他支持二跳页的用户产品形式中，比如外投二跳页等。

如图 1(b) 所示，微详情业务下的增广广告的展示页面由三个部分组成：主页，**“二跳广告”页 (i.e., 微详情广告页)** 和商品详情页。在增广广告中，淘宝主页信息流展示的广告被称为**“引导广告”**。当用户点击主页面上的“引导广告”时，用户将被显示一个“二跳广告”页。出于用户体验的考虑 (i.e., 所见即所得)，“二跳广告”页首坑位置展示的是用户在主页点击的“引导广告”。由于用户点击“引导广告”隐式表达出用户对该类相关物品的兴趣，所以“二跳广告”页在首坑以下的坑位展示的是与被点击的“引导广告”强相关的信息流广告。我们把这些“二跳广告”页上**非首坑**的广告称为”**二跳广告 (微详情广告)**”。在微详情页中，广告的展示模式为微详情广告卡模式，每个微详情卡片展示了简短的产品信息，如价格、评价、销售量，短视频等；并为用户提供了更多的互动操作，如购买、添加到购物车、添加到收藏夹等。

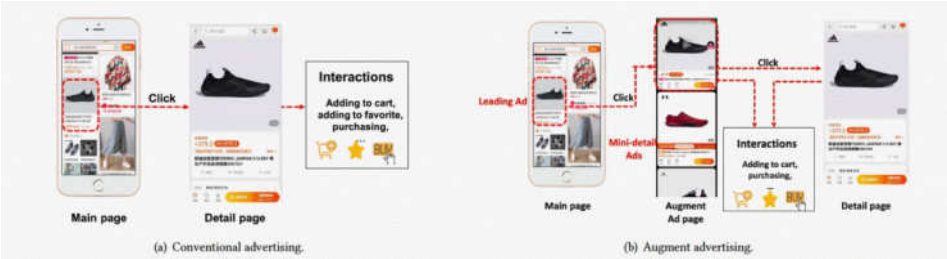


Figure 1: Examples of the two types of advertising. (a) In conventional advertising, once a user clic[illegible]on a leading ad, she will be directly show the product detail page. (b) In augment advertising, when a user clicks on a leadi[illegible]ad, she is guided to an augment ad page, which displays additional mini-detail ads relevant to the leading ad in a feed manner.

为保障客户体验并兼顾平台长期收益，我们不仅要考虑投放广告的整体效率，还必须考虑到扣费设计对广告主策略行为所带来的影响。对增广广告的广告拍卖机制设计需要考虑到以下几个问题：

1. **在大规模系统中，“引导广告”和“二跳广告”的异步检索过程导致“引导广告”排序效率优化的困难。**考虑到工程链路的效率问题，只有当用户点击“引导广告”后，系统才会生成二跳页。因此，当我们在主页决定“引导广告”排序时，其对应二跳页的潜在价值是未知的，这导致了优化增广广告整体社会福利的困难。
2. **广泛应用的按点击计费方案存在潜在的“搭便车“问题，可能导致广告主的利益受损。**按点击计费的方案要求“引导广告”的广告主分别为用户在主页和“二跳广告”页面的两次点击分别付费。而用户可能在二跳页被其他相关广告分散注意力，不会再次点击“引导广告”以进入详情页。在这种情况下，“引导广告”会为主页面上的无效点击付费，而其他广告主则从“引导广告”所付费的流量中获益，这将影响“引导广告主”的投入产出比。因此，我们需要设计一个新的计费方案来避免这种“搭便车“的问题。
3. **两阶段的拍卖机制之间的关系不清晰，导致现有拍卖机制经济学性质难以保证。**从优化增广广告整体效率来说，“引导广告”的排序必定受到二跳页的潜在价值的影响。然而，广泛应用的 GSP 拍卖和 VCG 拍卖由于没有考虑到二跳页的潜在价值无法实现社会福利最大化，进而不能保证拍卖的经济学性质，影响广告市场的长期繁荣。

针对这个两阶段的广告展示场景，我们首先将拍卖设计解耦为一个两阶段的拍卖机制设计，包括一个“**引导广告”拍卖**和一个“**二跳广告”拍卖** (i.e., 微详情广告拍卖)。我们为“引导广告”拍卖设计了新的 Potential Generalized Second

Price~(PGSP) 拍卖机制，并将计费点后移到用户在二跳页对“引导广告”的再次点击以应对“搭便车”问题。同时，PGSP 机制对广告的排序和扣费都考虑了潜在二跳页的收益。我们证明了新提出的拍卖机制存在非空的对称纳什均衡 (Symmetric Nash Equilibrium, SNE)。相较于广泛应用的 GSP 和 VCG 广告拍卖机制，我们提出的机制在社会福利和广告收入上均有提升。至于“二跳广告”拍卖，在给定“引导广告”的前提下，其投放形式跟传统的广告投放形式一致，因此我们仍采用常规的 GSP 拍卖机制。

2. 问题建模

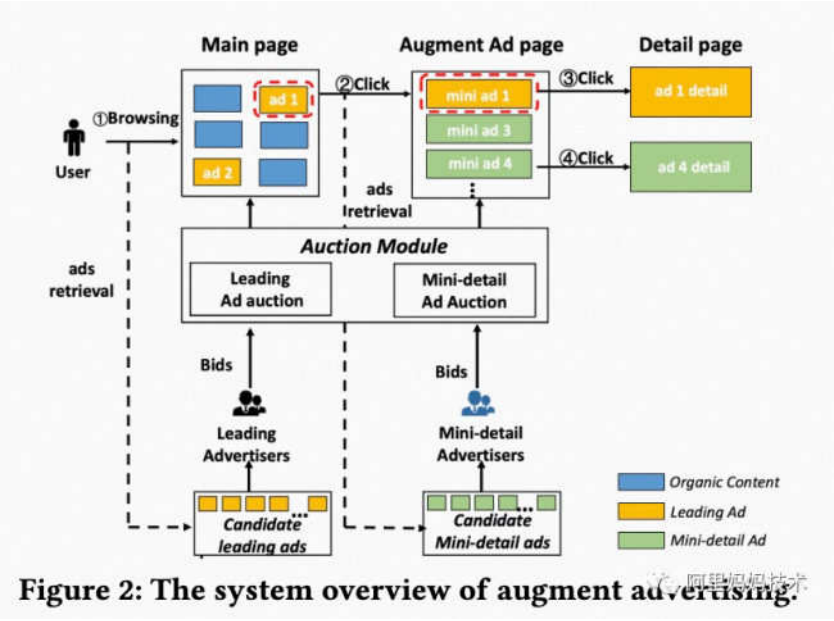


Figure 2: The system overview of augment advertising.

信息流主页广告及其“二跳广告”页面的拍卖过程被解耦为两个独立拍卖：“**引导广告”拍卖**和“**二跳广告”拍卖**。如图 2 所示，在用户访问淘宝主页时，平台将发起一次“引导广告”拍卖以确定主页的“引导广告”排序与计费。当用户点击了主页中的“引导广告”后，平台会发起一次“二跳广告”拍卖以确定对应的“二跳广告”页的广告排序与计费。每次拍卖过程包括都包括完整的召回，粗排，精排，计费过程。

一次“引导广告”拍卖中有 n 个广告主的集合 $N = \{1, \ldots, n\}$ 在竞争主页上的 K 个“引导广告”坑位。每个广告主 $i \in N$ 都有一个私人估值 v_i，他会向系统提出一个竞价 b_i 以表示愿意为这次点击支付的最高价格。

$\boldsymbol{b}=(b_i,\boldsymbol{b}_{-i})$表示所有“引导广告主”的出价，其中$\boldsymbol{b}_{-i}=(b_1,\ldots,b_{i-1},b_{i+1},\ldots,b_n)$。我们对主页的 K 个坑位采取分离点击率的假设，即“引导广告”被二次点击的概率可以分解为 $\gamma_i=\beta_j\times c_i$，其中 β_j 是曝光率，表示广告在第 j 个坑位被曝光给用户的概率。很明显，曝光率 β_j 是随着位置降低而递减的，即 $\beta_1\geq\beta_2\geq\cdots\geq\beta_K$。$c_i$ 是广告商品的 pCTR，即被广告曝光后被首次点击的概率。α_i 表示用户再次点击二跳情广告页上的“引导广告”的概率。在本文中，我们不失一般性地假设广告 i 为放在第 i 个位置的广告。

在用户点击“引导广告” i 后，平台将会发起一轮二跳拍卖。在该拍卖的召回阶段中，系统对召回物料做一次相关性过滤，精排后的候选广告集合为 $N_i=\{1,2,\ldots,n_i\}$。v_{il},γ_{il} 代表“二跳广告” $l\in N_i$ 的估值和点击率。在收到一条用户对主页的请求后，“引导广告”的排序和计费由拍卖机制 $\mathcal{M}^L=(\mathcal{X}^L,\mathcal{P}^L)$ 决定。具体来说，分配规则 $\mathcal{X}^L$ 是一个函数，它将所有广告主的出价作为输入，并输出广告序列 $W\subseteq N$ 和 $W_i\subseteq N_i$，其中 W 表示在主页上的“引导广告”序列，W_i 表示“引导广告” i 所对应的“二跳广告”页面上的“二跳广告”序列。在产生了广告序列后，平台会按照计费规则 $\mathcal{P}^L$ 计算每个“引导广告” i 在“二跳广告”页被再次点击所需要付出的费用 p_i。因此，获胜的“引导广告主” i 的效用是 $u_i=\gamma_i\alpha_i(v_i-p_i)$。

“引导广告”拍卖的目标是最大化所有广告主的期望社会福利 (expected social welfare)，使得满足即我们需要分别确定广告序列 W 和 W_i 来最大化下面这个式子：

$$\max_{W,W_i}\sum_{i\in W}\gamma_i(\alpha_i v_i+\sum_{l\in W_i\subseteq N_i}\gamma_{il}v_{il}).\ (1)$$

这个式子代表“引导广告”以及对应“二跳广告”的期望社会福利总和。为了广告市场的稳定性，我们还要求广告拍卖机制存在某种博弈论的均衡，比如对称纳什均衡。

定义 1. 我们称一组广告主达到了对称纳什均衡 (Symmetric Nash Equilibrium)，如果任意一个广告主都不能通过与其他人交换位置而得到更好的 utility，即对任意 $i,j,\ \beta_i(v_i-p_i)\geq\beta_j(v_i-p_j)$。

3. 机制设计

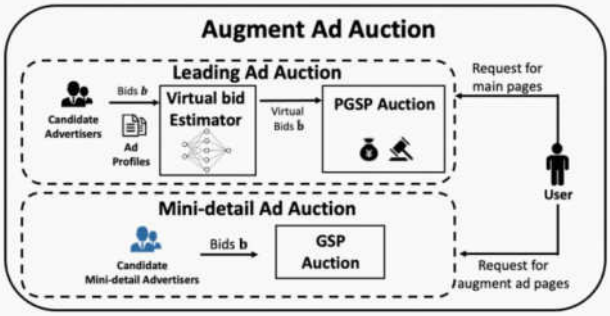


Figure 3: The auction processes for augment advertising. Leading ads are determined by the PGSP auction and mini-detail ads are determined by the GSP auction.

阿里妈妈技术

3.1 引导广告拍卖机制

在“引导广告”拍卖中，我们首先假设已经得知“二跳广告”的社会福利 $g_i = \sum_{l \in W_i} \gamma_{il} v_{il}$，那么式子 (1) 的优化目标就可以转换为

$$\max_{W} \sum_{i \in W} \gamma_i(\alpha_i v_i + g_i). \quad (2)$$

那么我们可以定义“引导广告”的虚拟竞价为：

$$\hat{b}_i = \alpha_i b_i + g_i. \quad (3)$$

这样，“引导广告” i 在第 j 坑位展示的期望社会福利即 $\beta_j c_i \hat{b}_i$。由于 β_j 是随着位置下移而递减的，所以我只需要将“引导广告”按照 $c_i \hat{b}_i$ 降序排序并取前 K 个广告即可得到最大化式子 (1) 的最优展示序列。基于此项观察，我们设计了 Potential GSP (PGSP) 拍卖机制 $\mathcal{M}^L = (\mathcal{X}^L, \mathcal{P}^L)$，详细定义如下：

- 排序方案 $\mathcal{X}^L$：我们按照期望社会福利 $c_i \hat{b}_i$ 对“引导广告”做降序排序，并取前 K 个“引导广告”作为最终展示序列。
- 扣费方案 $\mathcal{P}^L$：为了避免“搭便车”问题的出现，我们对用户在主页对“引导广告”的第一次点击不计费，只对在二跳页对“引导广告”的二次点击进行计费。我们采用 GSP 的计费机制的思想，对每个“引导广告”收取他能保持当前位置的最小价格。在我们新的排序方案下，每个曝光广告主的价格为

$$p_i = \frac{(\alpha_{i+1} b_{i+1} + g_{i+1}) \times c_{i+1} - g_i c_i}{\alpha_i c_i}. \quad (4)$$

我们可以证明 PGSP 机制一定存在非空的对称纳什均衡解。我们将证明过程放在了附录中。

定理 1. 在 PGSP 拍卖机制中一定存在一个非空的对称纳什均衡解。

此外，考虑到如果 $g_i c_i > (\alpha_{i+1} b_{i+1} + g_{i+1}) \times c_{i+1}$，$p_i$ 存在为负数的情况。这种情况在较少发生，因为二跳页的收益 g_i 通常远小于“引导广告主”的 eCPM。尽管如此，我们为了尽量避免这种情况的发生，我们为每一个广告都设定了一个足够低的保留价 ϵ，并另最终的点击价格为 $\hat{p}_i = \max(p_i, \epsilon)$。

在“二跳广告”拍卖中，我们直接采取 GSP 拍卖机制，就不再做过多介绍。整体拍卖流程如图 3 所示。

3.2 虚拟竞价的预估

最后需要解决的问题就是如何预测每个“引导广告”所产生的二跳页面的收益。这可以建模为一个回归任务并使用轻量级的机器学习模型进行预测。

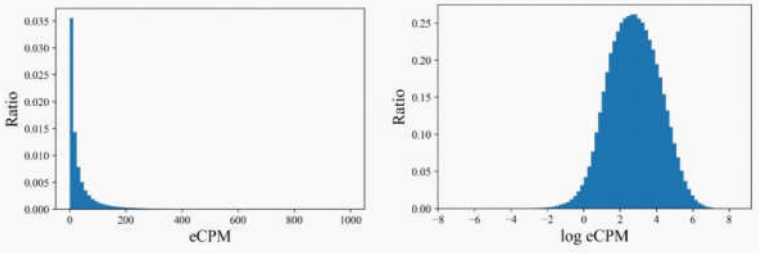
- **标签**：我们考虑了两种方案来作为预估标签。第一个方案是使用用户在二跳页的点击所产生的广告收入作为标签。第二个方案是使用用户在二跳页曝光广告的 ecpm 总和作为标签进行预估。

然而，如果采用第一个方案，如果将广告收益定义为二跳页增量收益，则与式子 (1) 中最大化 social welfare 的目标不一致。此外，样本中会存在大量广告消耗为零的样本，这些由于部分用户在二跳页为未点击任何广告而直接退出所导致的。这些样本会导致模型学习到的分布存在严重的偏差。

在采用第二个方案前，我们随机抽取了部分“引导广告”并观察其所产生的二跳页中曝光广告的 eCPM 总和的分布。我们发现 eCPM 的对数是服从高斯分布的，如图 4 所示。所以我们最终采用二跳页曝光广告的 log ecpm 作为标签进行训练回归模型。

- **特征**：我们采用了用户特征（用户 id），“引导广告”特征（品牌，销量，价格等）以及 RTP 模型的预测信息（pCTR，pCVR）作为模型输入特征。值得一提的是，为了保证广告拍卖机制的理论性质，我们将广告主的 bid 剔除在输入特征外。

- **模型：**考虑到链路时延问题，我们采用简单的多层神经网络来搭建回归模型。



(a) eCPM of exposed mini-detail ads. (b) log eCPM of exposed mini-detail ads.

Figure 4: Histograms of eCPM and log eCPM of exposed mini-detail ads.

阿里妈妈技术

4. 实验

为了充分验证业务以及提出的拍卖机制的表现，我们使用淘宝主页信息流采集的数据集对我们提出的增广广告业务以及 PGSP 拍卖机制进行了离线实验。

4.1 离线实验

我们在离线实验环节中分别验证了采用 PGSP 机制是否能带来更多的广告收益? 以及在采用 PGSP 机制的前提下，virtual bid 的预测模型的准确度对用户的浏览和广告收益有什么影响?

首先，我们验证与广泛应用的 GSP 拍卖机制和 VCG 机制相比，在增广广告业务中采用 PGSP 机制是否能够给广告收益和用户体验带来增益。我们关注如下评价指标：

- 每条流量平均点击量 (CPS): $CPS = \frac{\sum_m click + \sum_a click}{\sum session}$，其中 $\sum_m click$ 代表用户在主页产生的点击，$\sum_a click$ 代表用户在对应二跳页面产生的点击。
- 每条流量平均曝光量 (IPS): $IPS = \frac{\sum impression}{\sum session}$
- “引导广告”的平均价格 (Avg.CPC).
- 每条流量平均广告收益 (RPS)：$RPS = \frac{\sum_a click \times CPC}{\sum session}$

CPS 和 RPS 反映了用户对投放的广告的满意程度。在图 6 (a) 和图 6(b) 中，我们比较了在主页投放 K 个坑位的情况下不同拍卖机制产生的 CPS 和 IPS，其中 $K \in [1, 5]$。从图中我们可以得知，相较于 GSP 和 VCG 机制，PGSP 机制所产生的“引导广告”可以吸引用户在二跳页面产生更多的浏览和点击。这是由于 PGSP 拍卖

机制在排序时考虑了潜在的二跳页的质量，从而将更容易吸引用户浏览和点击相关广告的“引导广告”排在前面。

在图 6 (c) 中我们同时观察了在不同拍卖机制下每个坑位的平均价格。我们发现 PGSP 拍卖机制下每个坑位的平均价格会略小于使用 GSP 拍卖机制的平均价格，这说明 PGSP 拍卖机制能在一定程度上缓解拍卖的竞争激烈程度。虽然 PGSP 拍卖的平均价格更低，但是从图 6 (d) 中我们可以得出 PGSP 拍卖机制相比 GSP 拍卖机制和 VCG 拍卖机制来说可以产生更多的广告收益，这大多是来源于用户在二跳页更多的点击所带来的收益。

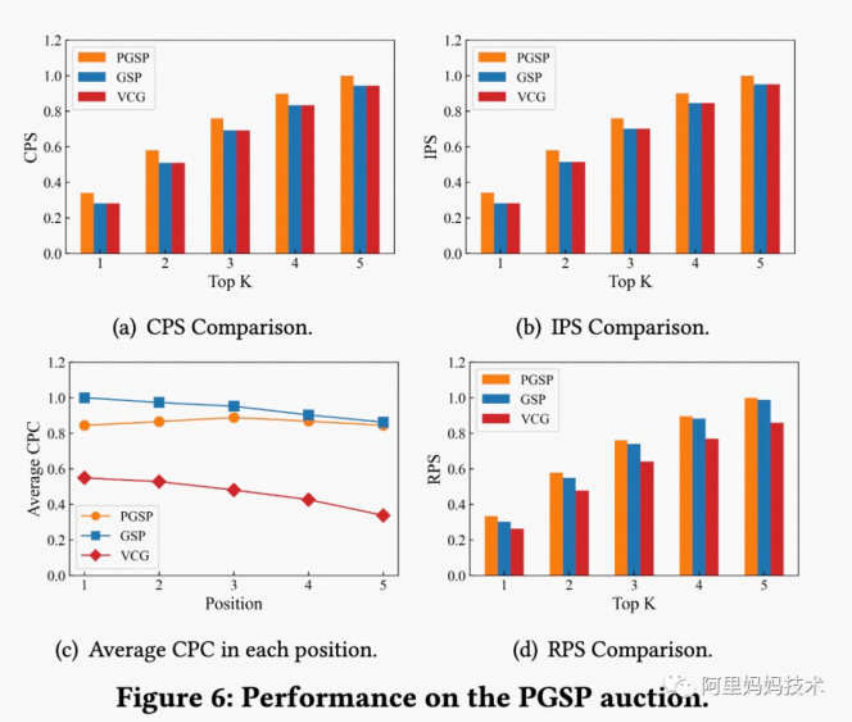


Figure 6: Performance on the PGSP auction.

然后，我们评估了模型的性能和广告收益的之间的关系。由于没有可以参考的相关工作，我们仅使用线性回归模型 (LR), GBDT 模型和 MLP 模型作为比较模型。我们采用均值模型 (MEAN Model) 作为比较基准。均值模型使用广告商品的二跳页收益的历史均值作为 virtual bid 中的预估值。

我们采用 Normalized Root Mean Square Error (NRMSE) 来评估不同模型的表现，同时使用前五个坑位的累计 RPS, CPS，IPS(用 RPS@5, CPS@5，IPS@5 表示) 来评估不同模型下的广告收益，如表 2 所示，我们可以推论出使用 MLP 模型

能够得到更好的模型表现，同时也能相应提升更多的广告收益。

Table 2: Performance of Virtual Bid Estimation Models.

Metric	MEAN	LR	GBDT	MLP
NRMSE	0.881	0.879	0.850	**0.840**
CPS@5	–	+1.244%	+1.390%	**+1.730%**
IPS@5	–	+1.473%	+1.472%	**+2.012%**
RPS@5	–	+1.151%	+1.313%	**+1.681%**

阿里妈妈技术

4.2 在线实验

我们在淘宝主页的“猜你喜欢”信息流中对增广广告业务模式进行了为期三周的在线 A/B 测试，并观察了投放增广广告相较于投放传统展示广告的 PV,Clicks,GMV, 广告收益 (REV) 和平均价格 (AVG.CPC) 的提升。值得一提的是这些指标都是将二跳页的数据纳入了统计范围。如表 3 所示，通过投放增广广告业务，平台的 PV,Click,GMV 都得到相应的提升。拍卖的平均价格因为更多的广告展示坑位缓解了拍卖的竞争压力而得到了下降。尽管如此，采用增广广告形式的广告收入仍然得到了增长。

Table 3: Online performance improvement of the augment advertising compared with conventional advertising.

Metric	Week1	Week2	Week3
PV	+3.077%	+3.128%	+3.591%
Clicks	+12.939%	+13.408%	+14.904%
GMV	+2.781%	+2.088%	+3.278%
REV	+0.876%	+0.540%	+0.690%
Avg.CPC	-9.639%	-9.091%	-9.41[illegible]%

阿里妈妈技术

5. 总结

本文介绍了一种新的广告投放模式——增广广告，它通过在用户浏览路径中插入广告二跳页面的方式扩充广告的展示空间，以探索广告收入的潜在提升空间。我们为这个新形式广告展示形式设计了两阶段解耦的广告拍卖，即“引导广告”拍卖和“二跳广告”拍卖。为了优化整体效率，我们为“引导广告”拍卖设计了被称为 PGSP 的拍

卖机制，它基于数据模型预估的 virtual bid 对“引导广告”进行排序和计费。我们从理论上证明了潜在 Potential Generalized Second Price~(PGSP) 拍卖的经济性质，并在淘宝主页信息流上部署了增广广告业务，大量的离线和在线实验数据表明增广广告相比一些现有方案在用户指标和广告收益指标上都能得到更好的效果。

6. 参考文献

[1] Xiang Chen, Bowei Chen, and Mohan Kankanhalli. Optimizing trade-offs among stakeholders in real-time bidding by incorporating multimedia metrics. In Proceedings of SIGIR, page 205 – 214, 2017.

[2] Paul D ü tting, Felix A. Fischer, and David C. Parkes. Truthful outcomes from non-truthful position auctions. In Proceedings of EC, page 813, 2016

[3] Negin Golrezaei, Max Lin, Vahab Mirrokni, and Hamid Nazerzadeh. Boosted second price auctions: Revenue optimization for heterogeneous bidders. In Proceedings of SIGKDD, page 447 – 457, New York, NY, USA, 2021.

[4] Varian H R. Position auctions[J]. international Journal of industrial Organization, 2007, 25(6): 1163-1178.

[5] Xiangyu Liu, Chuan Yu, Zhilin Zhang, Zhenzhe Zheng, Yu Rong, Hongtao Lv, Da Huo, Yiqing Wang, Dagui Chen, Jian Xu, Fan Wu, Guihai Chen, and Xiaoqiang Zhu. Neural Auction: End-to-End Learning of Auction Mechanisms for E-Commerce Advertising. In Proceedings of KDD, pages 3354 – 3364, 2021.

[6] Yiqing Wang, Xiangyu Liu, Zhenzhe Zheng, Zhilin Zhang, Miao Xu, Chuan Yu,and Fan Wu. On designing a two-stage auction for online advertising. In Proceedingsof WWW , page 90 – 99, New York, NY, USA, 2022.

关于我们

我们是阿里妈妈展示广告机制策略算法团队，致力于不断优化阿里展示广告技术体系，驱动业务增长，推动技术持续创新；我们不断升级工程架构以支撑阿里妈妈展示广告业务稳健 & 高效迭代，深挖商业化价值并优化广告主投放效果，孵化创新产品和创新商业化模式，优化广告生态健壮性；团队创新工作发表于 KDD、CIKM、WSDM、AAAI 等领域知名会议。在此真诚欢迎有 ML 背景的同学加入我们 ~

投递简历邮箱（请注明 – 展示广告机制策略）：

alimama_tech@service.alibaba.com

Score-Weighted VCG：考虑外部性的智能拍卖机制设计

作者：远乎、井枫、妙临

1. 摘要

智能广告拍卖机制（Learning-based Ad Auction Design）在在线广告中扮演着越来越重要的角色。但现有的方法并没有很好的考虑外部性，如自然结果会影响到广告结果的点击率。在本文中，我们提出了一种考虑外部性的广告拍卖通用框架，即Score-Weighted VCG。该框架将考虑外部性的最优拍卖设计分解为两部分：单调得分函数的学习和加权福利最大化分配算法的设计。我们通过完备的理论证明了该框架在各种感知外部性的点击率模型下都能产出满足激励兼容和个体理性的近似最优广告拍卖。此外，我们提出一种基于匹配的分配算法来实现所提出的框架的方法。实验效果表明该算法能显著提升拍卖的优化目标（如 Revenue 和 Social welfare）。该项工作依托阿里妈妈 – 北大人工智能创新联合实验室完成，基于该项工作整理的论文已被 KDD 2023 接收，欢迎阅读交流。

论文：Learning-Based Ad Auction Design with Externalities: The Framework and A Matching-Based Approach

作者：Ningyuan Li, Yunxuan Ma, Yang Zhao, Zhijian Duan, Yurong Chen, Zhilin Zhang, Jian Xu, Bo Zheng, Xiaotie Deng

下载（点击↓阅读原文）：https://dl.acm.org/doi/abs/10.1145/3580305.3599403

2. 引言

广告拍卖机制在广告系统中起到重要的作用，直接决定了广告效果和平台收入。在典型的点击计费（CPC）多坑广告拍卖场景下，在一次用户请求中有多个广告坑位以供销售。广告拍卖机制获取候选广告主的出价，根据分配和定价算法，决定广告位的分配结果以及点击后广告主需要支付的价格。广告物料和来自推荐系统的推荐物料一起向用户展示。从经济学视角看，最优广告拍卖设计可以看作一个优化问题：目标是最大化期望收入，即广告主的总付费，同时需要满足激励相容性（Incentive Compatibility, IC）和个体理性（Individual Rationality, IR）的约束。IC 要求广告主

真实报价总是能最大化其自身效用，而 IR 要求广告主付费不超过其对广告点击的真实估值。

可以看出，广告拍卖机制的设计依赖于对于以点击率（CTR）为代表的广告商品展示效果的精确预估，但在实际场景中，商品展示效果会受到相互之间的外部性影响。例如，当类似种类的商品同时展示时，用户会比较它们的价格、外观和功能，以作进一步决策，因此每个商品的展示效果都受到周围商品的影响，称为外部性影响，这一现象在近年来开始受到学术和工业界的广泛关注。然而，传统的广告拍卖通常简化或忽略了外部性。例如，广泛使用的 GSP 拍卖机制基于可分离 CTR 模型（Separable CTR Model），假定广告的点击率只由广告内容和位置决定，而完全忽略了其他商品的影响。因此传统的广告拍卖机制在考虑外部性时不再适用。

外部性影响对于最优广告拍卖的设计带来了许多挑战。由于现实中外部性影响依赖于用户特征、商品特征等，具有很高的复杂性，通常只能进行黑盒建模。由于广告的点击率受到上下文中其他商品的影响，即使对分配进行微小修改，也可能导致广告拍卖的预期收入发生复杂的变化。一般而言，当对于外部性结构不作具体假设时，计算具有最大社会福利（Social Welfare）的分配方案是 NP 困难的。因此，如何设计高效实用的分配算法是一个非平凡的问题。另一方面，由于外部性影响的存在，拍卖机制更难控制每个广告主得到的效用，因此 IC 和 IR 等约束更难满足。

本文旨在提出一个数据驱动的广告拍卖框架，以在考虑外部性的情况下实现收入最大化，同时确保满足 IC 和 IR 约束。我们首先结合理论分析提出 Score-Weighted VCG 框架，将最优拍卖机制的设计拆解为一个单调得分函数的学习和一个加权福利最大化算法的设计。基于这一框架，我们提出一个实用的实现方案，利用数据驱动的模型实现最优拍卖机制。

3. 建模与前置结论

我们将任意外部性影响下的多广告位拍卖建模为一个单参数拍卖模型（即每个广告主在一次请求中仅提交一个报价，代表一次点击对其价值，而不是对每个坑位分别报价），其中可行分配集合反映了外部性影响，然后利用理论文献结果 [1] 对最优拍卖机制进行刻画。

假设在一次请求中，页面上有 m 个广告坑位需要分配给 n 个广告主。每个广告主 i 的特征信息表示为一个向量 c_i，包含其广告的内容、类别等特征以及与用户和自然结

果的交互特征。每个广告主对于一次点击的真实估值为 v_i，服从一个由 c_i 决定的先验分布。每个广告主根据自己的真实估值 v_i 向拍卖机制提交出价 b_i，随后广告拍卖机制根据全部广告主的出价决定一个分配方案，表示为一个 $n \times m$ 的 0-1 矩阵 $\mathbf{a}$，其中 $\mathbf{a}_{i,j} = 1$ 当且仅当第 j 个坑位分配给了第 i 个广告主。

在外部性影响下，每个广告的点击率可能依赖于整个分配方案。我们使用 $\mathrm{CTR}_i(\mathbf{a}; \mathbf{c})$ 表示当分配方案为 $\mathbf{a}$、全体广告主特征为 $\mathbf{c} = (c_1, \cdots, c_n)$ 时广告主 i 的广告被用户点击的概率。特别地，如果在 $\mathbf{a}$ 中广告主 i 没有得到广告坑位，则 $\mathrm{CTR}_i(\mathbf{a}; \mathbf{c}) = \mathbf{0}$。

一个拍卖机制 $M = (A, P)$ 由分配规则 $A(\mathbf{b}; \mathbf{c})$ 和定价规则 $P(\mathbf{b}; \mathbf{c})$ 组成。分配规则 $A(\mathbf{b}; \mathbf{c})$ 输出一个分配方案，而定价规则计算每个广告主的广告的点击价格 $P_i(\mathbf{b}; \mathbf{c})$。在点击率模型 CTR 下，每个广告主 i 的期望点击量记为 $x_i(\mathbf{b}; \mathbf{c}) = \mathrm{CTR}_i(A(\mathbf{b}; \mathbf{c}); \mathbf{c})$，而期望付费记为 $p_i(\mathbf{b}; \mathbf{c}) = x_i(\mathbf{b}; \mathbf{c}) \cdot P_i(\mathbf{b}; \mathbf{c})$。每个广告主希望最大化自己的效用 $u_i(v_i; \mathbf{b}; \mathbf{c}) = v_i \cdot x_i(\mathbf{b}; \mathbf{c}) - p_i(\mathbf{b}; \mathbf{c})$。

拍卖机制通常需要满足激励相容性（Incentive Compatibility, IC）与个体理性（Individual Rationality, IR）的条件。IC 要求对每个广告主来说，无论其他广告主出价为何，诚实报价 $b_i = v_i$ 总是能最大化他的效用 $u_i(v_i; \mathbf{b}; \mathbf{c})$。这使我们可以总是假设 $b_i = v_i$。而 IR 要求广告主的效用总是非负，即 $P_i(\mathbf{b}; \mathbf{c}) \le v_i$。

广告拍卖机制设计的目标是在 IC 和 IR 条件约束下最大化平台期望总收入 $\mathrm{Rev}(M; \mathbf{b}; \mathbf{c}) = \sum_{i \in [n]} p_i(\mathbf{b}; \mathbf{c})$，可以写为一个带约束优化问题：

$$\max_{M=(A,P)} \mathbb{E}_{b,c}[\mathrm{Rev}(M; \mathbf{b}; \mathbf{c})], \text{s.t. IC and IR}$$

根据拍卖理论文献 [1]，最优拍卖机制的充分必要条件是：

1. 分配规则满足单调性，即 $x_i(b_i, \mathbf{b}_{-i}; \mathbf{c})$ 关于 b_i 单调不减。
2. 分配规则最大化“虚拟福利”（Virtual Welfare）$\sum_{i \in [n]} \tilde{\phi}(b_i; c_i) \cdot x_i(\mathbf{b}; \mathbf{c})$，其中 $\tilde{\phi}(b_i; c_i)$ 是 Myerson 虚拟估值函数 [1]，由广告主的真实估价分布 $F(v_i | c_i)$ 决定。
3. 定价规则由分配规则唯一确定：

$$p_i(\mathbf{b}; \mathbf{c}) = b_i x_i(b_i, \mathbf{b}_{-i}; \mathbf{c}) - \int_0^{b_i} x_i(t, \mathbf{b}_{-i}; \mathbf{c}) dt$$

4. Score-Weighted VCG 框架

为了更好地处理外部性影响，我们提出一个通用框架，称为 Score-Weighted VCG（SW-VCG）框架，将一个单调得分函数 Scr 和一个加权福利最大化算法 Alc 组成一个拍卖机制。$\mathrm{Scr}(b_i; c_i)$ 尝试估计 $\tilde{\phi}(b_i; c_i)$，同时需要关于 b_i 单调不减；$\mathrm{Alc}(\mathbf{s}; \mathbf{c})$ 对于任意向量 $\mathbf{s} = (s_1, \cdots, s_n)$ 尝试找到最大化加权福利 $\sum_{i\in[n]} \mathrm{CTR}_i(\mathbf{a}; \mathbf{c}) \cdot \mathbf{s_i}$ 的分配方案 $\mathbf{a}$。一个遵循 SW-VCG 框架的拍卖执行如下步骤：

1. 使用单调得分函数 Scr，将每个广告主 i 的出价 b_i 映射为他的分数 $s_i = \mathrm{Scr}(b_i; c_i)$。
2. 运行加权福利最大化算法 $\mathrm{Alc}(\mathbf{s}; \mathbf{c})$，输出分配方案 $\mathbf{a}$。
3. 根据公式 $p_i(\mathbf{b}; \mathbf{c}) = b_i x_i(b_i, \mathbf{b}_{-i}; \mathbf{c}) - \int_0^{b_i} x_i(t, \mathbf{b}_{-i}; \mathbf{c}) dt$ 计算定价

在论文中，我们从理论上证明了 SW-VCG 框架具有良好的性质。

1. SW-VCG 框架能够表达最优拍卖机制。当得分函数 $\mathrm{Scr}(b_i; c_i) = \tilde{\phi}(b_i; c_i)$，且 $\mathrm{Alc}(\mathbf{s}; \mathbf{c})$ 总是最大化加权福利 $\sum_{i\in[n]} \mathrm{CTR}_i(\mathbf{a}; \mathbf{c}) \cdot \mathbf{s_i}$ 时，构成的拍卖机制严格满足 IC 和 IR，并且获得理论最优的收益。
2. SW-VCG 框架是可学习的。泛化误差界的证明，表明从样本中学习 $\tilde{\phi}(b_i; c_i)$ 以得到得分函数 Scr 是可行的。
3. SW-VCG 框架对于误差有较强的鲁棒性。只需得分函数 $\mathrm{Scr}(b_i; c_i)$ 与 $\tilde{\phi}(b_i; c_i)$ 充分接近，且 $\mathrm{Alc}(\mathbf{s}; \mathbf{c})$ 所得的加权福利与最优值充分接近，其组成的 SW-VCG 拍卖机制就能获得接近最优的收益。

通过 SW-VCG 框架，我们将外部性下的拍卖机制设计分解为两个独立的子任务，其一是从广告主出价样本数据中学习虚拟估值函数 $\tilde{\phi}(b_i; c_i)$，其二是设计合适的算法求解在外部性影响下最大化加权福利的分配方案。我们可以发现前者与分配结果以及外部性影响下的点击率无关，而后者与广告主出价分布无关，因此分别解决两个子任务相比于直接着手设计在外部性下的广告拍卖降低了难度，更易于取得良好效果。

5. 基于匹配的 SW-VCG 拍卖模型

在 SW-VCG 框架下，我们提出了一个基于数据驱动方法的实现：引入对于外部性影响的结构化假设，设计了具有实用性的广告拍卖机制模型。我们的模型包括四个关键组件：1）特征预处理模块，2）单调评分函数模块，3）分数加权福利最大化模块，4）定价模块。

5.1 特征预处理模块

特征预处理模块使用一个多层神经网络，将每个广告主的特征 c_i 转换为一个表示向量 $e_i = \mathrm{MLP}^{\mathrm{preprocess}}(c_i)$，表示此广告主出价分布的特征。

为了更好的学习特征 embedding，我们采用了一种基于分位数回归 [2] 的预训练任务，帮助从原始特征中提取广告主出价分布特征。具体地，我们在 e_i 后添加了一个辅助神经网络，以表示向量 e_i 和分位数 $q \in [0,1]$ 作为输入，并输出估计出价 $\hat{b}_{i,q} = \mathrm{MLP}^{\mathrm{pretrain}}(e_i, q)$。我们使用一个变体的分位数回归损失函数：

$$\mathcal{L}^{\mathrm{pre}}(b_i, \hat{b}_{i,q}, q) = b_i[q(b_i - \hat{b}_{i,q})^+ + (1-q)(\hat{b}_{i,q} - b_i)^+]$$

对于 $\mathrm{MLP}^{\mathrm{preprocess}}$ 和 $\mathrm{MLP}^{\mathrm{pretrain}}$ 进行梯度下降训练。这使得 e_i 能够有效表示给定 c_i 时的出价分布，并区分不同的出价分布。

5.2 单调得分函数模块

单调得分函数模块将 (b_i, e_i) 映射到每个 $i \in [n]$ 的得分 s_i。为了保证 s_i 关于 b_i 中单调不减，我们将 e_i 转换为一个单调函数的参数，然后将该函数应用于出价 b_i 以获得得分 s_i。具体地，我们令 $s_i = H(b_i; e_i)$，$H(b_i; e_i)$ 是如下关于 b_i 的分段线性函数：

$$H(b_i; e_i) = \rho_0(e_i) + \sum_{k=1}^{K} \rho_k(e_i)\mathrm{clip}(\lambda_k(e_i)b_i + \mu_k(e_i), 0, 1)$$

其中 ρ_k, λ_k, μ_k 都由神经网络以 e_i 为输入计算得到。对于任意 $k = 1, \cdots, K$，我们保证 $\rho_k(e_i), \lambda_k(e_i)$ 非负，从而 $H(b_i; e_i)$ 关于 b_i 单调。可以证明这一结构能够近似任意单调不减的函数。

特征预处理模块和单调得分函数模块组成了 SW-VCG 框架中的得分函数 $\mathrm{Scr}(b_i; c_i)$。我们已经在前文提到了特征预处理模块的预训练方法，它指导特征预处理模块学习出价分布的特征。在预训练结束后，我们设计了一种训练整个得分函数的方法，与其后的分配算法完全解耦，不需要获取整个拍卖机制的分配结果就能实现得分函数的训练。通过分析一种假想的单广告主的拍卖场景，我们设计了一个随机化的损失函数：给定数据样本 (b_i, c_i)，我们从 $U[0, b_i]$ 中采样 b_i'，并定义损失函数：

$$\mathcal{L}(\mathrm{Scr}; b_i, b_i', c_i) = -b_i\big(\mathrm{Scr}(b_i; c_i) - \mathrm{Scr}(b_i'; c_i)\big) + \frac{1}{2}[\mathrm{Scr}(b_i; c_i)]^2$$

可以证明，使期望损失最小化的得分函数就是 Myerson 虚拟价值函数 $\tilde{\phi}(b_i; c_i)$。注意到这一损失函数提供了有意义的梯度，因而我们可以通过随机梯度下降等优化算法对于得分函数进行训练。由于每个广告主 i 的出价分布仅取决于她自己的特征 c_i，因此得分函数可以在不知道其他广告商的出价和最终分配的情况下正确地训练。

5.3 分数加权福利最大化模块

分数加权福利最大化模块以得分 $\mathbf{s}$ 和上下文信息 $\mathbf{c}$ 作为输入，并解决 SW-VCG 框架中的加权福利最大化问题。然而，当不对外部性影响的结构作任何假设时，这是一个 NP 困难问题，无法高效求解。在我们的实现中，我们作出一个实际假设，即点击率由一个基于位置的外部性模型给出，其中每个广告的 CTR 仅取决于上下文 $\mathbf{c}$ 和它被分配到的坑位，而与其他广告的分配无关。形式地，在任何将广告 i 分配给坑位 j 的分配 $\mathbf{a}$ 中，广告主 i 都获得相同的点击概率，表示为 $\mathrm{CTR}_{i,j}(\mathbf{c})$。我们可以写成：

$$\mathrm{CTR}_i(\mathbf{a}; \mathbf{c}) = \sum_{j=1}^{m} \mathrm{CTR}_{i,j}(\mathbf{c}) \cdot \mathbf{a}_{i,j}$$

这一 CTR 模型能够在实践中提供良好的准确性。我们观察到，在实际场景（如信息流）中，为了保证用户体验，页面上的大部分结果都是自然结果，而广告数量较少。因此，每个广告受到的外部性影响主要来自周围的自然结果。而自然结果的分配和上下文可以包含在 $\mathbf{c}$ 中，这允许我们充分精确地估计 $\mathrm{CTR}_{i,j}(\mathbf{c})$。此外，这种 CTR 模型的表达能力能够统一表示许多现有的 CTR 模型。例如，常用的可分离 CTR 模型可以表示为 $\mathrm{CTR}_{i,j}(\mathbf{c}) = \alpha_i \beta_j$，其中 α_i 和 β_j 分别表示可分离 CTR 模型中的广告质量和坑位质量。

这种 CTR 模型为加权福利最大化问题提供了足够的结构，消除了计算困难。在这种模型下，加权福利最大化问题可以写成：

$$\arg\max_{\mathbf{a} \in \mathcal{A}} \sum_{i=1}^{n} \sum_{j=1}^{m} s_i \cdot \mathrm{CTR}_{i,j}(\mathbf{c}) \cdot \mathbf{a}_{i,j}$$

这实际上是一个最大权二分匹配问题，权重为 $w_{i,j} = s_i \cdot \mathrm{CTR}_{i,j}(\mathbf{c})$。由于最大加权二分匹配问题可以使用匈牙利算法 [3] 在 $O(nm^2)$ 时间内解决，我们可以高效地获得加权福利最大化分配。

5.4 支付模块

根据拍卖理论文献，对于任何满足分配规则单调的广告拍卖 M，激励相容性成立的充要条件是期望付费满足 $p_i(\mathbf{b};\mathbf{c}) = b_i x_i(b_i, \mathbf{b}_{-i};\mathbf{c}) - \int_0^{b_i} x_i(t, \mathbf{b}_{-i};\mathbf{c})dt$，我们称之为 IC 定价。一般地说，可以使用使用数值积分或蒙特卡洛估计精确计算 IC 定价。

而在我们的实现中，由于我们采用了基于匹配的加权福利最大化模块，精确的 IC 定价可以确定性且高效地计算。给定得分向量 $\mathbf{s} = (s_1, \cdots, s_n)$，设 $\gamma_i(s_i, \mathbf{s}_{-i};\mathbf{c})$ 表示由最大加权二分匹配问题确定的分配中广告商 i 的 CTR，则 $\gamma_i(s_i, \mathbf{s}_{-i};\mathbf{c})$ 是关于 s_i 的一个单调不减的分段常数函数。[4] 提出的一种算法可以在 $O(nm^2)$ 时间内计算匹配算法的分配曲线，即 $\gamma_i(s_i, \mathbf{s}_{-i};\mathbf{c})$ 关于 s_i 的间断点。由于 $s_i = H(b_i; e_i)$ 是关于 b_i 的单调分段常数函数，我们可以高效地对所有 s_i 的间断点求逆，得到对应的 b_i。从而我们可以将 $x_i(b_i, \mathbf{b}_{-i};\mathbf{c})$ 表示为关于 b_i 的分段常数函数，并精确计算对每个广告主的 IC 定价。

6. 实验

我们在虚拟数据集和淘宝真实数据集上通过实验评估基于匹配的 SW-VCG 实现的有效性。对于虚拟数据，我们生成已知分布下的出价数据，并将我们模型的收入与理论上限进行比较。对于真实数据，我们使用淘宝广告系统重排拍卖的真实离线数据作为训练集和测试集，并将 SW-VCG 模型效果与常用的 VCG、GSP 以及仿射福利最大化拍卖（AMA）[5] 进行比较。

6.1 合成数据

对于合成数据，我们生成具有不同出价分布的数据集，包括均匀分布、指数分布、对数正态分布和一种非正则分布。基于这些出价分布，可以精确计算每个广告商出价的 Myerson 虚拟价值函数，并计算出理论上的最优收入。我们在不同的情形下评估我们的模型，考虑了广告主特征连续或离散、广告商出价的条件分布类型、生成点击率的方法三方面的差异。对于每个设置，我们使用包含 100000 次拍卖的训练集训练我们的 SW-VCG 模型，并使用 5000 次拍卖测试模型的收入，实验结果列在表 1 中。在几乎所有情形下，我们的模型都获得了达到理论最优收入 99% 以上的收入，并且显著优于 VCG 和 GSP。

	Uniform		Exponential		Lognormal		Irregular	
	Discrete	Continuous	Discrete	Continuous	Discrete	Continuous	Discrete	Continuous
Optimal	1.1361	1.1489	1.3362	1.3536	2.5349	2.5481	5.8086	5.8008
VCG	0.9430	0.9531	1.1806	1.1954	2.1885	2.1960	4.9998	5.0142
GSP	0.9833	0.9921	1.2652	1.2863	2.2955	2.3081	5.4358	5.4488
SW-VCG	**1.1347**	**1.1444**	**1.3268**	**1.3497**	**2.5166**	**2.5348**	**5.7635**	**5.7338**

表 1　在虚拟数据集上 SW-VCG 框架的效果表现

6.2 真实数据

我们还使用淘宝广告系统的真实日志数据进行了评估，涉及 8662 个广告商和 100000 次拍卖。对于来自用户的每个广告请求，在最终拍卖阶段，大约有 30 个广告商被广告系统选为候选人参加拍卖，并提交出价。为了获得得分函数使用的特征向量，使用了离散和连续特征，包括广告的 ID、广告类别 ID、广告的价格、用户的分类信息和用户历史行为。日志数据还包含每对广告 – 坑位的考虑外部性的 pCTR。

训练和测试数据集分别包括 90000 次和 10000 次拍卖。我们使用训练数据集训练我们的 SW-VCG 模型和作为基线的 AMA 拍卖模型，然后测试每个模型的收入和社会福利，同时也评估了 VCG 和 GSP 拍卖。实验结果列在表 2 中。我们可以看到，SW-VCG 模型获得了最高收入，显著高于所有基线。此外，与 AMA 拍卖相比，我们的 SW-VCG 模型获得了更高的收入和更高的社会福利。这一结果表明，我们的单调得分函数模块能够有效地近似数据分布的虚拟价值函数，使我们的模型能够获得更高的收入，同时避免不必要的社会福利损失。值得注意的是，GSP 拍卖不满足 IC 约束，这意味着在某些特殊设置下，GSP 可能获得超过任何 IC 拍卖的收入的理论上界的收入。人们可能因此担心从目前广泛使用的 GSP 切换到 IC 的拍卖会导致收入降低。然而，我们在真实数据上的结果显示，在所有场景下，我们提出的 SW-VCG 拍卖都比 GSP 获得了更多收入，这表明不需要为了收入牺牲 IC 性质。

	$n = 10, m = 3$		$n = 15, m = 5$		$n = 20, m = 10$		
	Revenue	Welfare	Revenue	Welfare	Revenue	Welfare	IC?
VCG	4.4733	**9.0483**	7.1009	**14.3719**	10.3498	**23.4883**	Yes
GSP	4.9305	**9.0483**	8.1896	**14.3719**	13.7292	**23.4883**	No
AMA	5.3876	6.6327	8.3196	10.4771	13.4584	17.5440	Yes
SW-VCG	**5.8274**	8.3461	**9.5755**	13.1567	**15.6588**	21.4515	Yes

表 2　在真实数据集上 SW-VCG 框架的效果表现

7. 总结

本文研究了在外部性影响下的最优多坑广告拍卖，并提出了 SW-VCG 框架，能够在达到最优收入的同时严格遵守 IC 和 IR 性质，且对于近似误差具有鲁棒性。基于 SW-VCG 框架，我们设计了一种适用于实际场景的基于匹配的数据驱动的广告拍卖模型。在合成数据和真实数据上的实验表明，我们的模型表现接近于理论最优结果，并在所有场景下都超过了现有方法。

参考文献

[1] Dütting P, Feng Z, Narasimhan H, et al. Optimal auctions through deep learning[C]//International Conference on Machine Learning. PMLR, 2019: 1706-1715.

[2] Koenker R, Bassett Jr G. Regression quantiles[J]. Econometrica: journal of the Econometric Society, 1978: 33-50.

[3] Kuhn H W. The Hungarian method for the assignment problem[J]. Naval research logistics quarterly, 1955, 2(1 - 2): 83-97.

[4] Cavallo R, Sviridenko M, Wilkens C A. Matching auctions for search and native ads[C]//Proceedings of the 2018 ACM Conference on Economics and Computation. 2018: 663-680.

[5] Sandholm T, Likhodedov A. Automated design of revenue-maximizing combinatorial auctions[J]. Operations Research, 2015, 63(5): 1000-1025.

合约广告中端到端流量预估与库存分配

作者：望羲

导读：

传统的合约广告售卖系统将流量预估和库存分配视为两个独立的模块两阶段求解，本文采用可微拉格朗日求解器，以新的端到端建模视角重新看待这一问题。

摘要

合约广告（Guaranteed Delivery Advertising）需要平台提前数周与广告客户签订合同，承诺提供广告展示次数并满足广告客户的定向要求。因此对于合约广告来说，除了投中分配策略，投前广告售卖技术也同样重要。传统的合约售卖技术将流量预估和库存分配视为两个不同的阶段。然而，这种两阶段优化难以保证最优的合约分配表现。本文旨在通过一种端到端方法获得更优效果。具体而言，我们提出了 Neural Lagrangian Selling(NLS) 模型，通过统一的学习目标对流量预估和库存分配进行统一建模。为此，我们首先开发了一个可微分的拉格朗日层，通过神经网络反向传播库存分配问题，使得网络直接优化最终的合约分配结果；然后为了有效地优化各种分配目标和约束，我们设计了一个图卷积神经网络，可以从二分配图中提取预测特征。实验表明，相比现有的两阶段方法，端到端的方法可以有效提升合约售卖的准确性。特别地，我们的优化层在计算效率和求解质量上均优于基线求解器。

论 文：End-to-End Inventory Prediction and Contract Allocation for Guaranteed Delivery Advertising

下载（点击↓阅读原文）: https://dl.acm.org/doi/pdf/10.1145/3580305.3599332

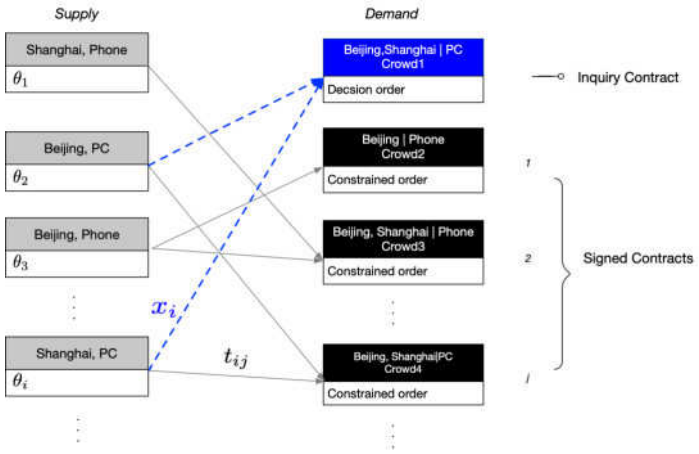
一、背景

合约广告客户会在广告投放的日期的几个月 / 几周前与平台签订合约，以提前锁定所需的广告展示次数。合同规定定向目标下（人群、频控、城市、渠道）的广告展现数量。所以，如果库存过度出售将导致合约难以完成保量（即展示次数少于目标量），平台需要赔付，而库存少售卖也会损害平台收入。随着合约广告业务的发展，我们发现已有研究存在以下问题：首先，大部分对合约广告的研究集中在合约广告投放阶段，

对合约广告投前售卖的研究相对较少；其次，之前投前预估一般采用流量预估和库存分配两阶段的建模方式，两阶段方法假定流量预估误差更小可以自动转化为更好的分配质量。然而，在复杂的决策情境中，该假设的实证证据并不明确。同时，学术界已有研究表明端到端的预测优化可以在多项任务上优于两阶段方法，我们调研了现有的端到端学习优化框架，如 QPTL 或 IntOpt，但是他们不能直接应用于合约广告售卖问题：由于合约售卖库存分配中必须同时考虑大量约束条件，可能导致通用求解器出现显著误差甚至难以求解，并且我们需要考虑售卖系统性能，因此必须设计一个高效的可微组合优化求解器。此外，先前对于端到端预估 + 分配的研究仅限于静态问题，而库存分配问题是复杂多变的，我们需要设计额外的网络结构来提取必要特征。为了解决以上问题，我们做了如下工作：

1. 我们以全新的视角处理合约投前售卖问题，借助深度学习进行端到端优化，我们利用设计的 Neural Lagrangian Selling(NLS) 直接最小化广告售卖问题的最终损失；
2. 我们推导了一个可微分的拉格朗日层用于库存分配问题。拉格朗日层可以处理通用的分配约束，并且在大规模问题上具有相对较低的计算复杂度；
3. 为了应对所有广告客户的复杂和动态约束，我们设计了一个基于库存分配的灵活 GCN 模块。通过简单适配，我们的框架可以扩展到其他不确定的二部图匹配问题；
4. 就预测和分配表现而言，我们的 NLS 显著优于两阶段方法和其他端到端方法。同时实现了更好的合约广告完成率和库存利用率。

二、问题建模



投前预估分配问题如上图所示，其中每个 supply 节点是库存预测和分配的最小单位。例如，一个 supply 节点可以表示一个特定的城市 / 设备组合，不同的广告主会定向不同的组合形式。在右侧，demand 节点表示广告客户的合同。给定一个新合同，我们的分配优化任务是在广告投放约束条件下，保证已签署合同都能分配到足够流量的前提下，最大化新合同的分配量。我们将表示新合同的 demand 节点视为决策节点，将已签署合同的 demand 节点称为约束节点。我们使用 x_i 表示从供应节点 i 到决策节点的流量分配比例，t_{ij} 表示从供应节点 i 到约束节点 j 的分配比例。

2.1 流量预估

在传统的两阶段预测分配方法中，首先训练流量预估模型（通常基于神经网络）。其中 θ 表示 supply 节点的库存，$\{(z_i, \theta_i) : i = 1, 2, \cdots\}$ 表示训练样本，其中特征 z_i 包括历史记录和上下文信息，如星期、月份、假期等。模型 $\theta \approx g(z; \omega)$ 通过优化参数 ω 来使预测损失最小化：

$$Loss = \|\theta - g(z; \omega)\|_2^2.$$

2.2 库存分配

库存分配问题可以被建模为：

$$\begin{aligned} \max_{x,t,u} \quad & (I \odot \theta)^\top x - p^\top u, \\ s.t. \quad & Gx + Bt - u \le h, \\ & u \ge 0, \\ & x, t \in [0, 1]. \end{aligned}$$

具体而言，我们优化决策变量 x，其中 x_i 表示将 supply θ_i 分配给新合同的比例。注意，$\theta \in \mathbb{R}^m$ 包含了所有 supply 节点的流量，$I \in \{0, 1\}^m$ 表示 supply 节点是否与决策变量相关。优化目标为最大化 $(I \odot \theta)^\top x$，即所有和决策变量相关的 supply 分配流量之和，也就是给新合同分配的最大库存。我们对已有合同进行重分配，即 t_{ij}。为了简化后续推导，分配矩阵 $T = (t_{ij}) \in \mathbb{R}^{m \times n}$ 被展平为向量 $t \in \mathbb{R}^{m*n}$，其中 n 是约束节点的数量。矩阵 $G \in \mathbb{R}^{k \times m}, B \in \mathbb{R}^{k \times (m*n)}$，向量 $h \in \mathbb{R}^k$ 用于定义一组分配约束条件 $Gx + Bt \le h$。为了确保我们有一个满足这些约束条件的可行解，我们在目标函数中惩罚 $p^\top u$，其中 $p \in \mathbb{R}^k$ 是惩罚系数（即不同约束条件的重要性），$u \in \mathbb{R}^k$ 是约束条件的松弛变量。所有合约定向要求比如人群、频控、目标量等都可以被建模在该约束

中，来确保优化问题的通用性和可行性。

2.3 两阶段与端到端

传统的两阶段方法首先通过最小化预测损失来预估流量，然后使用精确求解器来解决库存分配问题以优化分配决策。我们通过直接优化分配最终结果 Regret 达到端到端学习的目的：

$$Regret = \|(I \odot \hat{\theta})^\top \hat{x} - (I \odot \theta)^\top x\|_2^2,$$

其中 $\hat{\theta} = g(z; w)$ 是预测的库存量，$\hat{x}$ 是相应的分配决策。端到端方法可以从库存分配的历史决策案例中学习。通过 Regret 的定义可以推导出：

$$\frac{\partial Regret}{\partial \omega} = 2 \cdot \frac{\partial (I \odot \hat{\theta})^\top \hat{x}}{\partial \omega}((I \odot \hat{\theta})^\top \hat{x} - (I \odot \theta)^\top x)$$

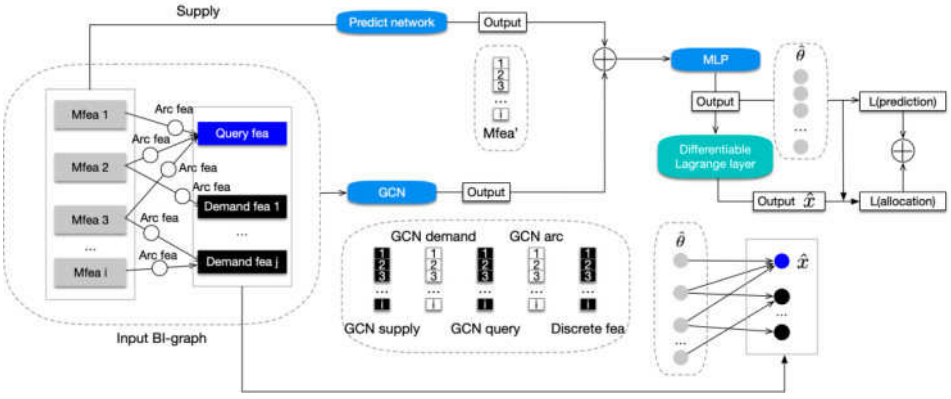
其中：

$$\frac{\partial (I \odot \hat{\theta})^\top \hat{x}}{\partial \omega} = \frac{\partial (I \odot \hat{\theta})^\top}{\partial \omega} \hat{x} + (I \odot \hat{\theta})^\top \frac{\partial \hat{x}}{\partial \omega}$$

$$= \frac{\partial (I \odot \hat{\theta})^\top}{\partial \omega} \hat{x} + (I \odot \hat{\theta})^\top \frac{\partial \hat{x}}{\partial \hat{\theta}} \frac{\partial \hat{\theta}}{\partial \omega}$$

显然，对于优化深度学习模型，$\frac{\partial \hat{\theta}}{\partial \omega}$ 可以自动计算，直接最小化 Regret 的挑战在于反向传播 $\frac{\partial \hat{x}}{\partial \hat{\theta}}$，必须通过可微分配优化问题来计算 Regret 的梯度。

三、问题求解



我们的模型结构如上图所示。具体而言，预测模块提取与库存相关的特征。预测模块使用与两阶段方法中相同的神经网络设计实现。GCN 模块从分配二分图中提取重要信息。Lagrangian 求解器模块是最具挑战性的部分，需要实现库存分配模型的正向和反向传播。我们的方法是松弛库存分配问题并推导相应的 KKT（Karush–Kuhn–Tucker）条件。下面介绍细节的部分。

3.1 拉格朗日对偶优化

为了方便后续推导，我们首先将原库存分配问题松弛为 QP 问题：

$$\ell(x,t,u) = (I \odot \theta)^\top x - p^\top u - \frac{\lambda}{2}\left(\|x\|_2^2 + \|t\|_2^2\right)$$

约束为：

$$\begin{gathered} Gx + Bt - u \leq h \ , \\ u \geq 0 \ , \\ x, t \in [0,1] \end{gathered}$$

对偶问题公式为：

$$\min_{\alpha,\beta \geq 0} \max_{x,t,u} L(x,t,u,\alpha,\beta),$$

可得拉格朗日公式为：

$$L(x,t,u,\alpha,\beta) = \ell(x,t,u) + \alpha^\top (h + u - Gx - Bt) + \beta^\top u$$

通过 KKT 条件计算一阶导数 $\frac{\partial L}{\partial x} = 0$，$\frac{\partial L}{\partial t} = 0$，$\frac{\partial L}{\partial u} = 0$，可以得到对偶问题的解：

$$\begin{gathered} x = \min(\max(\frac{I \odot \theta - G^\top \alpha}{\lambda}, 0), 1), \\ t = \min(\max(\frac{-B^\top \alpha}{\lambda}, 0), 1), \\ \alpha \odot (h + u - Gx - Bt) = 0, \\ \beta \odot u = 0, \\ \alpha + \beta = p. \end{gathered}$$

进而可以得到对偶变量 α 的梯度为：

$$\frac{\partial L}{\partial \alpha}=(I \odot \theta-\lambda x-G^{\top} \alpha) \frac{\partial x}{\partial \alpha}-(B^{\top} \alpha+\lambda t) \frac{\partial t}{\partial \alpha} \quad+(h-G x-B t)$$

3.2 高效拉格朗日对偶层

Algorithm 1 Forward Pass for Lagrangian Dual Layer (LDL)

Input: θ

Output: $x = LDL(\theta)$

Constants: G, B, h, p

Initialization: $\alpha = 0, mt = 0, vt = 0, \mu = 100, b_1 = 0.9, b_2 = 0.99, e = 10^{-9}, decay_rate = 0.95$

$\lambda = \min(\theta)$

repeat

  Stage 1: Calculate original variable

    $x = \frac{I \odot \theta - G^{\top}\alpha}{\lambda}$

    $t = \frac{-B^{\top}\alpha}{\lambda}$

    $x = \min(0, \max(x, 1))$

    $t = \min(0, \max(t, 1))$

  Stage 2: Calculate dual variable gradient

    $\frac{\partial L}{\partial \alpha} = h - Gx - Bt$

  Stage 3: Update dual variable

    $mt = b_1 * mt + (1 - b_1) * \frac{\partial L}{\partial \alpha}$

    $vt = b_2 * vt + (1 - b_2) * (\frac{\partial L}{\partial \alpha} \odot \frac{\partial L}{\partial \alpha})$

    $\alpha = \min(\max(0, \alpha - \mu * \frac{mt}{\sqrt{vt+e}}, p))$

    $\mu = \mu * decay_rate$

until Convergence or maximum iterations

如 Algorithm1，我们用 3.1 中推导得到的结论来解决库存分配问题。其中在 Stage3，我们使用 Adam 梯度下降法，$b_1, b_2, decay_rate$ 是优化器超参数。该拉格朗日层可以灵活地嵌入任何神经网络结构中并且反向传播可以直接使用 Tensorflow 或 Torch 这种通用框架。优化层的计算并不涉及复杂的操作比如矩阵求逆，并且可以方便地进行分 batch 并行求解。

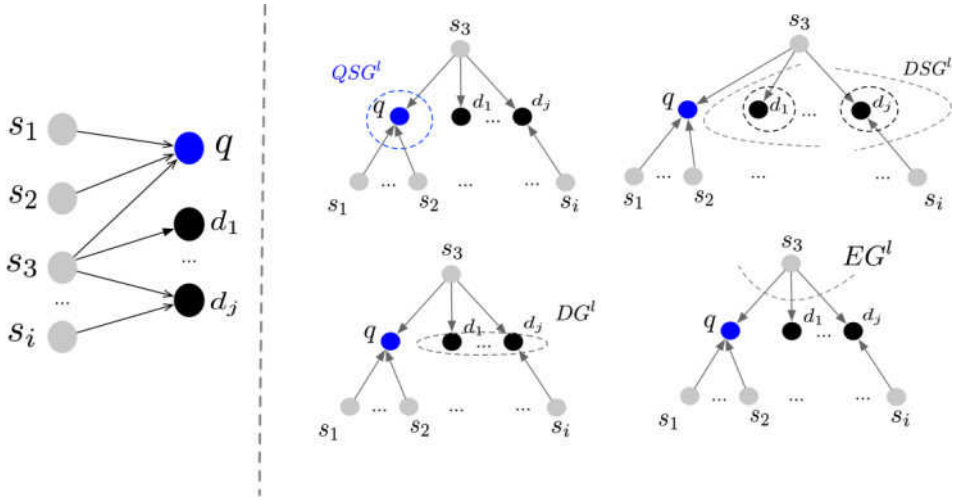
3.3 GCN 模块

为了高效地进行端到端训练，我们需要从合约售卖问题中提取到有用的特征。不同的广告主定向条件不同，因此每次的合约广告售卖都是一个新的问题。我们用设计的图卷积神经网络结构来进行适配。在我们的问题定义里 (问题建模中的二部图)，supply 节点的数量用 m 表示，demand 节点的数量用 n 表示。supply 节点和约束

demand 节点的邻接矩阵用 $A \in \mathbb{R}^{m \times n}$ 表示，supply 节点和决策节点的邻接矩阵用 $I \in \{0,1\}^m$ 表示。GCN 模型的输入特征如下（f_1, f_2, f_3 为特征维度）：

1. Supply 节点特征 $Z^0 \in \mathbb{R}^{m \times f_1}$ 包含节日，日期等信息；
2. Demand 节点特征 $D \in \mathbb{R}^{n \times f_2}$ 包含约束节点的售卖类型，目标量等；
3. Edge 特征 $C \in \mathbb{R}^{m \times n \times f_3}$ 包含边之间的人群竞争，边上分配比例上下限等信息；

由于库存分配二部图的特殊性，已有的 GCN 模型不能直接应用在端到端的训练过程中。



我们设计了新的图特征处理流程，上图展示了 supply 节点 s_3 的特征提取流程。具体地说，每一层特征提取包含以下 5 步，（$W_Z, W^l_{QSG}, W^l_{DSG}, W^l_{DG}, W^l_{EG}$ 是神经网络参数，$l = 1, \cdots, L$ 是层数）：

1. QSG (Query to Supply GCN) 特征：由于我们的目标是学习决策节点的可用库存，我们聚合与决策节点相连的 suplly 节点特征 Z^{l-1}，并且将该特征 broadcast 到所有相关的 supply 节点上：

$$QSG^l = Broadcast(Relu(\frac{I^T Z^{(l-1)} W^l_{QSG}}{\sum_{i=0}^m A_{ij}})).$$

2. DSG (Demand to Supply GCN) 特征：我们首先聚合与约束节点相连的 supply 特征 $\frac{A^T Z^{(l-1)}}{\sum_{i=0}^m A_{ij}}$，再把该 embedding 聚合到 supply 维度：

$$DSG^l = Relu(\frac{A}{\sum_{j=0}^{n} A_{ij}} \frac{A^T Z^{(l-1)}}{\sum_{i=0}^{m} A_{ij}} W_{DSG}^l).$$

3. DG (Demand GCN) 特征：我们将原始 demand 节点特征 D 聚合到 supply 维度 $DG^l = Relu(\frac{ADW_{DG}^l}{\sum_{j=0}^{n} A_{ij}})$.:
4. EG (Edge GCN) 特征：我们聚合边特征到每个 supply 节点，来考虑同一个 supply 出边的竞争（比如频控，人群等因素）：$EG^l = Relu(\frac{\sum_{j=0}^{n}(A_{ij} \cdot C_{ij})}{\sum_{j=0}^{n} A_{ij}} W_{EG}^l)$.
5. 最终，我们聚合 $(QSG^l, DSG^l, DG^l, EG^l)$ 得到：

   $$Z^l = Relu((QSG^l, DSG^l, DG^l, EG^l) \cdot W_Z)$$

   作为下一层 GCN 网络 supply 节点的输入特征。

3.4 端到端优化

Algorithm 2 Forward Pass for End-to-End Allocation Optimization

Input: Z^0, D, C
Output: x
Constant : G, B, h, p, A, I

$Z' = DNN(Z^0)$
for $l = 1, \cdots, L$ **do**
  $QSG^l = Broadcast(Relu(\frac{I^T Z^{(l-1)} W_{QSG}^l}{\sum_{i=0}^{m} A_{ij}}))$
  $DSG^l = Relu(\frac{A}{\sum_{j=0}^{n} A_{ij}} \frac{A^T Z^{(l-1)}}{\sum_{i=0}^{m} A_{ij}} W_{DSG}^l)$
  $DG^l = Relu(\frac{ADW_{DG}^l}{\sum_{j=0}^{n} A_{ij}})$
  $EG^l = Relu(\frac{\sum_{j=0}^{n}(A_{ij} \cdot C_{ij})}{\sum_{j=0}^{n} A_{ij}} W_{EG}^l)$
  $Z^l = Relu((QSG^l, DSG^l, DG^l, EG^l) \cdot W_Z)$
end for
$\hat{\theta} = Relu(MLP(Z^L, Z'))$
$\hat{x} = LDL(\hat{\theta})$

整个端到端算法的前向计算过程如 Algorithm 2 所示，整个流程从全连接网络 DNN 抽取特征 Z' 开始，Z' 会和 GCN 特征 CONCAT 在一起。CONCAT 后的特征经过带有 Relu 优化器的 MLP 层来预估 supply 节点流量 $\hat{\theta}$。$\hat{\theta}$ 输入到拉格朗日对偶层 LDL 得到预估的库存分配结果 $\hat{x}$。最终的分配误差可以被直接优化用 loss:

$$loss = \|(I \odot \hat{\theta})^\top \hat{x} - (I \odot \theta)^\top x\|_2^2 + \epsilon\|\theta - \hat{\theta}\|_2^2.$$

loss 由两部分组成，端到端误差和一阶段误差，超参 ϵ 来平衡端到端误差和第一阶段预估误差。

四、实验

4.1 实验说明

为了评估 NLS 端到端模型的效果，我们在离线和在线广告售卖数据上都做了实验，实验数据的 supplyxdemand 规模以及约束规模如下表：

Table 1: Summary of offline and online data sets.

Dimensions	Offline			Online PVA	Online OSA
	full targeting	single targeting	random targeting	real targeting	real targeting
supply × *demand*	100 × 10	100 × 10	100 × 10	500 × 20	500 × 50
constraints	110	110	110	**10520**	**25550**

4.2 实验结果

4.2.1 baseline

我们用 NLS 模型对比以下 baseline 方案：

1. Two-Stage：对比端到端方案和两阶段方案的效果；
2. PF(Pure Fully-Connected): 端到端模型最朴素的解决方案是采用一个纯全连接网络进行预估，为了验证 NLS 是否真实有效，我们和该基准网络进行对比；
3. PPG(Pure Prediction GCN): 以二部图的建模视角，投前询量问题也可以被直接建模为 GCN 网络进行端到端预估（去掉拉格朗日部分），我们该消融实验验证 Lagrangian solver 是否有用。需要说明的是，即使 GCN 网络的预估结果符合预期，也难以在线上真实使用，因为单纯的神经网络模型是难以保证所有约束都被满足的。
4. PL(Prediction Network + Lagrangian solver): 我们添加 GCN 网络的目的是应对不同复杂的合约售卖环境，我们将 GCN 网络去掉进行消融实验。
5. GCN+QPTL/GCN+InOpt (End-to-End)：QPTL 与 InOpt 是已有的可以

端到端求解 LP/QP 问题的求解器，我们将 baseline 求解器与我们设计的 Lagrange Solver 进行比较，包括求解效率与端到端预估质量。

4.2.2 评估指标

由于不同售卖问题的最优解方差较大，因此我们采用 Normalized Deviation（ND）指标来评估效果，我们用 $E2E_k = (I \odot \theta)^\top x$ 来表示真实的库存分配结果。其中分配比例 x 的真实值来自精确求解器 mindopt，输入为流量观测真值 θ，$\widehat{E2E_k}(= I \odot \hat{\theta})^\top \hat{x}$ 表示我们的预估结果。我们采用以下评估指标：

1. $ND_{reg} = \frac{\sum_{k=0}^{K}\left|E2E_k - \widehat{E2E_k}\right|}{\sum_{k=0}^{K}|E2E_k|}$

   表示端到端结果的整体误差，值越小越优；

2. $ND_{pre} = \frac{\sum_{k=0}^{K}|\theta_k - \hat{\theta_k}|}{\sum_{k=0}^{K}|\theta_k|}$ 表示一阶段流量预估结果的误差；

3. $ND_{allo} = \frac{\sum_{k=0}^{K}\left|E2E_k - \widehat{AE_k}\right|}{\sum_{k=0}^{K}|E2E_k|}$ 表示第二阶段库存分配结果的误差；

4. $ARPD = \frac{\sum_{k=0}^{K}(E2E_k - |E2E_k - \widehat{E2E_k}|) * Price_k}{nn}$表示日均平台收益，nn 为统计天数，库存多卖或少卖都会影响平台收益；

5. $DR_k = \begin{cases} \frac{\widehat{E2E_k} - E2E_k}{\widehat{E2E_k}} & E2E_k < \widehat{E2E_k} \\ 1 & E2E_k \geq \widehat{E2E_k} \end{cases}$

   代表合约完成率；

6. $UR_k = \begin{cases} \frac{E2E_k - \widehat{E2E_k}}{E2E_k} & E2E_k > \widehat{E2E_k} \\ 1 & E2E_k \leq \widehat{E2E_k} \end{cases}$

   代表库存利用率；

4.2.3 结果分析

4.2.3.1 离线数据评估结果

Table 2: Experimental Results on Offline Datasets

Methods	full targeting		single targeting		random targeting	
	ND_{pre}	ND_{reg}	ND_{pre}	ND_{reg}	ND_{pre}	ND_{reg}
Two Stage	$0.101_{\pm 0.003}$	$0.023_{\pm 2e-4}$	$0.101_{\pm 0.003}$	$0.125_{\pm 0.005}$	$0.101_{\pm 0.003}$	$0.045_{\pm 0.002}$
PF	$0.130_{\pm 0.010}$	$0.045_{\pm 0.005}$	$0.115_{\pm 0.008}$	$0.132_{\pm 0.010}$	$0.121_{\pm 0.010}$	$0.076_{\pm 0.007}$
PPG	$0.125_{\pm 0.010}$	$0.015_{\pm 1e-4}$	$0.127_{\pm 0.007}$	$0.112_{\pm 0.001}$	$0.135_{\pm 0.011}$	$0.036_{\pm 0.001}$
PL	$0.102_{\pm 0.001}$	$0.008_{\pm 1e-4}$	$0.103_{\pm 0.001}$	$0.113_{\pm 0.003}$	$0.101_{\pm 0.002}$	$0.047_{\pm 2e-4}$
NLS	$\mathbf{0.096}_{\pm 0.002}$	$\mathbf{0.007}_{\pm 2e-4}$	$\mathbf{0.097}_{\pm 0.001}$	$\mathbf{0.098}_{\pm 0.001}$	$\mathbf{0.095}_{\pm 0.001}$	$\mathbf{0.029}_{\pm 1e-4}$

离线数据评估实验结果如上表所示。两阶段结果值得注意，在 full targeting(定向所有 supply 节点) 的情况下，ND_{reg} 显著低于 ND_{pre}，这表明当合并 supply 节点时，第二阶段的分配任务更容易。在 single targeting(单一定向) 的情况下，整体分配问题与流量预估问题等价，因此 ND_{reg} 略高于 ND_{pre} 接近。随机定向的 ND_{reg} 介于 fulling targeting 和 single targeting 之间。总体而言，**两阶段实验结果展示了端到端学习的潜力**。我们强调几个有趣的实验结果：

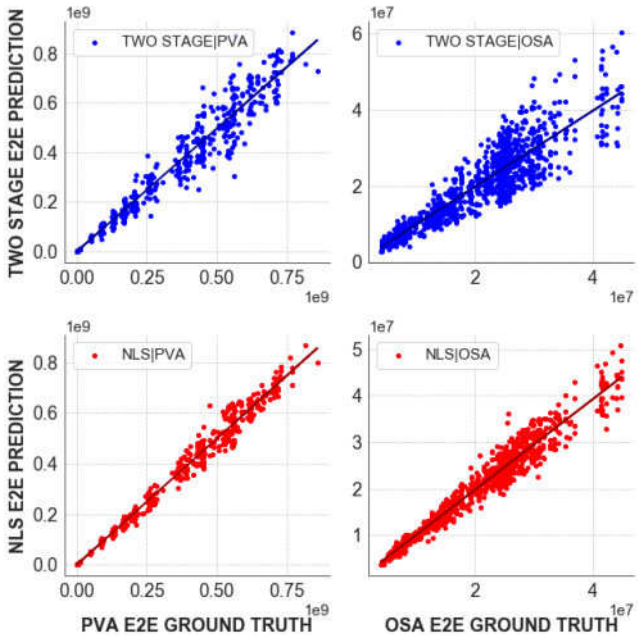
1. PF: 两阶段方法在所有指标上均优于 PF，表明纯黑盒神经网络不足以解决端到端售卖问题。
2. PPG：由于 GCN 可以从二部图中学习信息，因此 PPG 优于 PF。虽然 PPG 的 ND_{pre} 略高于两阶段网络，但其 ND_{reg} 显著降低。然而，PPG 无法在生产环境中部署，因为很难保证分配约束而没有超出。
3. PL: 在 fulling targeting 和 single targeting 情况下，PL 的 ND_{reg} 均低于两阶段方法，但在随机定向情况下，模型在不同训练样本之间振荡，导致随机定向 ND_{reg} 更差。但是，与 PPG 方法相比，PL 的 ND_{reg} 更稳定。
4. NLS: 最后，我们可以观察到 NLS 明显优于两阶段方法和其他基线端到端网络。**改进最显着的是在随机定向情况下的 ND_{reg} 指标，NLS 相对减少了 35.5%**。这突显了 NLS 通过高质量的售卖决策来优化平台收入的有效性。我们可以得出结论，**NLS 结合了 GCN 模块和 Lagrangian 层的优点**。

4.2.3.2 在线数据评估结果

Table 3: Experimental results on two online data sets.

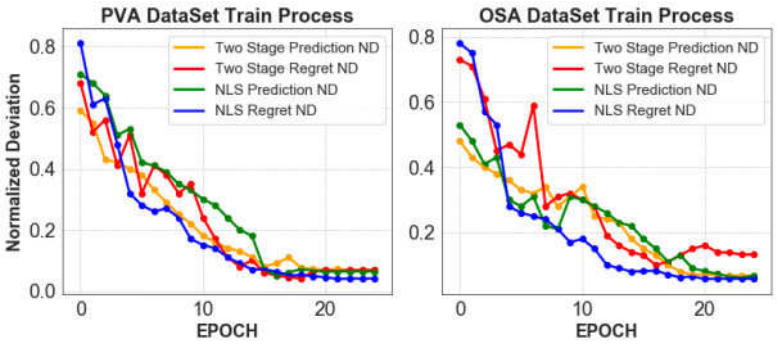
Methods	PVA		OSA	
	ND_{pre}	ND_{reg}	ND_{pre}	ND_{reg}
Two Stage	$0.069_{\pm 0.002}$	$0.068_{\pm 0.004}$	$0.067_{\pm 0.002}$	$0.132_{\pm 0.005}$
PF	$0.083_{\pm 0.015}$	$0.095_{\pm 0.020}$	$0.075_{\pm 0.015}$	$0.128_{\pm 0.021}$
PPG	$0.085_{\pm 0.005}$	$0.054_{\pm 0.002}$	$0.078_{\pm 0.003}$	$0.086_{\pm 0.004}$
PL	$0.065_{\pm 0.002}$	$0.061_{\pm 0.002}$	$\mathbf{0.065}_{\pm 0.001}$	$0.136_{\pm 0.004}$
NLS	$\mathbf{0.064}_{\pm 0.003}$	$\mathbf{0.041}_{\pm 0.001}$	$0.068_{\pm 0.003}$	$\mathbf{0.058}_{\pm 0.001}$

上表的结果表明，NLS 方法在 PVA 和 OSA 数据集上均超过了两阶段方法和其他基线端到端方法。特别是，结果表明，NLS 在降低 PVA 数据集的 ND_{reg} 方面表现出色，降低了 39%，而在 OSA 数据集上降低了 56%。这突显了 NLS 在处理具有复杂和动态约束的数据集的优势。PF 仍然表现最差，而相比于 PVA 数据，PPG 在 OSA 数据上表现更好，这表明更大的售卖问题具有更多的优化潜力。另一方面，PL 方法没有处理复杂约束的能力。



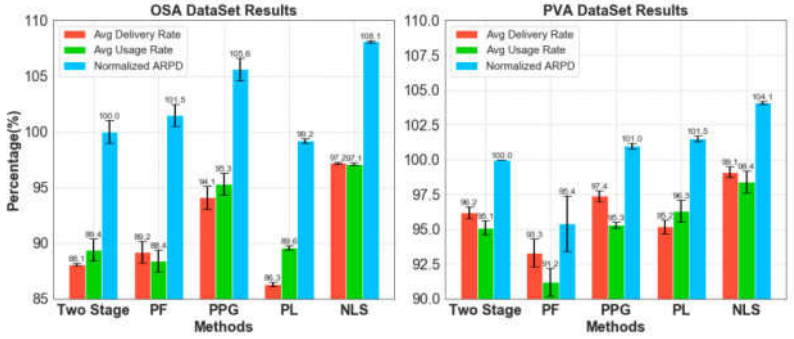
此外，上图散点图显示了 NLS 和两阶段方法在 PVA 和 OSA 数据集上的 $\widehat{E2E}$ 结

果。结果表明，与两阶段方法相比，NLS 的离群值较少，因为 NLS 没有像两阶段方法那样过度拟合第一阶段的预测任务。



上图展示了 NLS 和两阶段方法的训练过程，支持了端到端方案更优的理论。在 OSA 数据集上，16 个 epoch 之后，尽管第一阶段的预测误差 ND_{pre} 仍在优化，但两阶段方法的 ND_{reg} 在增加。

4.2.3.3 收入分析



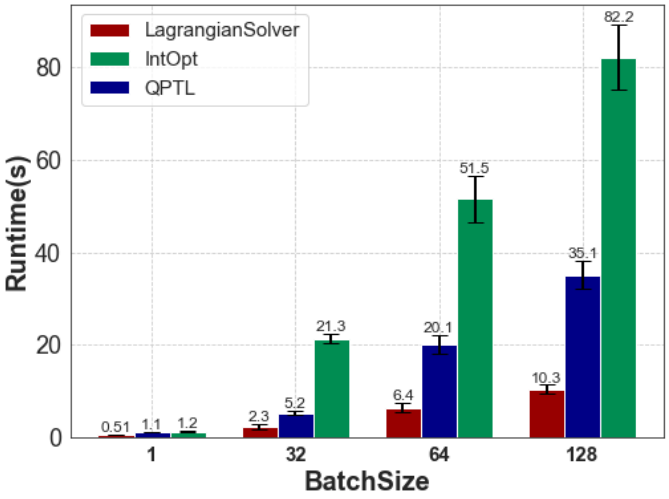
上图比较了不同方法对平台收入的影响，以两阶段 ARPD 为基线，**NLS 在 OSA 上实现了 8.1% 的收入增长，在 PVA 上实现了 4.1% 的增长**。它还在合约完成率和库存利用率方面始终优于其他模型，在 OSA 上实现了 97.1% 的使用率和 97.2% 的交付率，在 PVA 上实现了 98.4% 的使用率和 99.1% 的交付率。

4.2.3.4 Lagrangian Layer vs QPTL vs IntOpt

Table 4: Comparison of NLS with other LP/QP solvers. Our NLS can handle complex constraints and achieve improved solutions.

Methods	Offline (random)			PVA			OSA		
	ND_{pre}	ND_{allo}	ND_{reg}	ND_{pre}	ND_{allo}	ND_{reg}	ND_{pre}	ND_{allo}	ND_{reg}
GCN + QPTL	$0.098_{\pm 0.008}$	$0.15_{\pm 0.02}$	$0.032_{\pm 0.005}$	-	-	-	-	-	-
GCN + IntOpt	$0.100_{\pm 0.003}$	$0.25_{\pm 0.03}$	$0.038_{\pm 0.005}$	$0.068_{\pm 0.006}$	$0.32_{\pm 0.04}$	$0.063_{\pm 0.002}$	$0.070_{\pm 0.008}$	$0.33_{\pm 0.04}$	$0.083_{\pm 0.006}$
NLS	$\mathbf{0.095}_{\pm 0.001}$	$\mathbf{1e\text{-}4}_{\pm 1e-5}$	$\mathbf{0.029}_{\pm 1e-4}$	$\mathbf{0.064}_{\pm 0.003}$	$\mathbf{7e\text{-}4}_{\pm 1e-5}$	$\mathbf{0.041}_{\pm 0.0031}$	$\mathbf{0.068}_{\pm 0.003}$	$\mathbf{6e\text{-}4}_{\pm 1e-5}$	$\mathbf{0.058}_{\pm 0.001}$

首先，我们评估了三个求解器的第二阶段准确性 ND_{allo}，如上表所示，以精确 LP 求解器 mindopt 计算出的实际值作为基准，**Lagrangian 求解器的误差几乎为零**，而 QPTL / IntOpt 的误差相对较大。这可能是因为 Lagrangian 求解器专门为库存分配问题设计和优化。此外，由于约束数量与离线数据集相比增加，QPTL 无法解决实际的库存分配问题，因为其内存超限问题很严重。接下来，我们比较三个求解器在端到端销售问题中的性能，如表所示，**NLS 优于两个基线求解器**。但令人惊讶的是，在离线数据上，尽管 QPTL / IntOpt 的分配误差虽然显著，它们的端到端性能相对没有很差。我们认为原因是，尽管 QPTL / IntOpt 的前向传播分配结果不够准确，但其与最佳解相比的优化轨迹（梯度下降方向）相对正确。在两个在线数据集上，我们的端到端求解器的优越性更加明显，能够更好地解决 GD 销售问题。最后，比较了每个求解器的批次时间消耗，并在下图中显示了结果。使用单个 Tesla p100 GPU 进行性能比较，我们的 Lagrangian 求解器明显优于 QPTL / IntOpt。



五、结论

本文聚焦合约广告售卖问题，提出并验证了端到端方法相比于传统的两阶段方法的优越性。具体而言，Neural Lagrangian Selling (NLS) 模型可以学习复杂和动态的合约广告售卖问题，并通过 Lagrangian 对偶层和 GCN 模块进行端到端的学习。在离线和在线生产数据集上的实验表明，与两阶段方法相比，NLS 提高了合约完成率、库存利用率进而增加了平台的收入。此外，相比基准求解器，拉格朗日求解器具有更高的求解效率和更低的求解误差。我们对合约广告售卖问题的端到端预测优化任务进行了较为全面的探讨分析。

参考文献

[1] Priya Donti, Brandon Amos, and J Zico Kolter. 2017. Task-based end-to-end model learning in stochastic optimization. Advances in neural information processing systems 30 (2017).

[2] Zhen Fang, Yang Li, Chuanren Liu, Wenxiang Zhu, Yu Zheng, and Wenjun Zhou.2019. Large-scale personalized delivery for guaranteed display advertising with real-time pacing. In 2019 IEEE International Conference on Data Mining (ICDM).IEEE, 190 – 199.

[3] Ali Ugur Guler, Emir Demirović, Jeffrey Chan, James Bailey, Christopher Leckie, and Peter J Stuckey. 2022. A divide and conquer algorithm for predict+ optimize with non-convex problems. In Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 36. 3749 – 3757.

[4] Jayanta Mandi and Tias Guns. 2020. Interior point solving for lp-based prediction+optimisation. Advances in Neural Information Processing Systems 33 (2020), 7272 – 7282

[5] Bryan Wilder, Bistra Dilkina, and Milind Tambe. 2019. Melding the data-decisions pipeline: Decision-focused learning for combinatorial optimization. In Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 33. 1658 – 1665.

[6] Zonghan Wu, Shirui Pan, Fengwen Chen, Guodong Long, Chengqi Zhang, and S Yu Philip. 2020. A comprehensive survey on graph neural networks. IEEE transactions on neural networks and learning systems 32, 1 (2020), 4 – 24.

[7] Hong Zhang, Lan Zhang, Lan Xu, Xiaoyang Ma, Zhengtao Wu, Cong Tang, Wei Xu, and Yiguo Yang. 2020. A request-level guaranteed delivery advertising planning: Forecasting and allocation. In Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. 2980 – 2988.

强化学习在广告延迟曝光情形下的保量策略中的应用

作者：聆涛、时玮

本文分享阿里妈妈外投广告 UD 效果 & 用增算法团队针对广告延迟曝光问题通过强化学习 pacing 策略来完成曝光保量等各类业务目标的算法实践，相关技术方案已总结为论文发表在 KDD 2023，欢迎阅读交流。

论文：RLTP: Reinforcement Learning to Pace for Delayed Impression Modeling in Preloaded Ads

链接（点击↓阅读原文）: https://arxiv.org/pdf/2302.02592.pdf

1. 背景

预加载是开屏广告中的一种常见策略：在当前流量请求中，曝光的广告是过往请求中所填充的，而本次填充的广告在后续的某次流量请求中才会曝光。媒体侧将流量请求发送给阿里侧 DSP 来询问是否选择本条流量，我们需要在完整投放周期后满足广告主对曝光量、点击率等目标的诉求。由于预加载，我们只能决定流量的选择与否，但最终曝光与否由媒体策略决定。预加载策略引出的延迟曝光现象给保量投放带来挑战主要包括：当前观测的曝光量实际上是不完整的，还有部分已选择流量会在未来曝光，因此当前无法得知真实曝光量。如不考虑潜在曝光量进行 pacing 调控，可能会导致超投影响收益；如果过度考虑，则可能会造成缺量影响效果。

预加载广告的两个特点启发我们通过强化学习解决：首先是反馈信号延迟性，即只有当投放结束后才能准确观测是否完成广告主各类目标；其次是媒体预加载策略对我们黑盒，很难精确模拟。强化学习的目标是最大化长期奖励，通过与环境交互来改进 policy，具备应对上面两个特点的可能性。我们提出曝光保量框架 RLTP: Reinforcement Learning to Pace，学习一个 pacing agent 来直接输出流量选择概率，将保量策略从多阶段“曝光预测模型 + 人工规则 +PID 调节”简化至端到端的 pacing agent。为满足广告主对于曝光量、点击率等方面的要求，设计 reward estimator 来鼓励接近预设曝光量、选择高价值流量、惩罚超量及选择概率剧烈变化。实验验证了端到端 pacing agent 完全有替代多阶段框架来完成曝光保量的能力。

2. 引言

本文针对预加载广告的曝光保量策略展开研究。当用户打开媒体 App 时，媒体侧将流量请求发送给阿里侧 DSP（需求方平台）来询问是否选择本条流量，我们预估流量价值后做出选或不选的决策，如果选择会填充我方的广告并返回给媒体，总体目标是在完整的投放周期结束后满足广告主对于曝光量、点击率等目标的诉求。



由于预加载的存在，我们只能决定流量的**选择**与否，但最终广告**曝光**与否是由媒体策略决定的，我方不能感知媒体策略。因此，预加载策略引出的**延迟曝光**现象给以保量为主要诉求的投放活动带来一定的算法挑战。

- 假设我们选择流量并填充广告后会立即在媒体曝光，那么可以通过 PID Controller 等 pacing 算法，根据实时的观测曝光量来每过一定间隔（通常是 5min）调节初始的流量选择概率，完成保量的目标；同时可以根据对请求按照预估 CTR 分层，建立多组 PID 系数来服务于不同层的流量，完成点击率的目标 [1]。
- 但当存在延迟曝光现象时，当前观测到的曝光量实际上是不完整的，还有部分已经选择的流量将会在未来产生曝光，因此真实曝光量在当前无法得知。如果不考虑这部分潜在曝光量来进行 pacing 调控，最终会超投影响利润；如果过度考虑了这些量，又会造成缺量，影响广告主后续和我们的合作。

综上，针对延迟曝光，需要设计特定的算法调控方式来满足广告主诉求。一种很实用的方案是根据历史投放数据学习一个模型，基于当前 context 预测潜在曝光量（历史

数据中可获取每一个曝光对应的广告填充时刻），从而在 PID 调控的时候认为“观测曝光量 + 预测的潜在曝光量”是传感器信号，对流量选择概率进行调节。但是会存在：1）潜在曝光量预测模型存在固有的预测偏差；2）PID Controller 需要事先指定每个时段的目标曝光量作为 target，这依赖于历史数据上的经验和人工规则，同时有很多超参数需要设置，对 CTR 分层后超参数数量成倍增多；3）预测模型和 PID Controller 是相互割裂的，二者的不适配会使得广告主的各项目标不能被很好地满足。

预加载广告的两个特点启发我们通过 Reinforcement Learning 解决。首先是反馈信号的延迟性，即只有当投放周期结束后才能准确观测是否完成了广告主各类目标，而投放期间不能实时获取准确的完成度；其次是媒体预加载策略对我们完全黑盒，很难精确地模拟。RL 的目标是最大化长期奖励而非即刻奖励、并且通过与环境不断交互来改进 policy，具备应对上面两个特点的可能性。因此我们基于 RL 思想，学习一个 pacing agent 来直接输出流量选择概率，摒弃此前通过观测信息以及人工规则来调整初始概率的做法，将整个保量策略从多阶段“曝光预测模型 + 人工规则 +PID 调节”简化至完全**端到端的 pacing agent**。

针对存在延迟曝光情况下的保量诉求，我们提出曝光保量框架 RLTP：Reinforcement Learning to Pace，**将投放周期内的流量选择建模为序列决策问题，序列化地输出流量选择概率**。为了满足广告主对于曝光量、点击率等方面的要求，针对性地设计了 reward estimator 来鼓励接近预设曝光量及选择高价值流量、惩罚超量及选择概率剧烈变化。为了完成 RL 模型训练和评测，我们构造了一个离线模拟器，将 baseline 和 RL agent 在统一的环境下对比效果，实验验证了端到端 pacing agent 完全有替代多阶段框架来完成曝光保量的能力。

3. 问题形式化

3.1 投放目标

对于一个投放周期，记 N_{target} 是广告主设定的期望曝光量。假设流量请求量 N_{req} 是充足的 (但未知)，设计 pacing 算法使投放结束后达成以下两个目标：

- **曝光量**：最终的曝光量 N_{imp} 不能超投 $N_{\text{imp}} \leq N_{\text{target}} + \epsilon$、且越接近越好

$$
\begin{aligned}
\min \quad & N_{\text{target}} - N_{\text{imp}} && \text{; The closer the better.} \\
\text{s.t.} \quad & N_{\text{imp}} - N_{\text{target}} \leq \epsilon && \text{; Avoid over-delivery.} \\
& \epsilon \geq 0
\end{aligned}
$$

其中 $\epsilon \geq 0$ 是一个可容忍的超量阈值，一般设置为较小值。另外，在投放过程中也应尽可能满足曝光量是**平滑**释放的，而非过于集中在某一时段，从而触达更多时段的活跃人群。

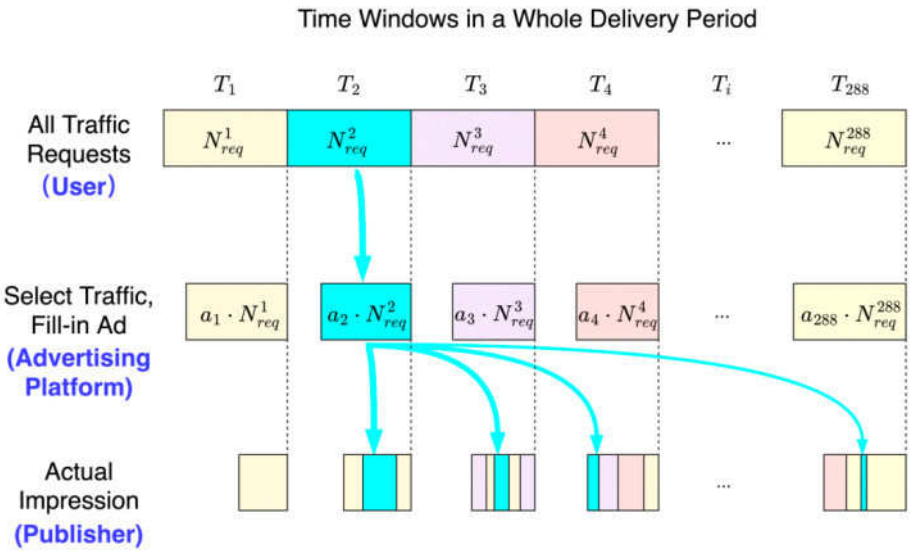
- **投放效果**：本工作中以一跳点击率作为效果衡量指标，希望挑选出的流量的价值尽可能高。

3.2 延迟曝光

下图给出了延迟曝光情形的简要示意。对于完整投放周期 (e.g., 一天)，以固定时间间隔 (e.g., 5min) 划分成 L 个窗口 $\{T_1, T_2, \ldots, T_L\}$。在每个窗口 T_i，记请求量为 N^i_{req}、pacing 算法给出的流量选择概率为：$a_i \in [0, 1]$，那么该窗口内选择的请求量为 $a_i \cdot N^i_{\text{req}}$。由于媒体侧预加载策略的存在，真实曝光量需要等到整个投放周期结束后才可观测到。形式化地，窗口 T_i 填充的广告对应的真实曝光量“分散”在窗口 T_i 至窗口 T_L：

$$N^i_{\text{imp}} = \beta_i N^i_{\text{imp}} + \beta_{i+1} N^i_{\text{imp}} + \ldots \beta_L N^i_{\text{imp}}, \quad \text{where} \quad \sum_{j=i}^{L} \beta_j = 1$$

其中 N^i_{imp} 表示真实曝光量，β_j 是窗口 T_j 的曝光量占真实曝光量的延迟系数，由媒体预加载策略决定。因此在窗口 T_i 结束后只能观测到曝光量 $\beta_i N^i_{\text{imp}}$，给保量带来了比较大的挑战。



3.3 建模为 MDP

真实曝光量只有投放结束后才能观测到、以及延迟系数很难模拟，这两个特性启发我们使用 RL 来构建 pacing 算法。具体来说，我们将投放周期内的流量选择问题建模为序列决策，序列化地在每个窗口的起始时刻输出流量选择概率，对应的 MDP 五元组 $(\mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}, \gamma)$ 定义如下：

- $\mathcal{S}$ 是状态空间。对于窗口 T_i，状态 $\boldsymbol{s}_i$ 需要反映窗口 $\{T_1, \dots, T_{i-1}\}$ 结束后的投放状态。
- $\mathcal{A}$ 是动作空间。对于窗口 T_i，动作 a_i 即为流量选择概率。
- $\mathcal{P} : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to [0, 1]$ 是状态转移函数，$p(\boldsymbol{s}' \mid \boldsymbol{s}, a)$ 描述了从状态 $\boldsymbol{s}$ 执行动作 a 后转移到状态 $\boldsymbol{s}'$ 的概率。
- $\mathcal{R} : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ 是奖励函数，$r(\boldsymbol{s}, a)$ 描述了在状态 $\boldsymbol{s}$ 执行动作 a 后收到的奖励值，它的设计是本文算法的关键。
- $\gamma \in [0, 1]$ 是折扣系数，平衡即刻奖励与长期奖励。

在上述 MDP 下，我们的目标就是通过最大化期望累积奖励 $\max_\pi \mathbb{E}\left[\sum_{i=1}^{L} r(\boldsymbol{s}_i, a_i)\right]$ 来学习一个 pacing policy $\pi : \mathcal{S} \to \mathcal{A}$，完成广告主的各类诉求。

4. RLTP: Reinforcement Learning to Pace

本节介绍我们提出的 RLTP 框架，包括网络结构以及 state、action、reward 的设计。记经过 L 步决策后形成的轨迹为：$\tau = (\boldsymbol{s}_1, a_1, r_1, \dots, \boldsymbol{s}_L, a_L, r_L)$。

4.1 State Representation

对于窗口 T_i，状态 $\boldsymbol{s}_i$ 的设计是要体现窗口 $\{T_1, \dots, T_{i-1}\}$ 结束后的投放状态。具体来说我们主要是使用统计类特征、context 特征和用户 / 广告特征：

- **统计类特征**：观测曝光量序列 $\{N_1, \dots, N_{i-1}\}$、点击数序列 $\{C_1, \dots, C_{i-1}\}$，以及由这两个值所导出的相关统计量，例如累计的观测曝光量 $N_{1:i-1} = \sum_{t=1}^{i-1} N_t$，观测的曝光量完成率 $D_{i-1} = N_{1:i-1} / N_{\text{target}}$，观测完成率序列 $\{D_1, \dots, D_{i-1}\}$，相邻两窗口间的观测完成率 diff $\Delta_{i-1} = (N_{1:i-1} - N_{1:i-2}) / N_{\text{target}}$，平均点击率 $\overline{CTR}_{1:i-1} = \sum_{t=1}^{i-1} C_t / \sum_{t=1}^{i-1} N_t$ 等等，此外我们也考虑平均选择概率 $\overline{a}_{1:i-1} = \frac{1}{i-1} \sum_{t=1}^{i-1} a_t$。
- **context 特征**：描述流量请求信息，包括 week、hour、minute 时间信息等。

- **用户 / 广告特征**：描述此前窗口的曝光用户信息和相应的广告信息。由于这部分特征太稀疏，所以直接访问 pretrained embeddings，pacing policy 训练时不更新。

4.2 Action Space

实践中我们采取的是离散动作空间：以 0.02 为步长，流量选择概率 a_i 属于离散动作空间 $\mathcal{A} = \{0, 0.02, 0.04, \ldots, 0.98, 1.0\}$。连续动作空间则对应于流量选择概率 $a_i \in [0, 1]$。后续实验里对比了这两种方式。

4.3 Reward Estimator

Reward estimator 是 RLTP 框架中最关键的部分，它直接引导着 pacing agent 的学习。经过无数失败的尝试，总结出了如下四项奖励：

- **鼓励保量**：如果最终的曝光量没有达到目标，仍会收到未超投所带来的奖励。

$$r_i^{(1)} = \begin{cases} \exp(N_{1:i}/N_{\text{target}}), & \text{if} \quad N_{1:i} \leq N_{\text{target}} + \epsilon \\ 0, & \text{otherwise} \end{cases}$$

- **惩罚超量**：如果超投，置一个很大的负奖励

$$r_i^{(2)} = \begin{cases} 1 - \exp\left(N_{1:i}/N_{\text{target}}\right)^2, & \text{if} \quad N_{1:i} > N_{\text{target}} + \epsilon \\ 0, & \text{otherwise} \end{cases}$$

- **鼓励匀速**：相邻窗口之间的流量选择概率没有剧烈的变化时给一个小奖励

$$r_i^{(3)} = \begin{cases} +1, & \text{if} \quad \left| \dfrac{N_{1:i} - N_{1:i-1}}{N_{1:i-1}} \right| < \mathrm{c} \\ 0, & \text{otherwise} \end{cases}$$

- **最大化点击**：设置一个点击率基线值，希望挑选出来的流量能够超越这个值

$$r_i^{(4)} = \exp\left(100 \cdot \left(\frac{C_{1:i}}{N_{1:i}} - CTR_{\text{base}}\right)\right)$$

上述四项奖励以加权和的形式作为窗口 T_i 结束后的奖励：

$$r_i = \eta_1 \cdot r_i^{(1)} + \eta_2 \cdot r_i^{(2)} + \eta_3 \cdot r_i^{(3)} + \eta_4 \cdot r_i^{(4)}$$

由于每一项的尺度不同，实践中我们参考从 HER 中衍生出来的一种奖励融合方法 [2]，将每项奖励的系数 $\boldsymbol{\eta} = (\eta_1, \eta_2, \eta_3, \eta_4)^\top$ 作为可学习的参数向量，送入到神经网络中。

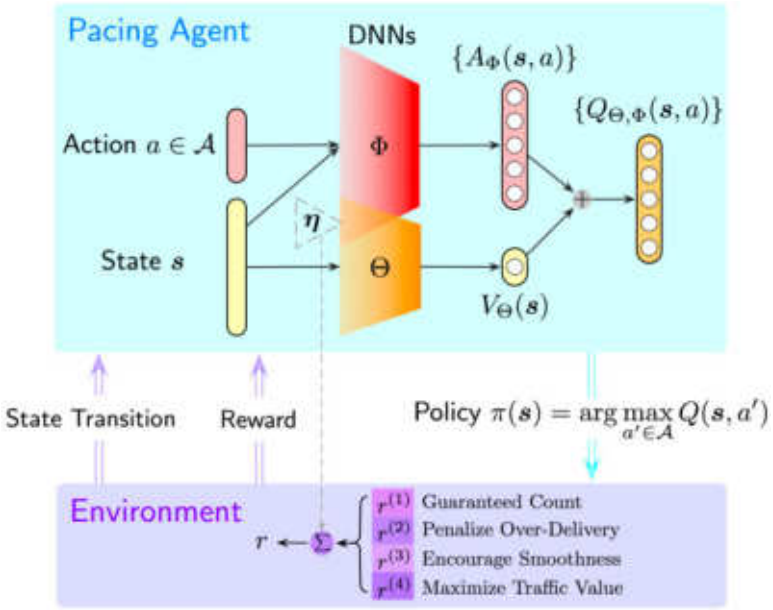
4.4 网络结构

网络结构使用常规的 Dueling DQN，通过两个神经网络 $V_\Theta(\cdot)$ 和 $A_\Phi(\cdot)$ 来逼近值函数 $V^\pi(\boldsymbol{s})$ 和优势函数 $A^\pi(\boldsymbol{s},a)$ 以应对巨大空间的 state-action pairs。Q 函数和对应的 policy 为：

$$Q_{\Theta,\Phi}(\boldsymbol{s},a)=V_\Theta(\boldsymbol{s})+\left(A_\Phi(\boldsymbol{s},a)-\frac{1}{|\mathcal{A}|}\sum_{a'\in\mathcal{A}}A_\Phi(\boldsymbol{s},a')\right)$$
$$\pi(\boldsymbol{s})=\arg\max_{a'\in\mathcal{A}}Q_{\Theta,\Phi}(\boldsymbol{s},a')$$

在考虑奖励系数后，神经网络的形式如下：$V_\Theta(\boldsymbol{s},\boldsymbol{\eta})=\mathsf{DNN}(\boldsymbol{s},\boldsymbol{\eta};\Theta)$，$A_\Phi(\boldsymbol{s},a,\boldsymbol{\eta})=\mathsf{DNN}(\boldsymbol{s},a,\boldsymbol{\eta};\Phi)$ 为了训练和评估 policy，我们通过构造的离线模拟器来实现（细节见实验章节）。RLTP 的 pacing agent 与模拟器不断交互来得到训练数据，优化目标为：

$$\min_{\Theta,\Phi}\sum_{(s_i,a_i,r_i,s_{i+1})}\left(r_i+\gamma\max_{a'\in\mathcal{A}}Q_{\Theta,\Phi}(s_{i+1},a')-Q_{\Theta,\Phi}(s_i,a_i)\right)^2$$



5. 实验

5.1 实验设置

1）离线模拟器

我们从历史投放的线上日志中取了一周的数据并进行了数据增强，用于构造离线模拟器。每个投放周期为一天、窗口时间间隔为 5min，也就是说一个投放周期内进行 288 步决策。根据我们的状态和奖励定义，都是对观测曝光量 N_* 和点击量 C_* 的各种组合，所以我们简化了模拟器的构造方式，只需对历史日志中的 $N_{i-1}, C_{i-1}, a_i \rightarrow N_i, C_i$ 这一映射关系进行拟合，具体是通过训练一个 multi-head DNN 来实现的，两个 head 分别输出 N_i 和 C_i。基于该映射的模拟器，pacing agent 可以与之交互来收集训练数据，同时也可以评估 agent 的效用。

2）评价指标

针对延迟曝光情形下的保量 pacing 算法主要通过以下指标来评价，baselines 和 RL agent 统一在模拟器下对比效果：

- **投放完成率**：投放周期结束后的真实曝光量与广告主期望曝光量的比值，衡量是否完成了保量目标
- **点击率**：投放周期结束后的点击量与真实曝光量的比值，衡量投放效果
- **累积奖励**：这是一个反映 pacing 算法效用的间接指标

3）Baselines

- **统计规则 + 截断**：从历史数据中统计一张从观测曝光量到实际曝光量的映射表，每个窗口进行查表，一旦查到的值接近目标曝光量就直接停止选择流量
- **预测潜在曝光量 +PID**：先从历史数据中学习一个预测潜在曝光量的模型，然后以 PID 进行概率调节
- **RLTP-continuous**：本文所提出的端到端方法，基于连续动作空间。网络结构基于 PPO，没有奖励系数学习
- **RLTP**：本文所提出的端到端方法，基于离散动作空间

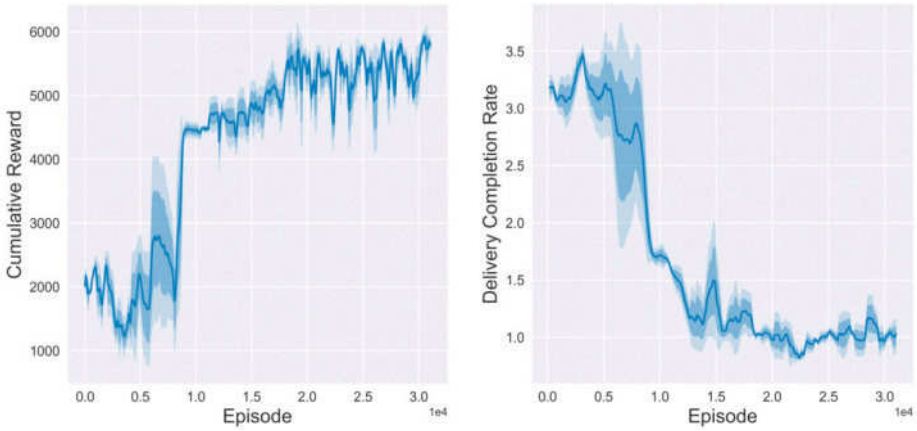
5.2 主实验

四个方法的效果对比见下表。可以看到除了两个 RL agent 以外，其他方法都出现了比较严重的超投现象，不能够较好地处理延迟曝光带来的真实曝光量在观测上的滞后

性。整体上来说，RLTP−continuous 与 RLTP 的性能接近。

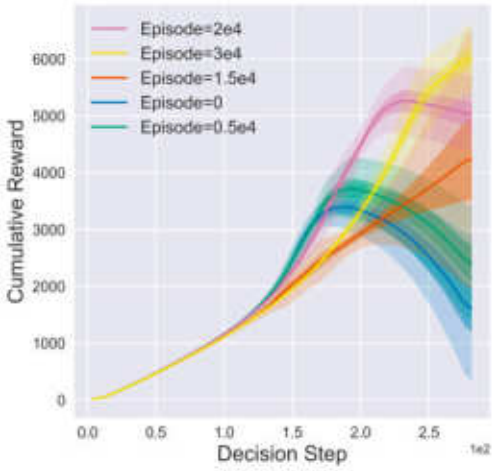
Methods	Completion Rate	CTR Uplift	Cumulative Reward
Stats. Rule-based Adjust.	151.8%	-	3984.5
Pred.-then-Adjust.	120.3%	+0.35%	5637.3
RLTP-continuous	116.5%	+3.18%	5750.1
RLTP	**107.7%**	**+4.01%**	**6080.2**

RLTP 大约用了 3w episodes 训练后可以得到较稳定的累积奖励和投放完成率：



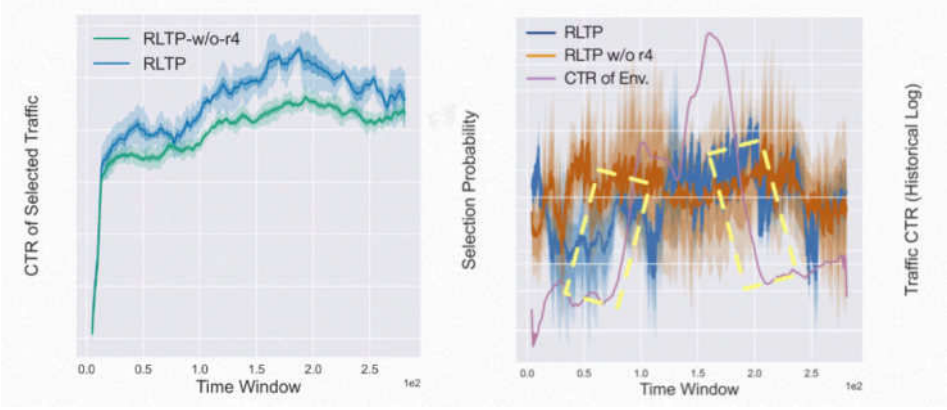
5.3 保量能力

下图展开了 RLTP 在训练不同的 episode 数时，pacing agent 在每一步决策后收到的累积奖励情况（共 288 步）。当 episode 较小时，经过约 200 步之后会超量，此时由于惩罚超投的负值奖励，累积奖励急剧下降；随着 episode 增大，超量情况不再出现，累积奖励呈现上涨趋势。因此 RLTP 在经过足够 episode 训练后，pacing agent 基本掌握了保量能力。



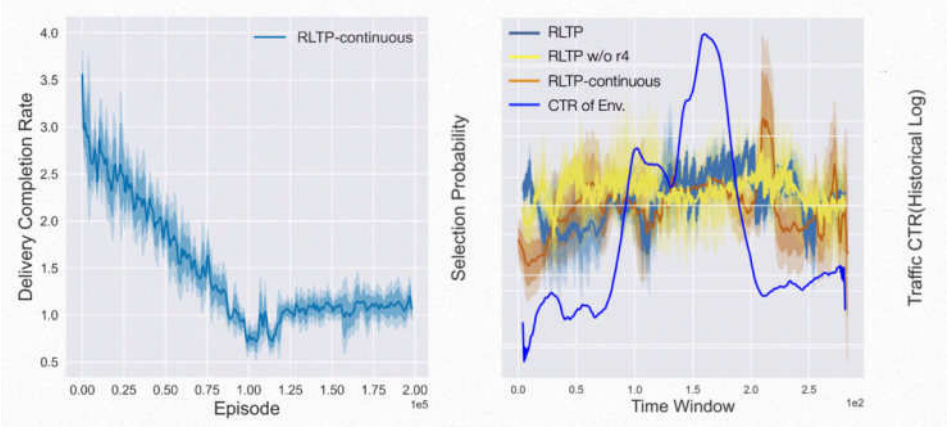
5.4 点击率比较

考察关于最大化点击的奖励项 $r^{(4)}$ 的作用。左图对比了挑选流量的点击率，表明在训练时去掉该项后，pacing agent 挑选的流量的点击率的确会有所下降。进一步考察 agent 输出的流量选择概率在每步决策中的变化趋势：我们希望在点击率高的时段 agent 能够增大流量选择概率来拿到更多高价值流量。右图展示了 agent 在 288 步决策过程中输出的流量选择概率与模拟器上的后验点击率的关系，可以发现在引入奖励项 $r^{(4)}$ 后，输出概率的大小随后验点击率的一致性更强。具体来说，在两处虚线黄框部分，后验点击率分别是增加和下降趋势，RLTP（蓝线）与之趋势具有一致性，而去掉奖励项 $r^{(4)}$ 后（橘线）则对该趋势不甚敏感，因此 RLTP 一定程度上具备在高点击率时段选择更多流量的能力。



附：离散动作空间 vs 连续动作空间

我们也对比了离散连续空间和连续动作空间。左图是 RLTP-continuous 的训练情况，大约用了 20w 个 episodes 得到了稳定的完成率。右图表示其 agent 输出的流量选择概率随后验点击率的关系，可以发现一致性相比 RLTP 略有差距，但整体上好于 RLTP 去掉奖励项 $r^{(4)}$。



6. 总结

针对预加载广告存在延迟曝光情形下的保量 pacing 策略设计，早先更多是依赖经验设计很多人工规则进行截断、或者基于多阶段式的预测 + 调控来解决。本次我们尝试绕过繁复的业务规则而直接对流量选择概率建模，提出了基于 RL 的框架 RLTP，实验效果验证了端到端的 pacing 框架完全有能力替代此前方案来完成广告主对于曝光保量等方面的诉求。

References

[1] Smart pacing for effective online ad campaign optimization. KDD 2015
[2] Generalizing Across Multi-Objective Re- ward Functions in Deep Reinforcement Learning. 2018
[3] Dueling DQN: Dueling Network Architectures for Deep Reinforcement Learning. ICML 2016

MiRO：面向对抗环境下约束竞价的策略优化框架

作者：筱哲

本文分享阿里妈妈外投算法团队在黑盒对抗环境下的约束竞价问题上的探索。该工作已经发表在 KDD2023，欢迎阅读交流。

论 文：Adversarial Constrained Bidding via Minimax Regret Optimization with Causality-Aware Reinforcement Learning

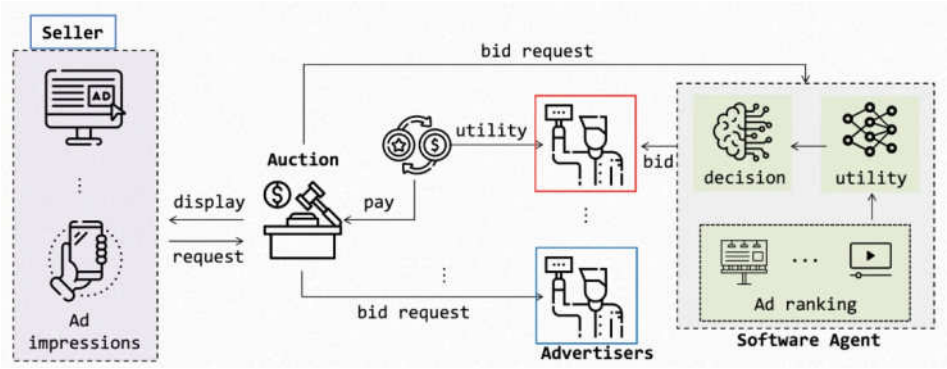
下载（点击↓阅读原文）：https://arxiv.org/abs/2306.07106

1. 引言

展示广告是在线广告的主要营销渠道之一。广告主通过 DSP 提供的程序化广告竞价服务来实现商家各自金融约束下的营销目标优化。近年来，约束下广告竞价算法已经逐步从反馈控制升级为基于强化学习的方法，取得了显著的升级。现有的强化学习方法往往基于经验风险最小化 (ERM) 原则，假设训练与测试条件独立同分布。然而，这个假设在展示外投的复杂博弈竞争环境中存在较大的局限性。

在外投场景下，广告系统存在多个利益方进行博弈，包括我方广告主、其他参竞广告主和媒体。所有参与方都试图优化自己的利益，这些目标可能彼此冲突。例如，媒体可以从数据中学习参竞者的私有价值分布，在拍卖中设定个性化的保留价格 [1]，甚至可以直接端到端学习拍卖机制 [2]。此外，其他参竞广告主往往采用复杂且未知的竞价策略来优化其利益，这导致我方策略面对的成本分布动态变化，从而影响我方算法的表现。这些案例反映了外投场景下竞价环境的对抗性。

对抗性环境中的广告竞价问题尚未得到充分探索。目前只有一些研究 [3,4] 针对无约束竞价和已知拍卖机制的对抗竞价问题取得了一些进展。然而，在展示外投场景中，广告主关心投产比约束，同时我们无法确定媒体实际采用的拍卖机制以及竞对方采用的竞价策略。因此，本文旨在解决黑盒对抗环境下的约束竞价问题，该问题不显式地假设环境中的对抗性外部因素。在博弈视角下，相当于竞价环境中存在一个黑盒对抗者，基于其对参竞者的了解按未知的目标扰动竞价环境，例如改变市场动态或价值分布，使得非适应性的竞价策略表现退化。



展示广告系统（其中媒体、我方广告主、竞对广告主之间存在利益博弈）

针对这个问题，阿里妈妈外投算法团队进行了初步探索，相关工作已经发表在KDD2023。由于篇幅限制，本文将着重介绍该方案背后的主要设计思路。

2. 方法

本文聚焦的竞价设定是包含投产比约束的约束竞价（ROI-Constrained Bidding），它可以方便地推广为广告主所关心的多种营销目标。我们延续近期的工作，也采用了基于强化学习的建模方式。为简便起见，我们在此省略了问题形式化和 MDP 建模的细节，感兴趣的读者可以移步论文。现有的基于强化学习的约束竞价方法通常是基于过去的竞价日志来构建模拟的训练环境。将竞价策略记为，环境记为，其优化目标为最大化训练环境中策略累积收益的期望。

$$\max_{\pi} \mathbb{E}_{\mathcal{M}} [V(\pi; \mathcal{M})],$$

其隐含的假设是训练环境与测试环境的分布是独立同分布的。

然而，由于展示外投场景面对各利益方的博弈，基于训练环境均匀训练的策略在线上面对的环境往往呈现出对抗性。此外，媒体机制和竞对策略的黑盒性质使得我们对于环境的对抗性没有可用的先验知识。针对这样黑盒对抗条件下的约束竞价问题，我们的想法是，不依赖训练测试独立同分布的假设，而是诉诸训练测试对齐。为此，我们必须首先回答以下问题：在对抗性环境的设定下，我们应该如何假设测试分布的性质？给定该性质，我们如何对齐训练和测试分布？

2.1 极小极大遗憾优化（Minimax Regret Optimization）

我们首先考虑严格的对抗设定。在博弈视角下，当对手完全了解我们的竞价策略，那么它可以将环境扰动为该策略表现最差的环境。基于这样一种 insight，我们假设在对抗条件下，某种测试环境出现的概率与策略在该环境中表现有多差成正比。策略在该环境中的表现越差，则认为该环境出现的概率越大。由此刻画出对抗设定下，测试环境的分布。

为了对这一想法进行形式化，我们首先介绍策略的性能度量。由于在对抗条件下，不存在一种稳态的最优策略，因此我们采用 regret 作为策略的性能度量，它衡量的是给定环境下，当前策略与可能的最佳策略在累积收益上的差距：

$$\mathrm{Reg}(\pi;\mathcal{M}) \stackrel{\text{def}}{=} V(\xi^*;\mathcal{M}) - V(\pi;\mathcal{M}) = V^*_{\mathcal{M}} - V(\pi;\mathcal{M}),$$

其中，最佳策略实际上是基于离线竞价日志计算出来的最优决策轨迹。

为了表达分布概率与策略性能之间的正比关系，我们采用了基于能量函数的分布（energy-based distribution）表示来刻画测试环境的分布，regret（遗憾）函数作为该分布的自由能（free energy）函数，α 为温度参数。

$$P_{\text{test}}(\mathcal{M}) = \frac{\exp\left(\frac{1}{\alpha}\mathrm{Reg}(\pi;\mathcal{M})\right)}{Z(\pi)},$$

在确定测试分布的潜在形式后，接下来我们讨论如何从训练环境集合中识别出与测试分布对齐的训练环境。我们选择 Kullback-Leibler（KL）散度来将训练分布投影到一个训练集合 $\mathcal{P}$。经过推导，我们可以得到一个带有熵约束的遗憾最小化优化目标

$$\begin{aligned}&\min_{P\in\mathcal{P}} \mathcal{D}_{KL}\left(P\|P_{\text{test}}\right)\\ =&\max_{P(\mathcal{M})\in\mathcal{P}} \mathbb{E}_{\mathcal{M}}[\mathrm{Reg}(\pi;\mathcal{M})] + \alpha\mathcal{H}(P) + \text{const.}\end{aligned}$$

对于上式的一种直观解释是，优化过程试图寻找到定义在训练集合上的一个环境分布作为训练分布，策略在这些环境中的表现在期望意义上较差（遗憾最大）。同时该优化过程遵循最大熵原则，因为我们对于黑盒对抗因素没有先验知识。上述优化目标中，温度超参数 α 控制熵约束的强度，直观上反映了对抗性的强度。具体而言，当 $\alpha = 0$,

所寻找的训练环境是策略表现最差的环境，代表了最严格的对抗性设定。相反，当 $\alpha \to \infty$，所寻找的训练环境即为训练集合上的均匀分布，而完全不考虑对抗性。可见，该优化目标实际上在严格意义的对抗设定和独立同分布设定通过温度超参数做了插值，**从而既能一定程度应对对抗环境，又能利用大量的日志数据进行离线训练。**

给定上述寻找到的训练分布，我们优化策略使得其最小化遗憾。因此，我们可以得到以下的带有熵约束的极小极大遗憾优化（Minimax Regret Optimization，MiRO）框架：

$$\min_{\pi} \max_{P \in \mathcal{P}} \mathbb{E}_{\mathcal{M}}[\mathrm{Reg}(\pi; \mathcal{M})] + \alpha \mathcal{H}(P).$$

该框架包含两层优化问题，其内层问题寻找在训练集合中与测试分布潜在对齐的训练分布，其外层问题在找到的训练分布下改进当前策略。与以往广泛采用的经验风险最小化（ERM）相比，ERM 假设小的训练经验风险可以推广到小的测试风险，这在对抗条件下往往不成立。**我们所提出的 MiRO 框架在对抗条件下能一定程度保障泛化，因为策略在优化最差情况的遗憾，该目标是测试时遗憾的一个上界。**

2.2 可微分博弈（Differentiable Game）

然而，MiRO 框架所面对的双层优化问题通常难以求解的。借鉴生成对抗网络（Generative Adversarial Nets）一系列工作，我们的主要思路是将极小极大问题转化为一类“可微分博弈”（differentiable game）[5]，这样我们可以借助对偶上升 [10] 来寻找一个有效的解。

MiRO 框架中的 regret function 对于环境 M 并非直接可微，这是由于黑盒的对抗环境中，我们无法观测环境中对抗因素的成因。为了克服这个挑战，我们提出重建世界模型的因果结构，从收集的离线环境（竞价日志）中学习对抗性因素的潜在表示 ω。该方法为我们提供了两点便利：

- 首先，我们可以在学习到的潜在表示空间中搜索训练分布 $p(\omega)$。由于对抗变量 ω 解释了环境的变化，我们将环境 $\mathcal{M}$ 在式上式中替换为 ω。
- 其次，世界模型重建建立了从 ω 到奖励 r 的映射，使得 regret function 可以进行微分。为此，我们可以直接通过基于梯度的优化在 MiRO 中搜索训练分布。

为了学习对抗因素的表征来反映数据的因果效应，我们首先分析了策略在环境中决策

过程中产生的因果结构，然后基于变分信息瓶颈（VIB）[6] 进行表征学习。策略在环境中决策所呈现出的因果模型如下图所示，其中有向箭头反映了因果关系。基于这些因果关系，我们构建了一个世界模型来刻画策略与环境的互动，包括：1）表征模型 $p(\omega_t|h_t)$：将决策轨迹映射为对抗变量；2）观测模型 $p(o_t, a_t, r_t|\omega_t, h_t)$；3）隐空间动态模型 $p(\omega_t|\omega_{t-1}, a_{t-1})$：在隐空间内的动态转移模型。

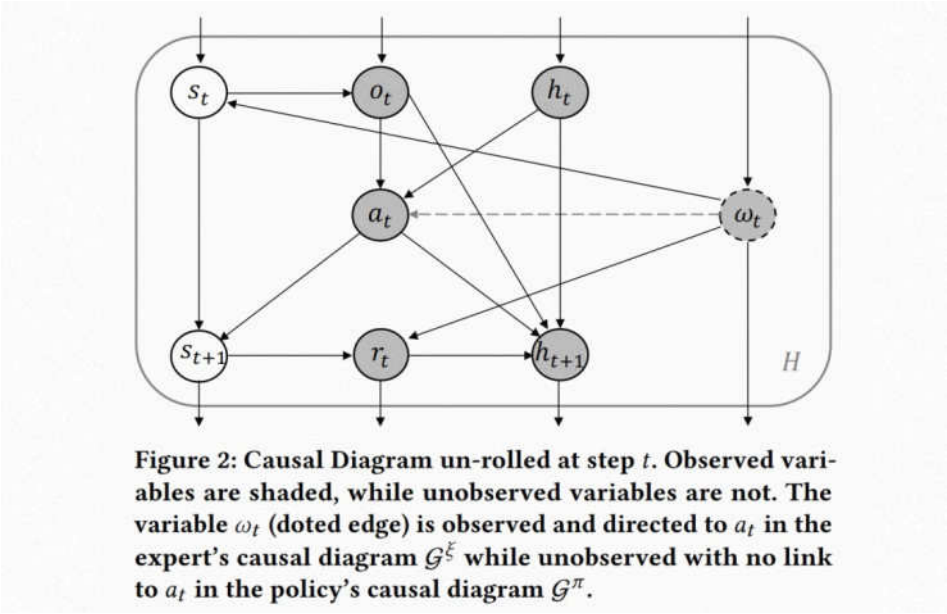


Figure 2: Causal Diagram un-rolled at step t. Observed variables are shaded, while unobserved variables are not. The variable ω_t (doted edge) is observed and directed to a_t in the expert's causal diagram $\mathcal{G}^{\xi}$ while unobserved with no link to a_t in the policy's causal diagram $\mathcal{G}^{\pi}$.

策略与环境交互过程对应的因果模型

其中表征模型提供了对抗因素的表示，通过以下的 VIB 优化目标进行学习：

$$\max \mathop{\mathbb{E}}_{d^{\pi}, d^{\xi}} [\log p(o_t, r_t|\omega_t)] + \mathop{\mathbb{E}}_{d^{\xi}} [\log p(a_t|\omega_t)] - \beta \mathcal{D}_{KL} \left(p(\omega_t|h_t) \| p(\omega_t|\omega_{t-1}, a_{t-1})\right),$$

同时，在优化重建目标时，观测模型包含一个奖励估计器 $E[r_t|s_t, a_t; \omega]$ 的学习来帮助实现 MiRO 的微分化。其优化目标是：

$$\min_{\theta} \mathbb{E}_D \left[\left(r^H - \sum_{t=1}^{H} r_{\theta}(o_t, a_t, h_t, \omega) \right)^2 \right],$$

r^H 代表轨迹的累积奖励。

因为对抗变量 ω 反映环境中对抗因素的成因，我们因此将 MiRO 框架优化目标中环境 $\mathcal{M}$ 替换为 ω，并引入上述的表征模型和奖励估计器，从而得到以下的可微博弈问题：

$$P^{(t)}(\omega) = \arg\max_{P \in \mathcal{P}} \mathop{\mathbb{E}}_{\omega \sim P} \left[\mathrm{Reg}(\pi^{(t-1)}, \omega) \right] + \alpha \mathcal{H}(P),$$

$$\pi^{(t)} = \arg\min_{\pi} \mathop{\mathbb{E}}_{\omega \sim P^{(t)}} \left[\mathrm{Reg}(\pi^{(t-1)}, \omega) \right].$$

直观上来看，想象教师给学生设计习题，学生完成习题获得提升的过程，上述的可微博弈优化过程在两步之间交替进行优化：

- teacher step：该步通过寻找当前策略表现不好的训练环境作为训练分布。
- student step：该步策略会基于所找到的训练分布改进其策略。

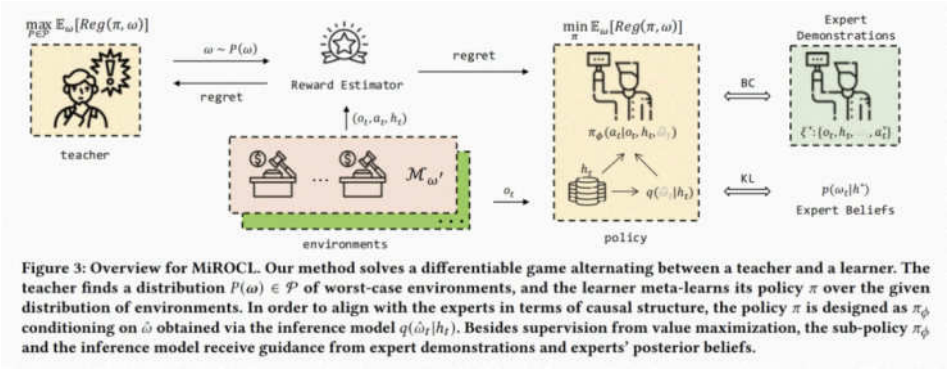


Figure 3: Overview for MiROCL. Our method solves a differentiable game alternating between a teacher and a learner. The teacher finds a distribution $P(\omega) \in \mathcal{P}$ of worst-case environments, and the learner meta-learns its policy π over the given distribution of environments. In order to align with the experts in terms of causal structure, the policy π is designed as π_ϕ conditioning on $\hat{\omega}$ obtained via the inference model $q(\hat{\omega}_t|h_t)$. Besides supervision from value maximization, the sub-policy π_ϕ and the inference model receive guidance from expert demonstrations and experts' posterior beliefs.

MiRO 优化框架概览

3. 实验

针对上述方案，我们在合成竞价环境和现实竞价环境中验证了所提出的方法的有效性。

我们发现，无论是在对抗条件还是独立同分布条件，MiRO 均表现出了优于现有方法的稳健性能。

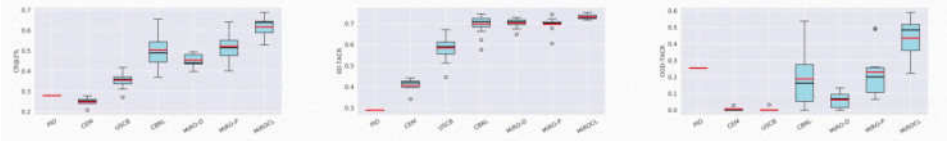


Figure 4: The results of CR@2% (Left), IID-TACR (Middle), and OOD-TACR (Right) in the Industrial dataset are shown above. Each boxplot shows the average (red) and median (black) results of 20 independent repeated runs.

现实竞价环境中，在对抗条件和 iid 条件下不同方法的性能比较

我们也进行了消融实验，验证了方法中的每一个组件的有效性。此外，我们也针对媒体可能采用的机制形式进行了分析和实验。

Table 2: Median scores on Synthetic. The reported mTACR and mCR scores take a median across 20 random trials for all models. We also report results on two types of auction formats.

	mTACR	mCR@2%	GSP-mTACR	MIX-mTACR
MiROCL	0.8094	0.8114	0.7932	0.8098
MiRO-P	0.7231	0.7469	0.7307	0.7116
CBRL	0.6856	0.7003	0.696	0.6793
USCB	0.5171	0.5304	0.5114	0.5256
CEM	0.5811	0.6094	0.5972	0.5438
PID	0.3343	0.3556	0.3728	0.2766

4. 总结

针对展示外投场景面对的黑盒对抗环境下的约束竞价问题，本文基于训练测试对齐的思路，提出了一个极小极大遗憾优化框架（Minimax Regret Optimization，MiRO）。通过对黑盒环境的因果结构进行建模，MiRO 学习了环境的表征以及奖励估计，从而将双层优化问题转化为可解的可微博弈进行交替优化。这种方案可以类比一种 teacher-student 之间交替迭代的稳健学习范式。在合成和现实数据上的实验表明了我们方法的有效性。

参考文献

[1] Alexey Drutsa. 2020. Reserve pricing in repeated second-price auctions with strategic bidders. In International Conference on Machine Learning. PMLR, 2678 - 2689.

[2] Paul D ü tting, Zhe Feng, Harikrishna Narasimhan, David Parkes, and Sai Srivatsa Ravindranath. 2019. Optimal auctions through deep learning. In International Conference on Machine Learning. PMLR, 1706 - 1715.

[3] Yanjun Han, Zhengyuan Zhou, Aaron Flores, Erik Ordentlich, and Tsachy

Weissman. 2020. Learning to bid optimally and efficiently in adversarial first-price auctions. arXiv preprint arXiv:2007.04568 (2020).

[4] Thomas Nedelec, Jules Baudet, Vianney Perchet, and Noureddine El Karoui. 2021. Adversarial Learning in Revenue-Maximizing Auctions. In Proceedings of the 20th International Conference on Autonomous Agents and MultiAgent Systems. 955 – 963.

[5] David Balduzzi, Sebastien Racaniere, James Martens, Jakob Foerster, Karl Tuyls, and Thore Graepel. 2018. The mechanics of n-player differentiable games. In International Conference on Machine Learning. PMLR, 354 – 363.

[6] Alexander A Alemi, Ian Fischer, Joshua V Dillon, and Kevin Murphy. 2016. Deep variational information bottleneck. arXiv preprint arXiv:1612.00410 (2016).

预估模型

排序和准度联合优化：一种基于混合生成 / 判别式建模的方案

本文作者：珞家、寒戍、义潮、越瑶、谨持

本文介绍一种基于混合生成 / 判别式模型——JRC，通过增加模型自由度以及设计巧妙的 logits 交互的方式，优化了排序能力和准度。

摘要

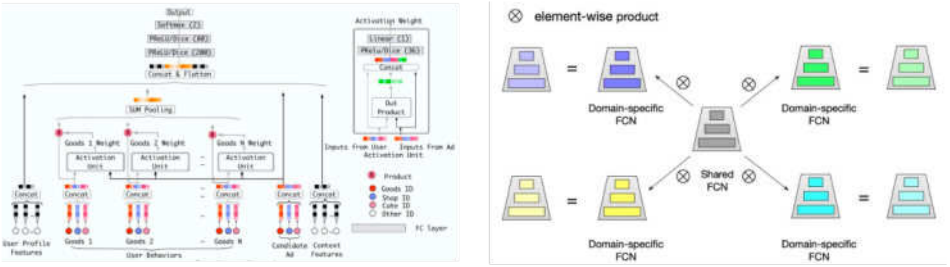
精排预估在广告及推荐系统中扮演着非常重要的角色。基于准度的要求，之前工业界对于模型预估的优化目标，大部分还是采用 pointwise 的 LogLoss。但对于线上效果来说，排序能力也同样重要，离线迭代时我们通常会优化 AUC/GAUC 等排序指标。本文中，我们设计了 JRC 模型，通过增加模型自由度以及设计巧妙的 logits 交互的方式，同时优化排序能力和准度。我们也证明了 JRC 近似于优化混合生成 / 判别式目标，其中判别式的优化目标作为准度约束，listwise 的生成式目标将正负样本 logits 拉开，优化排序能力。比较有意思的一点是，同时优化两个目标并未带来准度和排序能力上的 trade-off，JRC 在优化排序能力的同时也提高了预估准度。原因是 JRC 增加了同 session 样本中的点击 logits 大于未点击 logits 的监督信号，显著地提升了稀疏长尾用户排序能力和准度；此外，通过对预估值增加生成式的先验约束，JRC 在稀疏样本上提升了准度 / 排序能力。**JRC 已在阿里妈妈展示广告各个场景全量上线，取得了显著的线上收益。**基于该项工作整理的论文已被 KDD 2023 接收，欢迎阅读交流。

论 文：Joint Optimization of Ranking and Calibration with Contextualized Hybrid Model

链接（点击↓阅读原文）：https://arxiv.org/abs/2208.06164

1. 背景

回顾展示广告 RANK 团队之前在模型预估上的工作（以 CTR 预估为例），模型创新主要集中在挖掘数据和任务自身的特点，并根据归纳出的 inductive bias 设计精巧的模型结构建模用户兴趣。例如 DIN [1] 观察到用户兴趣具有多峰且局部激活的特性，因此采用 target attention 的方式表达用户对当前 target 的兴趣；STAR [2] 发现不同场景的数据分布相似但又存在差异，因此采用私有和共享参数相乘的方式建模场景差异化的用户兴趣。但另一方面，我们对于 CTR 模型优化目标上的探索相对较少，甚至工业界的相关研究工作也不多。在广告场景下，**评估 CTR 模型需要从准度和排序能力两个方面综合考虑，在实际迭代中，工业界通常在满足准度的要求下，不断优化模型排序能力** [2,3]。



[1] DIN　　[2] STAR

图 1　DIN 和 STAR

1.1 CTR 预估常用损失函数 LogLoss

对于损失函数，CTR 预估大部分还是采用基于 Pointwise 的 Negative Log Likelihood (NLL)，也称 LogLoss。在 Pointwise 模型中，对于每条样本 x，模型会输出一个 logit s_x（在本文中，我们将模型在最后一层激活之前的输出称为 logits），然后经过 sigmoid 得到模型预估点击概率 $\hat{p}(y=1|x)$，其中 $\hat{p}(y=1|x)=\frac{1}{1+\exp(-s_x)}$。LogLoss 优化每条样本层面的预估准度。对应的数学表达为：

$$\ell = -\sum_i \log \hat{p}(y_i|x_i) = -\sum_i y_i \cdot \log \hat{p}(y=1|x) + (1-y_i)\cdot \log\left(1-\hat{p}(y=1|x)\right).$$

我们将 $p(y|x)$ 记为真实点击概率。可以证明，当且仅当 $\hat{p}(y|x)=p(y|x)$ 时，LogLoss 的期望最小。

虽然实践中 LogLoss 可以很好地保持准度，但由于 LogLoss 没显式考虑正负样本的分辨能力，因此也未显式优化排序能力。如果想显式地提升排序能力，通常会使用

pairwise 或者 listwise 的 ranking loss，这些 ranking loss 的目标一般是将同 pair/list 下正样本排在负样本前。例如 pairwise 的 RankNet（如公式 (1) 所示）。

$$-\sum_{i\neq j}\mathbb{I}_{y_i>y_j}\log\frac{\exp(s_{x_i}-s_{x_j})}{1+\exp(s_{x_i}-s_{x_j})},(1)$$

其优化的是正样本 logits s_{x_i} 大于负样本 logits s_{x_j} 的概率。但是**各种优化 Ranking Metric 的损失函数，都会在不同程度上的破坏模型输出作为点击概率的含义，从而导致预估不准的问题 (miscalibration)**[3]。

1.2 Combined Loss

一种直观的想法是将 pointwise loss 与优化 ranking metric 的 loss 相加作为优化目标，通过这种多目标 Combined Loss 来提升排序能力。**实际上，包括 Twitter** [3]**、Google Ads** [4]**、淘宝搜索都尝试过 pointwise loss 与 pairwise loss (RankNet** [5,6]**) 相加作为优化目标，并发现这种方式可以在基本保持准度的情况下提升模型的排序能力。**但这种做法仍然存在两方面的问题：

- 第一，模型输出的 logits 同时定义了两个不同的物理量。一个是点击概率，另一个是 item i 排在 item j 前面的概率，这两个物理量含义不同甚至可能存在冲突，因此在优化的时候存在效果次优的问题，这也可能是 combined loss 在很多场景下无效的原因。
- 第二，我们观察到一个很有意思的现象，在合适的 loss 权重设置下，pointwise loss+ranking loss 的准度（LogLoss/ECE 等）甚至比 pointwise 的 LogLoss 要好。这似乎不符合直觉，因为当评估指标为 LogLoss，优化 LogLoss 是更一致的目标。对于这个现象，之前的方法也缺少相应的理论和实验分析。

1.3 本文的主要目标

我们定义预估模型的建模目标应该是在保持准度的情况下优化排序能力。为此，我们需要重新设计模型的输出以及优化目标。另一方面，我们也希望从理论和实验层面分析为什么各种 combined loss 方法相比 pointwise 模型可以取得更好的准度，为后续工作提供 insight。

2. 预估建模新框架：混合生成 / 判别式建模

由于 combined loss 仅仅使用了一维 logit（一个自由度）来定义两个不同的物理量，直接的 loss 相加的方式可能会导致优化冲突问题。基于这个分析，我们一个直观的想法是将模型的输出 logit 从一维扩展到二维，扩展到二维后用两个自由度来同时优化准度和排序能力。我们最先想到的是 multi-task 的方法，两个 logit 分别优化 ranking 和 calibration。但这种方式下两个 logit 比较独立，缺少合理的融合方式得到 pCTR。例如可以用一个主 logit 用来定义点击概率，另一个辅助 logit 优化排序能力作为辅助任务时，但我们实验发现这种方式下排序能力提升微弱 (GAUC)，这是因为 multi-task 方法中 **ranking loss 对最终排序能力的学习影响有限**。

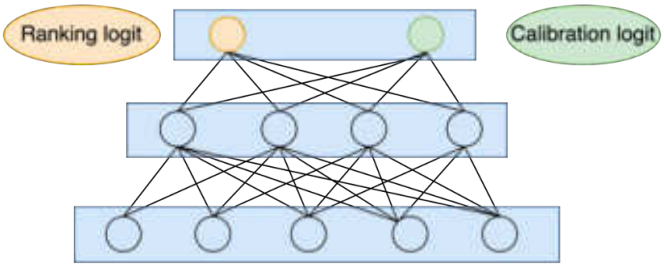


图 2　Multi-Task 方法

2.1 Joint Optimization of Ranking and Calibration (JRC) 模型

为了更好地联合优化准度和排序能力，我们需要重新设计模型的输出以及优化目标，解决之前 logit 定义不同而导致的优化冲突问题，并在统一的 logit 定义出满足该目标的损失函数。为了给两个 logit 建立显式合理的融合方式，我们提出了 Joint optimization of Ranking and Calibration (JRC) 方法，来直接优化排序和准度。

具体来说，在 JRC 的框架下，模型对于一条样本 x 输出两个 logits，$s_x^{[y=0]}$ 和 $s_x^{[y=1]}$，分别对应未点击状态和点击状态。进一步，我们将基于两个 logits 的差来重新计算点击概率，如公式（2）所示：

$$\hat{p}(y=1|x) = \frac{1}{1+\exp(-(s_x^1 - s_x^0))}, (2)$$

得到预测点击概率我们使用 LogLoss 来保证预测准度：

$$-\log \hat{p}(y|x) = -y \cdot \log \hat{p}(y=1|x) - (1-y) \cdot \log\left(1-\hat{p}(y=1|x)\right). (3)$$

可以看到，这一步其实等价于将**单 logit 的 sigmoid 替换成两个 logits 的 softmax**，然后计算 LogLoss。

与此同时，我们将**每个样本的点击 logit 和未点击 logit 都增加一个 listwise 损失提升排序能力，即让正样本的点击 logit 大于同 session 其他样本的点击 logit，让负样本的非点击 logit 大于同 session 样本的非点击 logit**。对于每个样本，listwise loss 形式如公式 (4) 所示：

$$-\log \frac{\exp(s_x^y)}{\sum_{x' \in \text{session}_x} \exp(s_{x'}^y)}. \quad (4)$$

其中 session_x 表示与 x 同 session 的所有样本的集合。

最终的损失函数将由公式 (3) 和公式 (4) 相加得到，

$$-\sum_{x,y} \alpha \log \frac{\exp(s_x^y)}{\exp(s_x^0) + \exp(s_x^1)} + (1-\alpha) \log \frac{\exp(s_x^y)}{\sum_{x' \in \text{session}_x} \exp(s_{x'}^y)}.$$

其中 α 是调节两个 loss 大小的超参，我们实验中发现，JRC 对 α 的大小并不敏感，在大多数情况下离线的准度和 GAUC 都有提升。

综上，我们通过增加自由度以及改变 logit 定义的方式，减少优化排序和准度时的冲突，在保持准度的情况下优化排序能力。如图 3 所示，直观上的理解，**JRC 令点击和非点击 logit 分别优化排序能力，同时约束他们的差作为最终的预测概率**。并且可以证明，在这个 loss 框架下，logits 是具有明确物理意义的。Section 2.2 及附录中给出了对应的证明，这种方式等价于我们用 logits 建模了联合概率 $p(x, y)$ 的 Energy-Based Model，并在此基础上建模混合生成 / 判别式模型。

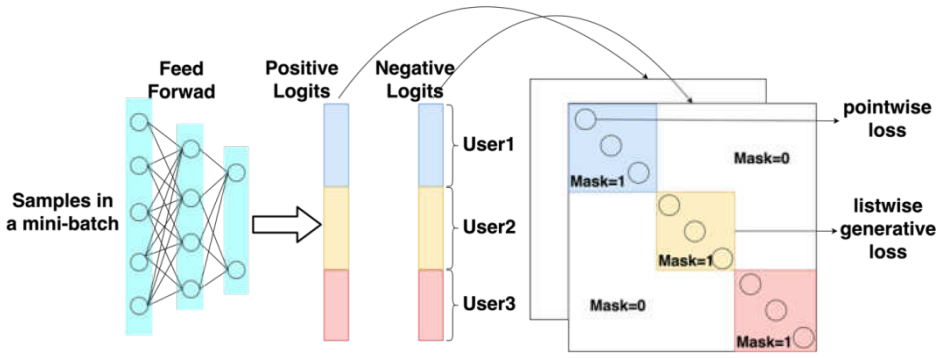


图 3　JRC Loss 计算

2.2 混合生成判别式建模

在统计机器学习的分类问题中，我们将建模 $p(y|x)$ 的模型称为判别式模型，将建模 $p(x|y)$ 的模型称为生成式模型。具体来说，判别式模型直接输出分类的概率，而生成式模型先建模数据生成方式 $p(x|y)$，再通过贝叶斯公式得到分类器。注意这里的“生成式”模型和目前火热的 AIGC 有一些不同，目前的深度生成模型更关注于从目标分布中直接得到高质量的采样，生成高质量样本。统计机器学习中的生成式模型则是直接建模数据生成分布 $p(x|y)$。

在数据量足够大的时候，判别式模型通常能取得较好的准确率。而在数据量较少的情况下，生成式模型可以通过增加先验来提升模型准确度。此外，生成式模型一般具有更好的预估准度。之前的一些工作发现，将生成式模型和判别式模型加以混合 $\log \hat{p}(y|x) + \log \hat{p}(x|y)$ 可以同时提升准确率 (accuracy) 和准度 (calibration) [7,8]。本文中，我们证明 JRC 的 loss 形式近似于优化混合生成判别式目标，这也可以从另一个角度解释为什么 JRC 可以同时提高准度和排序能力。区别于以往 pointwise 方法用 logit 定义判别概率 $p(x|y)$，JRC 将 logit 定义为 (x, y, z) 的联合 energy（其中 z 是当前 context 特征，如 session/user id 等），在此基础上，推导出判别概率（和以往 pointwise 方法相同的形式）和生成概率（类 listwise ranking loss）。以往 pointwise+ranking loss 的直接结合使得 logit 失去物理意义，而 JRC 的 logit 有明确的物理意义。具体来说，JRC 近似在优化如下所示的目标损失函数：

$$\min -\mathbb{E}_{p(x,y,z)} \log \hat{p}(y|x) + \log \hat{p}(x|y,z). \quad (5)$$

其中 $\hat{p}(x|y,z)$ 表示样本 x 出现在点击 $y=1$ 或未点击样本 $y=0$ 中的概率，具体证明详见附录及论文。

2.3 Online Training

我们最开始上线的 JRC 版本选取同 session 的样本作为当前 context，但我们发现在展示广告场景下，同 session 内样本仍然较少（受限于广告比例）。为了增大样本量，我们尝试将同 session 扩大，将每个 user 某个时间段内的样本 group 起来作为 context。这里存在一个 tradeoff – 较长的时间窗口可以引入更多的样本，但是这样可能增大 ODL 资源开销以及训练的延时，而较短的窗口可能会让同 context 下样本变少，模型难以提升排序能力。综合考虑，我们选取了 10 分钟作为窗口，具体实现中，为了支持 JRC 的 context-aware 训练，ODL 数据流内将用户 10 min 内的样

本 group 起来送到一个 batch，并在训练代码中通过 indicator 区分不同的 user。图 4 分别是 JRC 的 tensorflow 伪代码实现。

Algorithm 1 A Tensorflow-style Pseudocode of our proposed JRC model.

```
# B: batch size, label: [B, 2], context_index: [B, 1]
# Feed forward computation to get the 2-dimensional logits
# and compute LogLoss -log p(y|x, z)
logits = feed_forward(inputs)
ce_loss = mean(CrosssEntropyLoss(logits, label))

# Mask: shape [B, B], mask[i,j]=1 indicates the i-th sample
# and j-th sample are in the same context
mask = equal(context_index, transpose(context_index))

# Tile logits and label: [B, 2]->[B, B, 2]
logits = tile(expand_dims(logits, 1), [1, B, 1])
y = tile(expand_dims(label, 1), [1, B, 1])

# Set logits that are not in the same context to -inf
y = y * expand_dims(mask, 2)
logits = logits + (1-expand_dims(mask, 2))*-1e9
y_neg, y_pos = y[:,:,0], y[:,:,1]
l_neg, l_pos = logits[:,:,0], logits[:,:,1]

# Compute listwise generative loss -log p(x|y, z)
loss_pos = -sum(y_pos * log(softmax(l_pos, axis=0)), axis=0)
loss_neg = -sum(y_neg * log(softmax(l_neg, axis=0)), axis=0)
ge_loss = mean((loss_pos+loss_neg)/sum(mask, axis=0))

# The final JRC model
loss = alpha*ce_loss + (1-alpha)*ge_loss
```

图 4　JRC TensorFlow 实现

3. 实验分析

我们在阿里妈妈展示广告生产数据集上进行了一系列实验，来验证以下问题：

- Q1: JRC 相比 combined loss，即 pointwise 直接叠加 ranking loss，是否在准度和排序上取得提升。
- Q2: JRC 为什么相比 pointwise loss 能提升准度。
- Q3: JRC 线上效果 / 准度指标是怎么样的。

对于实验的设置，我们对比了 JRC 和其他三类 baseline 方法：

- Pointwise 模型，使用 LogLoss 为优化目标，这也是线上的主流量 base 模型。
- Combined 模型，这里我们选取了 pairwise 的 RankNet loss 和 listwise 的 ListNet。
- Multi-task 模型，其中 ranking logit 作为辅助任务间接优化排序能力。

对于评价指标，我们主要关注 GAUC/AUC/LogLoss/ECE 等排序和准度指标，具体详见论文实验章节。在大量的实验中，我们发现在优化 user 内排序能力 (GAUC) 的 setting 下，AUC（衡量整体排序能力）和线上准度相关，并且更加稳定，不同方法 GAP 更明显，而 LogLoss 和 ECE 等准度指标离线区分度较小，且受正样本比例影响波动大。因此下文主要汇报 GAUC 和 AUC 的结果。离线进行不同方法对比的时候，我们的原则是调整超参数，在所有方法 AUC 持平的情况下，看 GAUC 涨幅（满

足准度约束的同时优化排序)。

3.1 方法对比

各个对比方法的实验结果如图 5 所示，我们对各个方法进行了公平的调参，同一方法根据 loss 权重的不同（准度和排序 tradeoff）也会有不同版本的结果。

Method		GAUC	AUC
G1	Pointwise	+0%	+0%
G2	Pointwise+RankNet	+0.3%	+0%
	Pointwise+ListNet (V1)	+0.5%	+0%
	Pointwise+ListNet (V2)	+0.8%	-0.2%
G3	Multi-Task	+0.1%	+0%
Ours	JRC (V0)	+0.3%	+0.1%
	JRC (V1 上线版本)	+0.6%	+0%
	JRC (V2)	+1.0%	-0.2%

图 5　实验结果

首先可以看标绿的部分，标绿的部分代表通过调参，使得各个方法在 AUC 持平情况下 GAUC 的最大提升幅度，我们可以得到以下结论：

1. Combined loss（RankNet/ListNet）相对 Pointwise Loss 在排序能力有一定提升，验证了 Combined Loss 的有效性。
2. JRC（V1 上线版本）相对 Combiend loss 在 GAUC 上有进一步提升，说明了 JRC 的有效性。

我们也进行了其他实验，验证各个方法牺牲准度换取排序的能力。标橙色的部分表示 AUC 降幅两个千分点内各个方法的 GAUC 提升，可以看到在这种情况下，JRC 相对 Pointwise+ListNet 仍然更优。

另外，我们也可以看到 JRC (V0) 版本相对 pointwise 模型同时提升了准度和排序能力，这也和之前 Twitter [3] 的发现一致，即 pointwise loss+ranking loss 可以同时提升准度和排序能力。这一部分的实验分析详见论文 4.2.1 及 4.3.1.。

3.2 不同用户上的表现

为了进一步理解 JRC 到底是在哪部分样本上取得了提升，我们根据用户的活跃程度（过去 14 天的点击行为数量）做了分组，并计算 JRC 在不同用户分组上的相对提升。

分组 1 代表用户行为最少（少于 50 次点击），分组 4 代表用户行为最多，每个分组下的样本数量相等。从图 4 的结果中，我们可以得到两个结论。

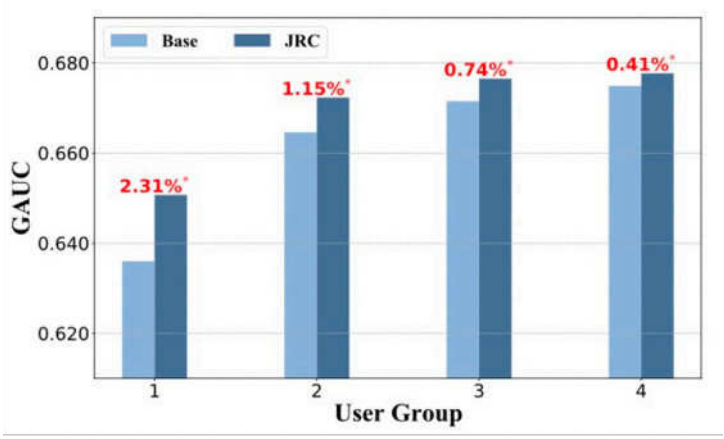


图 6　JRC 在不同活跃程度用户上的 GAUC 相对提升

- JRC 在不同活跃程度上的用户上相比 base 都有提升。
- JRC 在行为稀疏的用户上取得的提升更多，在全网行为少于 50 次的用户上，JRC 相比 pointwise 模型 GAUC 的相对提升达到了 2.31%。这是因为相对 pointwise 模型，JRC 的生成式 loss 增加了更多的监督信号（对比正负样本的 logits），这些补充的信息使得稀疏长尾用户的样本学习更充分。如果从生成式模型角度来看，生成式模型的一个好处就是在训练数据少的情况下通过增加先验提升模型表现，在我们的 setting 下，生成式 loss 帮助了邻域稀疏样本的学习。

3.3 在线效果

JRC 从 BDL（离线天级训练）迁移到 ODL（生产实时训练）后效果存在一定衰减（GAUC 提升幅度从 BDL +0.6% 到 ODL +0.3%），我们通过数据分析发现主要有以下原因：

- ODL 上模型训练充分，这时增加先验 / 监督信号带来的提升衰减。
- ODL 相对 BDL，同 context 下计算 listwise loss 的样本变少。具体来说，ODL 我们是将 user 10min 内的样本 group 起来，而 BDL 是将同 user 1h 内样本 group，同 context 下更多的样本有利于模型进一步提升排序能力。如果 ODL group 窗口进一步从 10min 降为 30s，效果还会继续衰减。

JRC 618 期间在展示广告信息流各场景全量上线，信息流整体（首猜 / 购后 / 动态场景）RPM+2.3%/CTR+4.4%/PPC−2.1%/ROI+2.8%。我们从准度和排序能力分析了模型表现，发现模型排序能力有了很大提升 GAUC+0.3%；JRC 准度相对 base 有一定提升 LogLoss −0.27%。

4. 总结和展望

基于准度的要求，之前工业界对于模型预估的优化目标，大部分还是采用 pointwise 的 LogLoss。但对于线上效果来说，排序能力也同样重要。为此，我们设计了基于混合生成 / 判别式的 JRC 模型，通过增加模型自由度以及设计巧妙的 logits 交互的方式，同时优化排序能力和准度。我们也从理论上证明了 JRC 近似优化混合生成 / 判别式目标，其中判别式的优化目标作为准度约束，listwise 的生成式目标将正负样本 logits 拉开，优化排序能力。比较有意思的一点是我们发现 JRC 可以提高预估准度，原因首先是实验发现由于 JRC 增加了同 session 样本中的点击 logits 大于未点击 logits 的监督信号，显著地提升稀疏长尾用户排序能力和准度，第二通过对预估值增加生成式的先验约束，JRC 在稀疏样本上提升了准度 / 排序能力。

事实上，目前 JRC 只使用了生成概率辅助排序和准度优化，但在 JRC 的 logit 定义下，其实可以做更多有意思的探索。例如我们可以通过联合概率推导出 $p(x)$，与目前火热的 AIGC 相结合，做一个真正的深度生成模型！另一方面得到 $p(x)$ 之后，我们可以进行半监督学习 /explore/ 不确定度的建模，改变目前的 CTR 预估模型建模范式。

References

[1] Guorui Zhou, Xiaoqiang Zhu, Chenru Song, Ying Fan, Han Zhu, Xiao Ma, Yanghui Yan, Junqi Jin, Han Li, and Kun Gai. Deep interest network for click-through rate prediction. KDD 2018.

[2] Xiang-Rong Sheng,Liqin Zhao, Guorui Zhou, Xinyao Ding, Binding Dai, Qiang Luo, Siran Yang, Jingshan Lv, Chi Zhang, Hongbo Deng, and Xiaoqiang Zhu. One Model to Serve All: Star Topology Adaptive Recommender for Multi-Domain CTR Prediction. CIKM 2021.

[3] Cheng Li, Yue Lu, Qiaozhu Mei, Dong Wang, and Sandeep Pandey. Click-through prediction for advertising in twitter timeline. KDD 2015.

[4] Rohan Anil, Sandra Gadanho, Da Huang, Nijith Jacob, Zhuoshu Li, Dong Lin, Todd Phillips, Cristina Pop, Kevin Regan, Gil I. Shamir, Rakesh Shivanna, Qiqi Yan. On the Factory Floor: ML Engineering for Industrial-Scale Ads Recommendation Models. RecSys 2022 ORSUM.

[5] Christopher JC Burges. 2010. From ranknet to lambdarank to lambdamart: An overview. Learning 11, 23-581 (2010), 81.

[6] Sebastian Bruch, Xuanhui Wang, Michael Bendersky, and Marc Najork. An analysis of the softmax cross entropy loss for learning-to-rank with binary relevance. SIGIR 2019.
[7] Rajat Raina, Yirong Shen, Andrew Y. Ng, and Andrew McCallum. Classification with Hybrid Generative/Discriminative Models. NIPS 2003.
[8] Hao Liu and Pieter Abbeel. 2020. Hybrid discriminative-generative training via contrastive learning. arXiv preprint arXiv:2007.09070 (2020).
[9] Yann LeCun, Sumit Chopra, Raia Hadsell, M Ranzato, and F Huang. 2006. A tutorial on energy-based learning. Predicting structured data 1, 0 (2006).
[10] Will Grathwohl, Kuan-Chieh Wang, Jörn-Henrik Jacobsen, David Duvenaud, Mohammad Norouzi, and Kevin Swersky. Your classifier is secretly an energy based model and you should treat it like one. ICLR 2020.

附录

区别于以往 logits 用来直接定义 $\hat{p}(y|x)$，我们可以将 logits s_x^y 定义为 (x, y) 的联合概率，即

$$\hat{p}(x, y) = \frac{\exp(s_x^y)}{Z(\theta)}, (6)$$

其中 $Z(\theta)$ 是 intractable 的 normalization 常数。注意到公式 (6) 是经典的 energy-based model（EBM）的形式 [9,10]，在这里 logits 作为 (x, y) 的联合 energy，并通过联合 energy 定义了 unnormalized 的联合概率。

因为 context 特征 z（如 user id）通常包含在特征 x 中，我们可以将 x, z 合并为 x 得到 $\hat{p}(x, y, z) = \hat{p}(x, y)$。

给定展示 context z，我们可以计算展示 context 条件下的 (x, y) 的联合概率。

$$\begin{aligned}\hat{p}(x, y|z) &= \frac{\hat{p}(x, y)}{\hat{p}(z)} = \frac{\hat{p}(x, y)}{\sum_{x_i \in \mathcal{X}_z} \sum_{y'} \hat{p}(x_i, y', z)} \\ &= \frac{\exp(s_x^y)}{\sum_{x_i \in \mathcal{X}_z} \sum_{y'} \exp(s_{x_i}^{y'})}.\end{aligned} \quad (7)$$

$\mathcal{X}_z$ 表示 share 相同 context z 的所有样本 x 的集合。随后，我们可以通过 marginalizing x 来计算 $\hat{p}(y|z)$,

$$\hat{p}(y|z) = \frac{\sum_{x_i \in \mathcal{X}_z} \exp(s_{x_i}^y)}{\sum_{x_i \in \mathcal{X}_z} \sum_{y'} \exp(s_{x_i}^{y'})}.$$

我们也可以对 $p(x,y,z)$ marginalizing y 来计算 $\hat{p}(x)$，

$$\hat{p}(x) = \hat{p}(x,z) = \frac{\sum_{y'} \exp(s_x^{y'})}{Z(\theta)}.$$

随后我们可以计算点击概率：

$$\begin{aligned}\hat{p}(y=1|x) =& \frac{\hat{p}(x,y=1)}{\hat{p}(x)} = \frac{\hat{p}(x,y=1,z)}{\hat{p}(x)} = \frac{\exp(s_x^1)}{\exp(s_x^0)+\exp(s_x^1)} \\ =& \frac{1}{1+\exp(-(s_x^1 - s_x^0))},\end{aligned} \tag{8}$$

注意到公式 (8) 和公式 (2) 具有相同的形式。我们也可以计算生成式概率 $\hat{p}(x|y,z)$,

$$\hat{p}(x|y,z) = \frac{\hat{p}(x,y|z)}{\hat{p}(y|z)} = \frac{\exp(s_x^y)}{\sum_{x_i \in \mathcal{X}_z} \exp(s_{x_i}^y)}, \tag{9}$$

生成概率的 negative log likelihood 为：

$$-\log \hat{p}(x|y,z) = -\log \frac{\exp(s_x^y)}{\sum_{x_i \in \mathcal{X}_z} \exp(s_{x_i}^y)} \tag{10}$$

可以发现，公式 (10) 和公式 (4) 形式相似，唯一的区别公式 (10) 的分母是 sum over $\mathcal{X}_z$，在实际中无法计算。而公式 (4) 采用了同 session 的样本做分母。也就是说 JRC 的 listwise-like loss 其实是在用当前 session 样本近似优化生成式目标 $\hat{p}(x|y,z)$。

转化率预估新思路：基于历史数据复用的大促转化率精准预估

作者：明言、携远、寒戍、珞家、胤水、蝉惜、谨持

引言

本文分享阿里妈妈展示广告 Ranking 团队针对淘宝平台电商大促周期内转化率精准预估的探索与实践。我们发现并分析了传统转化率预估模型在电商平台大促周期内性能显著下降的现象，并定义了大促周期内转化率精准预估这个具有极大实际业务价值的问题。围绕以数据为中心的建模思路，我们提出了基于复用历史大促数据的 **H**istorical **D**ata **R**euse 算法。该算法在 2022 年双 11 大促期间全量上线后，为展示广告大盘带来了极为显著的收益（**RPM+9%，CVR+16%，ROI+11%**）：创造可观营收增长的同时，还极大的提升了客户体验，达成了客户侧与平台侧的双赢。基于该项工作总结的论文已经被 KDD 2023 录用，欢迎阅读交流。

论 文： Capturing Conversion Rate Fluctuation during Sales Promotions: A Novel Historical Data Reuse Approach

链接（点击↓阅读原文）： https://arxiv.org/abs/2305.12837

一、背景

转化率（CVR）预估模型是搜推广业务中最为核心的算法模块之一，对于衡量流量价值、提升客户体验有着至关重要的作用。在阿里妈妈展示广告业务中，通常用两个核心指标来评价 CVR 预估的精准程度：衡量排序能力的 AUC 以及衡量预估准度的 PCOC（$=\frac{\sum \text{预估CVR}}{\sum \text{真实CVR}}$，理想值为 1）。经过多年技术沉淀（可参考我们以往的工作 ESMM [1]、DEFER [2]），展示广告场景的 CVR 模型日常在线 AUC 接近 0.9，PCOC 稳定在 1 附近。

然而，日常表现尚可的 CVR 模型，在淘宝平台的大促周期内存在着严重的预估性能下降现象：如下图 1，**CVR 模型在各大促周期内的 AUC 均下降数个百分点，PCOC 也呈现严重高 / 低估**。众所周知，大促活动在电商平台十分常见，而预估模型性能的下降会严重影响平台的流量分发效率及客户的投放体验，因此在大促周期内实现 CVR 的精准预估有着巨大的业务价值。

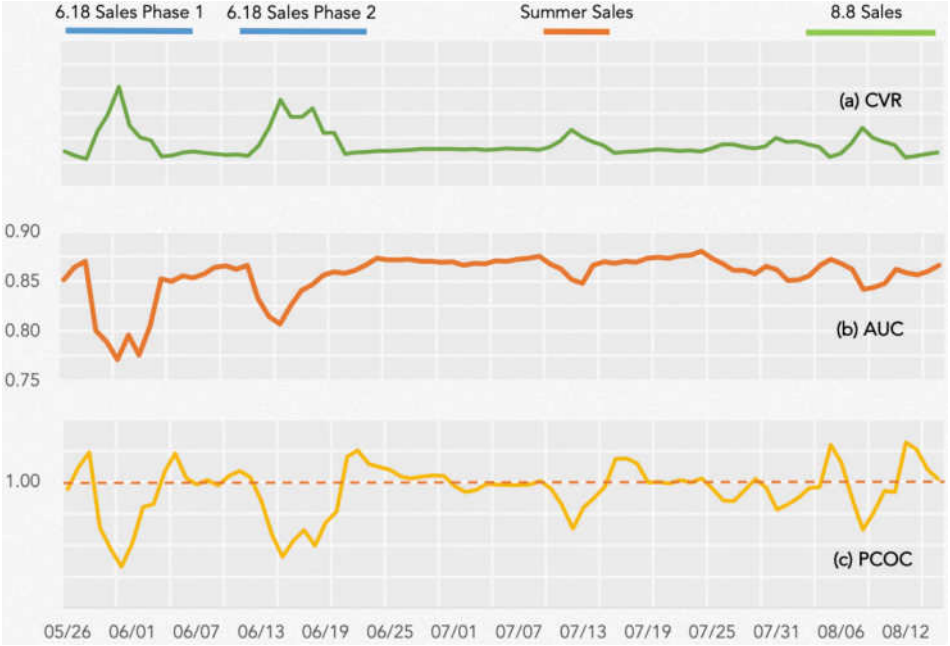


图 1　展示广告场景真实 CVR 变化与主模型预估性能指标变化情况，纵轴仅作为区间参考

为此，我们提出了以数据为中心的建模方案：**H**istorical **D**ata **R**euse 算法（**HDR**，项目名称为**智能数据复用**）。该算法在 22 年双 11 大促期间全量上线后，为展示广告大盘带来了极为显著的收益（大盘整体 RPM+9%，客户侧 CVR+16%，ROI+11%），**创造可观营收增长的同时，还极大地提升了客户体验，达成了客户侧与平台侧的双赢**。同时，我们认为这项工作最大的价值，是提出并实践了搜推广场景针对大促周期 CVR 精准预估的新视角：我们观察到该问题不仅存在于广告业务，同时也普遍存在于搜索 / 推荐业务场景，这使得该方案具备巨大的潜在业务价值。

二、问题分析

首先，我们深入分析了上述 CVR 模型在大促周期内预估性能严重下降的现象，并定位导致问题的主要原因：

- **直接原因 –– 大促周期内用户转化行为突变**：如图 1 (a)，我们可以观察到大促期间的真实 CVR 发生明显波动，其原因是用户的转化行为发生了剧烈变化。由于传统的 CVR 模型遵从 i.i.d. 假设（用于训练的数据与实际服务的数据独立同分布），当分布发生波动时，i.i.d. 假设失效，模型的预估性能将会受到影响；

举一个简单的例子：用户的转化（即购买）行为与大促活动节奏息息相关。大促前一

段时间，用户的实时购买行为会明显下降（除部分用户及不参加活动的店铺外，大部分购买行为都会发生在大促正式开始后）。等大促开始折扣价格生效，用户的购买行为会突然爆发。这种分布发生突变的情况下，i.i.d. 假设明显不再成立。

在搜推广领域，业界解决在线分布变化的常用方案是提升模型的时效性（如流式训练），尽可能用最新的数据训练模型来保证 i.i.d 假设的成立。该方案在 CTR 预估上效果很好，但是 CVR 的延迟反馈特性却进一步复杂化了问题：

- **万“恶”之源 -- 转化行为的延迟反馈**：与点击行为不同，转化行为的标签难以在短时间内被收集，因为转化可能发生在曝光 / 点击后数天甚至数周。因此，业界通常利用延迟反馈建模（DFM）技术 [2–6] 来实现 CVR 模型的实时训练：DFM 方法一般假设延迟转化模式稳定（例如，最终转化的延迟时间保持稳定），通过建模最终转化和转化延迟时间之间的关系来纠正归因不完整的实时标签。然而，由于大促周期内转化行为的剧烈波动同样影响了该假设的成立，因此 DFM 方法也并非合适解法，这也迫使我们必须探索新的建模思路。

再举一个例子帮助大家理解：展示广告场景预估三天内 CVR，即用户点击后三天内发生转化的概率。以淘宝 22 年双 11 大促第一波活动为例，该活动的折扣价格在 10 月 31 日 20 点正式生效。在 29 日到 31 日晚上 20 点之前，很多浏览跟点击发生，但此时用户实时购买会特别少（都在等打折呢）。这时模型从实时数据中获得的反馈是：“最近转化不给力呀，用户都是只看不买，得把预估值都降下去。”然而等 31 日 20 点一到，用户买起来了。大量的转化行为突然爆发，并且很多都会被归因到之前的点击上，导致了 29 日到 31 日晚上 20 点之前的三天转化 CVR 原地起飞（以美妆产品为例，29、30 号以及 31 号 20 点前的三天内 CVR 短时间内会被提升十几倍，从低于日常到远超日常），这就导致了模型此前的预估严重偏低。很明显，除了上帝之外，我们无法从当前的数据来判断数天后到来的大批转化会对当前的 CVR 产生多大影响。

通过对实际数据的观察与分析，我们发现相比起转化模式差异性极大的非大促与大促，大促之间的转化模式却是较为相似的。这启发了我们利用相似历史大促数据来微调模型，使其快速适配当前大促的思路。由于历史大促已经结束，我们可以直接获取到带有完整转化标签的历史大促数据，这也效规避了延迟反馈问题的干扰。基于此思路，我们设计了 **H**istorical **D**ata **R**euse（**HDR，智能数据复用**）算法。

在本文的后续篇幅中，我们将在第三章中详细解读我们提出的 **HDR** 方案；在第四章，

我们通过了多个角度的实验以及实际上线表现，展现了智能数据复用算法的效果；最后，我们会在第五章中讨论历史数据复用机制的潜在问题，以及给出我们对这个极具潜力的预估机制的总结与展望。

三、智能数据复用

在深入探讨算法细节前，我们首先提供该问题的正式描述：以 f_Θ 代表 CVR 模型，其中 Θ 表示可训练参数。我们的生产模型以监督学习的方式使用训练数据 $\mathcal{A}(x,y)$ 来优化 Θ。部署上线后，它将服务于新的在线实时数据分布 $\mathcal{B}(x,y)$（模型服务时，完整的 $\mathcal{B}(x,y)$ 不可获得）。如下图 2 所示，我们的生产模型基于流式训练，以保障模型的时效性。

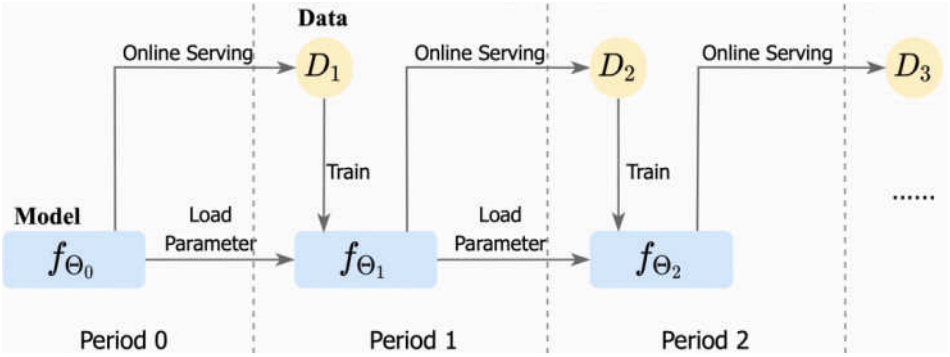


图 2　流式训练示意图

上述训练范式的有效性依赖于 $\mathcal{A}(x,y)$ 和 $\mathcal{B}(x,y)$ 之间的 i.i.d. 假设。然而该假设在大促周期内难以成立，因为转化行为的剧烈波动会带来严重的分布偏移（如图 1（a）所示，CVR 在剧烈波动）。在我们的智能数据复用方案中，我们首先寻找与即将到来的大促 $\mathcal{B}(x,y)$ 的分布相似的历史数据 $\mathcal{B}'(x,y)$，并使用 $\mathcal{B}'(x,y)$ 微调生产模型，过程如下式：

$$f_\Theta : Train(\mathcal{A}) \Longrightarrow f'_\Theta : Train(\mathcal{A}) + Finetune(\mathcal{B}') \qquad (1)$$

其中，f'_Θ 表示微调后适应了大促模式的 CVR 模型。需要强调的是我们采用的是“训练 + 微调”方案，而不是直接丢弃 $\mathcal{A}(x,y)$ 只使用 $\mathcal{B}'(x,y)$ 来训练模型。因为仅使用历史数据训练模型很显然会损害模型的时效性。

复用历史数据是一个简单直观的思路，但为了完成整体方案的构建与生产化，我们需要解决以下三个关键挑战：

- **挑战（1）：如何自动获取相似的历史大促数据 $\mathcal{B}'(x,y)$**。为了方案的生产化，以及防止人工定义复用数据带来的偏差，我们需要建设一个自动化数据检索方案，自动识别出可用的历史数据；
- **挑战（2）：如何修正历史数据中可能存在的偏差。**检索所得的历史大促数据 $\mathcal{B}'(x,y)$ 与即将到来的大促数据 $\mathcal{B}(x,y)$ 之间相似但又难以完全一致，因此我们需要考虑如何修正可能存在的偏差；
- **挑战（3）：如何高效的复用历史数据**。我们需要设计合理的结构以高效利用历史数据来微调模型，并同时避免历史数据重训带来的过拟合问题（又一个臭名昭著的问题：One-Epoch 现象 [7]，即推荐系统领域的深度模型在训练两遍同样的数据后，会出现过拟合，具体可参考文章《深度点击率预估模型的 One-Epoch 过拟合现象剖析》。

3.1 自动数据检索

对于挑战（1），我们设计了一种简单有效的向量检索方案来实现自动搜寻相似历史大促数据的生产化。在我们的系统中，历史数据以自然日为粒度存储，因此我们将历史上的每一天视为检索的候选之一。

为了构建自动检索链路，我们首先利用一组数值特征为每一天构建对应的表征向量。如下图所示，不同大促中的转化模式非常相似——都表现为类似的促前抑制、促中激增和促后平息。抓住这样的趋势，我们可以轻松区分大促周期与非大促周期。在此，我们利用前三天中的两种与转化行为强相关的数据来构建表征向量：

- **前序天内的 CVR：**每天的 CVR 很显然是一个相关性很高的特征。因此，我们利用前三天的一组 CVR 来构建表征。阿里妈妈展示广告的转化归因周期是三天，一天归因 / 两天归因的（不完整的归因）非转化数据可能会在第三天变为转化（完整的归因）。因此，我们同时考虑前一天、两天和三天内完整和不完整归因 CVR，即前一天的一天归因 CVR，前两天的一天 / 两天归因 CVR 以及前三天的一天 / 两天 / 三天归因 CVR。

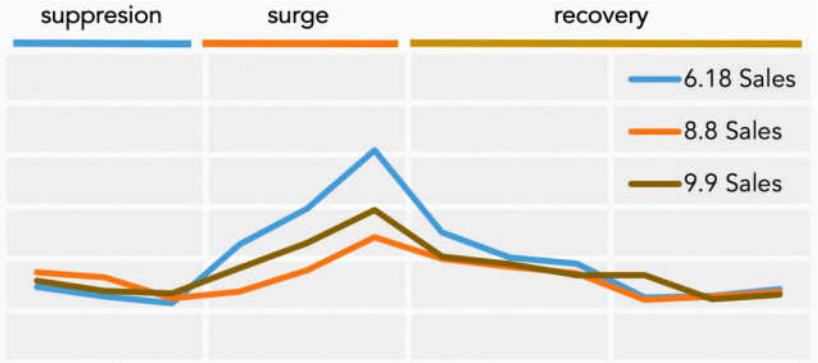


图 3　大促不同阶段 CVR 的变化示意图

- **代表性品类的展现占比：**除此之外，我们还发现某些品类的产品（如化妆品）的曝光占比在大促期间经常剧烈变化，且幅度与 CVR 变化趋势高度一致。因此，我们选定了部分代表性的品类，并利用每个品类在前三天每天的展现占比作为特征。

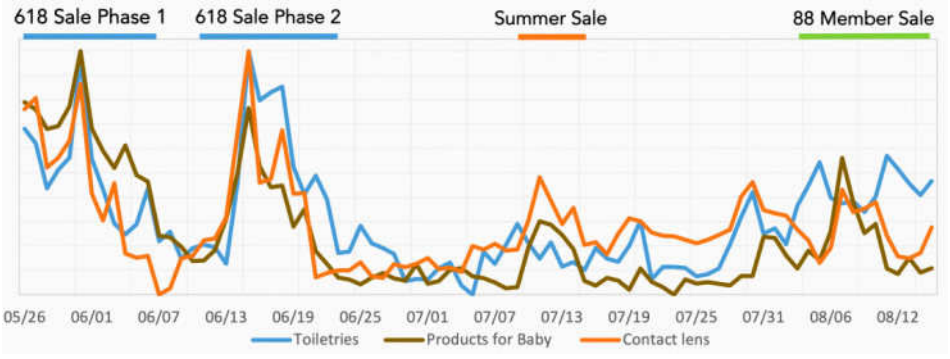


图 4　代表品类的展现占比变化示意图，纵轴为归一化后的展现占比

因为展现占比没有归因周期，很容易实时收集，因此我们还使用了大促当天前 10 个小时的展示占比。尽管这会导致近 10 小时的延迟（因为当天表征的构建需要等待前 10 小时的数据产出），但却显著提升检索结果的准度，使得模型服务在即使存在 10 小时延迟的情况下进一步提升了在线效果。

上述的两类数值特征将会被拼接并平铺成向量，作为对应天的表征。为每一天都构建了对应表征后，我们使用最近邻算法来检索最相似的历史数据：计算当天表征与历史每一天表征之间的余弦距离并排序。在生产中，我们将检索到相似度最高的两天作为相似历史数据输出 $\mathcal{B}'(x, y)$，以供后续的步骤使用。

3.2 基于历史数据的模型微调

Note：本文重点在于讲述方法的整体构思，为了更友好的阅读体验，我们没有遵循论文中的叙述结构，而是整体而概括的描述了微调部分的设计思路；感兴趣的读者可以在论文中看到更数学化而严谨的表述。

在检索到的相似历史数据 $\mathcal{B}'(x,y)$ 后，我们利用该数据微调当前的 CVR 模型，使其快速适配当前大促分布。在这个阶段我们需要解决挑战（2）与挑战（3）。

3.2.1 TransBlock 模块

首先，我们讲述针对挑战（3）的解决过程。在微调实践中，我们遇到了臭名昭著的过拟合问题，也被称为“One-Epoch”问题 [7]，即推荐系统领域的深度模型训练相同的数据两次时，模型将会发生过拟合。而由于我们系统的流式训练方案（见图 2）会加载之前的模型参数，因此对于 CVR 模型而言，历史数据都已经被训练过了。

因此，我们设计了 TransBlock 模块来解决这个挑战。TransBlock 的整体设计思路为直接建模非大促和大促转化之间的转换。如图 5 所示，TransBlock 是一个 Stacking 层，引入了一些主模型之外的可学习参数。微调过程中，TransBlock 参数和主模型参数都将通过训练检索到的数据 $\mathcal{B}'(x,y)$ 进行更新。为了防止主模型发生过拟合，我们训练时为其参数设置了一个极小的学习率，同时保持用常规的学习率训练 TransBlock 参数。这样，原始主模型将受到较少影响，而微调训练后的 TransBlock 模块能够帮助其快速适应当前大促模式。

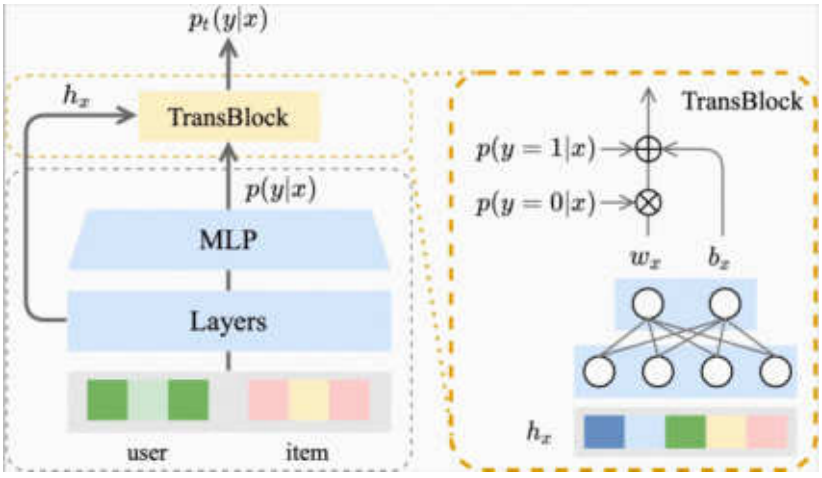


图 5　TransBlock 结构示意图

我们使用公式（2）来建模非大促转化分布和大促转化分布之间的差异。展开来说，我们使用三个部分来代表计算当前的转化概率：（1）从主模型中获取的转化概率 $p(y=1|x)$，（2）非大促模式下不转化但是大促模式下转化的概率 $v_x \cdot p(y=0|x)$，以及（3）从非大促模式下转化但是大促模式下不转化的概率 $u_x \cdot p(y=1|x)$。

$$p_t(y=1|x) = p(y=1|x) + v_x \cdot p(y=0|x) - u_x \cdot p(y=1|x) \qquad (2)$$

u_x 和 v_x 表示基于 x 的样本级别的转移概率，由 TransBlock 模块计算。通过简单的推导，我们可以获得：

$$\begin{aligned} p_t(y=1|x) &= p(y=1|x) + v_x \cdot p(y=0|x) - u_x \cdot p(y=1|x) \\ &= p(y=1|x) + v_x \cdot p(y=0|x) - u_x \cdot (1 - p(y=0|x)) \\ &= p(y=1|x) + (v_x + u_x) \cdot p(y=0|x) - u_x \\ &= p(y=1|x) + \underbrace{w_x \cdot p(y=0|x) + b_x}_{w_x = v_x + u_x, b_x = -u_x}. \end{aligned} \qquad (3)$$

在实践中，我们使用一个简单的全连接层构建 TransBlock，它以主模型中任一隐层的输出 h_x 作为输入，计算公式中的 w_x 和 b_x。具体过程如下式所示，W_{trans} 和 b_{trans} 为可训练参数：

$$\begin{bmatrix} w_x \\ b_x \end{bmatrix} = W_{\text{trans}} \cdot h_x + b_{\text{trans}}. \qquad (4)$$

由于 TransBlock 的输出值为三个部分之和，因此无法保证输出值的范围。因此，我们直接将小于 0 与大于 1 的输出值截断为 0 和 1，来保证输出值在 0 到 1 之间。最终，我们输出截断后的 $p_t'(y=1|x)$ 作为预估 CVR。

3.2.2 用于分布纠偏的微调目标

在获得预估值 $p_t'(y=1|x)$ 后，最直接的微调思路就是优化在历史大促数据 $\mathcal{B}'(x,y)$ 上计算出的交叉熵损失 $\ell(x,y)$，为了与后文形式统一，我们以下式概括：

$$\mathcal{L} = \int \ell(x,y)\, \mathrm{d}\mathcal{B}'(x,y) \qquad (5)$$

然而，如挑战（2）中所提，即使检索出的历史大促数据 $\mathcal{B}'(x,y)$ 和即将到来的目标大促数据 $\mathcal{B}(x,y)$ 非常相似，但我们仍无法保证其完全等价。如图 3 所示，不同的大促间的实际 CVR 是存在一定差异的，我们希望解决这种差异可能带来的影响。为此，我们基于重要性加权经验风险最小化框架（Importance-Weighted Empirical Risk

Minimization）设计了微调方案，通过最小化以下经验性风险来进行模型微调，同时纠正历史数据可能带来的偏差：

$$\mathcal{L} = \int \frac{\mathcal{B}(x,y)}{\mathcal{B}'(x,y)} \cdot \ell(x,y)\, \mathrm{d}\mathcal{B}'(x,y) \qquad (6)$$

其中 $\frac{\mathcal{B}(x,y)}{\mathcal{B}'(x,y)}$ 代表用于纠正历史分布与真实分布之间差异的重要性权重，而 $\ell(x,y)$ 仍旧表示交叉熵损失。

很显然在真实分布未知的情况下 $\mathcal{B}(x,y)$ 不可获得，因此我们无法直接使用公式（6）来进行模型微调。为了使该公式可用，在对实际场景进行深入分析后，我们将公式（6）近似为

$$\mathcal{L} = \int \frac{\mathcal{B}_h(y)}{\mathcal{B}'_h(y)} \cdot \ell(x,y)\, \mathrm{d}\mathcal{B}'(x,y) \qquad (7)$$

其中，$\mathcal{B}'_h(y)$ 代表历史数据对应当天前 10 小时的 CVR 均值，可以从历史数据中统计获得；而 $\mathcal{B}_h(y)$ 代表大促当天前 10 小时的真实 CVR 均值（不可实时获取），我们设计了一个简单的**无监督预估方案**（附录公式 13）对其进行估计（为了准确性，该估计不是样本级别，而是前 10 小时整体数据的 CVR，即期望）。在获得 $\mathcal{B}_h(y)$ 的估计值与 $\mathcal{B}'_h(y)$ 后，我们利用公式（7）作为微调的损失函数。

简单的说，在微调时，我们利用 $\frac{\mathcal{B}_h(y)}{\mathcal{B}'_h(y)}$ 来加权正负样本计算出的 loss，直接使用交叉熵可以视为权重为 1。**其中公式（7）的具体推导过程，以及 $\mathcal{B}_h(y)$ 的求解方案请见本文附录或论文原文 Section 3.2。**

3.2.3 优化过程

在对历史大促数据 $\mathcal{B}'(x,y)$ 中的每条样本计算出预估值 $p'_t(y=1|x)$ 后，我们可以直接利用其来计算交叉熵损失 $\ell(x,y)$。同时，我们将重要性权重 $\frac{\mathcal{B}_h(y)}{\mathcal{B}'_h(y)}$（在第 3.2.2 节中计算）放回公式（7）中，以获得最终的微调损失。在微调过程中，TransBlock 参数 W_{trans} 和 b_{trans} 将会被较大学习率 η_1 进行更新，而原始的主模型参数则使用较小的学习率 η_2。这种优化技巧非常重要，可以避免过拟合问题的同时显著提高模型性能。

3.4 在线部署

本节中，我们将重点描述 HDR 实际部署上线的整体流程。具体来说，我们将图 2 中的流式训练过程升级至图 6 所示的两阶段过程：我们保留原本模型的训练流程，而在

其训练完成后叠加一个微调过程，并将微调后的模型推送上线。

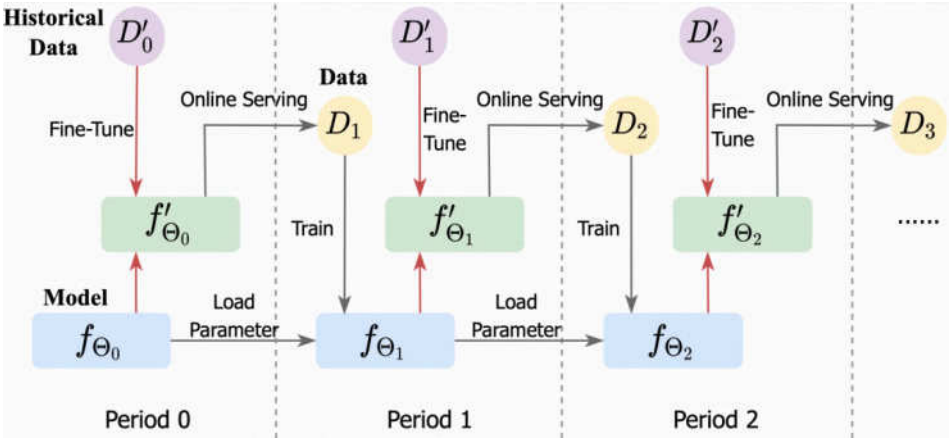


图 6　基于 HDR 的流式训练

对于历史数据检索，我们创建了一个天级的 Map-Reduce 任务，在每天前 10 小时数据产出后，依据 3.1 节中的思路，为当天构建一个向量作为表征。随后，我们计算当天的表征与每个历史每天的表征之间的余弦相似度，并选择前两个日期作为后续复用的历史大促数据。整个检索阶段通常在每天上午 10:30 左右完成。

在微调阶段，我们首先计算出公式（7）中的重要性权重 $\frac{\mathcal{B}_h(y)}{\mathcal{B}'_h(y)}$。在实践中，$\mathcal{B}'_h(y)$ 可以直接从历史数据中获取，而 $\mathcal{B}_h(y)$ 则需要通过方程式（13）求解得出。为此，我们需要获得 $\mathcal{B}_h(\hat{y})$ 和 $\mathcal{B}'_h(\hat{y}|y)$（请见附录）。其中，我们可以通过解析系统的实时日志来获得预估分布 $\mathcal{B}_h(\hat{y})$，而 $\mathcal{B}'_h(\hat{y}|y)$ 则需要通过当前模型下对历史数据进行打分获得。由于，模型微调过程与数据检索阶段是串行的，因此微调后的模型通常在每天上午 11 点左右部署上线。由于早晨流量整体数量较少且 CVR 偏低，因此延迟对整体效果影响不大。

四、效果分析

4.1 实验设置

由于大促 CVR 精准预估问题是第一次被提出，没有相关的公开数据集可使用。因此，我们直接在阿里妈妈展示广告平台的生产数据集上进行了分析实验。我们选取了几类具有代表性的基线方法：

- Base：阿里妈妈展示广告生产模型 DEFER，是一种基于延迟反馈建模的流式

训练方案；

- ES-DFM、DEFUSE：其他经典的延迟反馈建模方案；
- Base w/o DFM：去除延迟反馈建模的 DEFER 模型；
- Base w/ promotion indicator：除了利用历史大促数据进行微调之外，另一种可能的可行方法是在输入中加入指示当前是大促还是非大促的特征。通过这种方式，模型有可能可以学习大促和非大促转化模式的不同。因此我们将其作为待对比的基线之一；
- Base w/ direct retraining：最简单的数据重用方法是直接在检索到的历史数据上重新训练一次生产模型。实际上，这种方法等同于智能数据复用去掉了分布纠偏模块和 TransBlock 模块。然而，这可能会引入所谓的“One-Epoch”问题，我们也将它列为对比方法之一。

在模型的离线评测上，我们使用常见的 AUC 指标来评价模型的排序能力；使用 PCOC，LogLoss 以及 ECE 来评价模型的预估准度，其中 PCOC 越接近 1，LogLoss 以及 ECE 越低，代表模型的预估准度越好；

在模型的在线评测上，我们根据日常迭代的经验，用在线 AUC 以及 PCOC 来评测模型的预估性能；同时，我们用广告系统的业务指标 RPM、CVR、ROI 来评价模型实际带来的业务收益；CVR 自不必多说，而 RPM 是广告系统的核心指标，代表每千次广告展现给平台带来的收入；ROI 代表投入产出比，代表广告引导的成交金额与广告主投入的广告费的比值，正相关于客户体验。

4.2 离线效果

整体效果：如同预期，Base 模型在大促周期的数据上表现不佳（AUC 指标为 0.793，PCOC 仅为 0.471），这与图 1 表现的在线性能下降一致，再次证明了探索适用于生产的大促 CVR 预估方案的必要性。

我们进一步研究了不同延迟反馈建模方案的有效性。根据表 1 所示，我们发现所有的延迟反馈模型（ES-DFM 和 DEFUSE）的表现都与 Base 一样糟糕。相反，在移除延迟反馈模块后（即 Base w/o DFM），模型表现显著优于之前：尽管 PCOC 指标仍然较低，但 AUC 和 ECE 等指标均大幅变优。这直接论证了大促周期内延迟反馈模型的基础假设的不成立。

除此之外，我们也对另外两种可能的方案进行了简单实验，即 Base w/ promotion

indicator 和 Base w/ direct retraining。同样地，从实验结果我们可以发现，它们都没能获得更好的效果。Base w/ promotion indicator 大促 / 非大促特征的引入并没有多大帮助，这可能是由于在连续的学习中遗忘了以前的大促转化模式。同时，直接复用历史数据进行训练也未能取得好效果：AUC 甚至下降了 2.8%。我们认为其主要原因是数据的二次训练导致了过拟合问题。这也再次证明了探索更合适的历史数据复用机制的必要性。

我们提出的智能数据复用（HDR）算法就提供了这样的机制。如下表中所示，HDR 在排序能力和预估准度方面均显著优于所有其他方法：AUC 上优于 Base 方法 10.3%，而 PCOC 为 0.99，几乎达到了理想值 1.0。同时，它也在 Logloss 和 ECE 指标上获得了最佳表现，这些结果充分证明了 HDR 的有效性。

Methods	AUC	PCOC	Logloss	ECE
Base	0.793	0.471	0.143	0.0212
ES-DFM	0.798	0.441	0.149	0.0245
DEFUSE	0.806	0.452	0.145	0.0151
Base w/o DFM	0.826	0.539	0.103	0.0120
Base w/ promotion indicator	0.801	0.479	0.147	0.0191
Base w/ direct retraining	0.779	0.718	0.156	0.0241
HDR	**0.875**	**0.990**	**0.090**	**0.0010**
w/o DSC	0.874	0.803	0.091	0.0051
w/o TransBlock	0.771	0.861	0.167	0.0232

表 1　主实验

消融分析：为了了解分布纠偏（DSC，3.2.2 章所述）的影响，我们进行了移除 DSC 的实验（即 HDR w/o DSC），即公式（7）中的 $\frac{\mathcal{B}_h(y)}{\mathcal{B}'_h(y)}$ 固定为 1（简单的说就是直接用原始交叉熵微调）。如表 1 所示，我们获得了与完整 HDR 方法几乎相同的 AUC，这证明了复用历史数据已经能够带来模型排序能力上的巨大提升。不过预估准度下降了很多：PCOC 仅为 0.803，ECE 也大幅增加（ECE 越低越好）。这说明了 DSC 模块对于预估准度的重要性。

同时，我们也进行了移除 TransBlock 模块（即 HDR w/o TransBlock）的实验。与 HDR 相比，我们发现模型性能因为过拟合而显著下降了，这证明了 TransBlock 对于历史数据复用机制的必要性。

正如在 3.2.3 节中所提，我们在微调阶段中采用两个不同大小的学习率：大学习率 η_1 用于 TransBlock 的优化，而 η_2 用于优化原始主模型。此处，我们将 η_1 设置为 1e−3，并检查模型在不同 η_2 下性能表现。在表 2 中，我们观察到当 η_2 不够小（$\eta_2 = 1\text{e-}4$）的时候，模型性能不佳，因为此时模型发生了过拟合问题。然而，如果完全不更新原始主模型参数（即 $\eta_2 = 0$）也不是最佳选择。因此，选择适当的学习率也是微调的另一个重要因素。

Setting	AUC	PCOC	Logloss	ECE
$\eta_2 = 0$	0.866	0.953	0.094	0.0015
$\eta_2 = 1e^{-5}$	0.875	0.990	0.090	0.0010
$\eta_2 = 1e^{-4}$	0.842	1.012	0.085	0.0023

表 2　学习率消融实验

检索效果：在表 3 中，我们提供了几个真实检索结果来更好地展现数据检索的效果。第一个是查找与 99 大促相似的促销。我们检索到的前两个日期是 2022 年 8 月 8 日的 88 大促，以及 2022 年 6 月 14 日的 618 大促二峰，CVR 也都比较接近。第二个例子是寻找与 88 促销相似的促销。我们检索到的 Top2 结果是 2022 年 7 月 12 日的狂暑季大促，以及 7 月 31 日的七夕节大促（没有检索到 99 大促是因为 88 大促发生在 99 大促之前）。同时，我们还随机展示了一个低相似度的非大促日期。显然，这个随机日期的整体 CVR 与目标大促日期相差很大。

	Date		CVR	Similarity
Target	2022/09/06	9.9 Sales	**x**	-
Top Two Dates	2022/08/08	8.8 Sales	0.82 **x**	0.9923
	2022/06/14	6.18 Sales	0.84 **x**	0.9919
Randomly-picked Date	2022/05/28	No Sales	0.58 **x**	0.9212
Target	2022/08/08	8.8 Sales	**y**	-
Top Two Dates	2022/07/12	Summer Sales	0.97 **y**	0.9937
	2022/07/31	Qixi Sales	0.83 **y**	0.9910
Randomly-picked Date	2022/05/28	No Sales	0.71 **y**	0.8902

* For the purpose of data anonymity, we only report the relative CVR values.

表 3　历史数据检索结果展示 (CVR 相对值参考)

4.3 上线效果

业务效果：双十一大促期间，智能数据复用方案在展示广告信息流主场景全量上线，全周期（10 月 23 日 ~ 11 月 11 日）为展示大盘信息流整体带来了 RPM+9%，CVR+16%，ROI+11% 的显著提升，创造可观营收增长的同时，提升了客户体验，达成了客户侧与平台侧的双赢。以淘宝首页猜你喜欢场景的年同比为例，智能数据复用方案的加持使得该场景下的广告流量 CVR 年同比显著提升，广告流量转化的上涨甚至直接明显拉动了整个场景大盘的 CVR。

模型性能：如图 7 所示，智能数据复用方案显著提升了 CVR 模型大促周期内的排序能力以及预估准度：

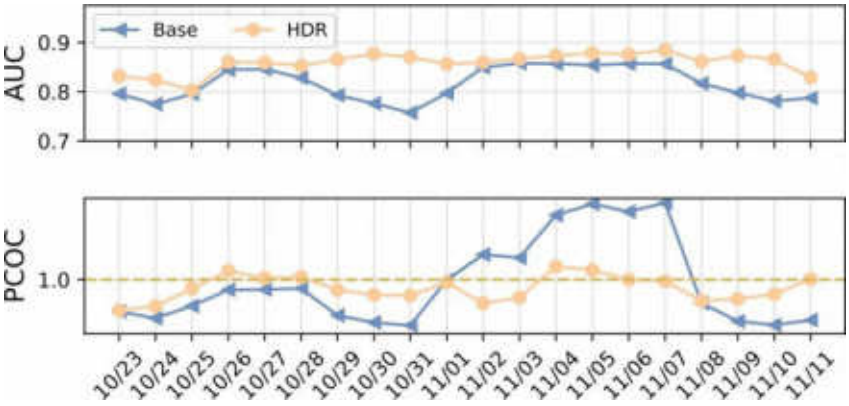


图 7　模型在线 AUC 与 PCOC

- **排序能力**：如图 7 上图所示，智能数据复用模型全周期 AUC 显著优于 Base 模型，基本解决了大促周期内预估模型排序能力下降的问题，全周期均有百分点级别增长；同时，其效果在大促爆发日效果尤为明显，可以获得接近 10 个百分点的相对提升；
- **预估准度**：如图 7 下图所示，智能数据复用模型显著提升了大促周期内的 CVR 预估准度，显著的缓解了促前和促中低估，以及促后高估问题。

排序能力 AUC 以及预估准度 PCOC 的巨大提升，是显著的业务效果的来源。

Category	Base	HDR
women's clothing	0.139	0.695
children's clothing	0.135	0.721
sports shoes	0.054	0.735
women's shoes	0.152	0.699
sportswear	0.071	0.743

表 4　分产品品类预估准度

- **细粒度预估准度**：从更细粒度看，如表 4 所示，智能数据复用方案显著提升了大促周期内所有类目的预估准度。其中部分类目（例如“运动鞋”）甚至从低估 10 ~ 20 倍提升至与其他类目准度相近。证明了智能数据复用模型能够学习到不同类目的高低估的差异性，而不是“一刀切”的整体校准。

五、总结

5.1 潜在问题讨论

智能数据复用方案的一个潜在问题是历史数据的重用可能受到数据过时的影响。具体而言，在线广告系统是动态的，因为活跃的广告主和广告活动不断变化，因此不同周期的活跃广告可能会有显著差异。特别是在大促期间，市场需求的激增也带来了很多不可能从历史上观测到的新广告出现。在过时的历史数据上微调的模型可能并不是最优的，甚至可能对在线性能有损。

然而，在实践中我们并没有观察到任何效果下跌的问题。我们认为这归功于 TransBlock 微调机制的设计：在该设计中，原始模型因为学习率很小（带有稀疏 ID，表示每个特定广告）很少更新，而仅具有少量参数 TransBlock 模块学习到的是粗粒度的知识（例如产品品类级别的差异，如表 4 所示）。我们将在未来的研究中进一步探索这种现象，以更好地理解历史数据重用机制。

5.2 结论与展望

针对大促周期 CVR 预估性能下降导致的流量价值衡量不准的问题，我们开创性的提出了智能数据复用算法，解决了大促周期下的 CVR 精准预估问题，获得了显著的线上收益。更重要的，我们认为这项工作最大的价值是提出并实践了一个提效新视

角，所谓算法诚可贵，问题价更高，在与其他团队的交流中，我们发现大促 CVR 预估性能下降，不仅局限于推荐广告场景，在搜索广告、搜索与推荐的自然流量上普遍存在，目前阿里内部已经有多个团队尝试了智能数据复用所提出的复用历史数据的机制，并均获得了显著效果。

后续我们将继续从智能数据复用全周期生效（例如本周复用上周对应时间点数据，全周期替换传统 CVR 预估方案），无监督微调方案细粒度化等角度继续探索该建模思路的潜力。同时，我们认为该种方案不仅局限于电商大促场景，类似的问题在其他常见的搜推广场景中也存在。例如新闻或短视频推荐，在这些场景中，用户兴趣可能会在特殊事件期间突然转移。例如，在超级碗或圣诞节期间，社交媒体平台上的用户行为可能会偏离日常。我们希望我们的研究也为这些场景提供有价值的参考。

附录：分布纠偏微调设计详解

如挑战（2）中所提，尽管检索出的历史大促数据 $\mathcal{B}'(x,y)$ 和即将到来的目标大促数据 $\mathcal{B}(x,y)$ 非常相似，但我们无法保证其完全等价。如图 3 所示，不同的大促间的实际 CVR（特别是促中激增阶段）还是存在一定差异的，我们希望在微调阶段解决这种差异的影响。为此，我们基于重要性加权经验风险最小化框架（Importance-Weighted Empirical Risk Minimization）设计了微调方案，通过最小化以下经验性风险来进行微调，同时纠正历史数据中可能存在的偏差：

$$\mathcal{L} = \int \frac{\mathcal{B}(x,y)}{\mathcal{B}'(x,y)} \cdot \ell(x,y)\, \mathrm{d}\mathcal{B}'(x,y) \qquad (8)$$

其中 $\ell(x,y)$ 代表在历史数据 $\mathcal{B}'(x,y)$ 上计算的交叉熵损失。$\frac{\mathcal{B}(x,y)}{\mathcal{B}'(x,y)}$ 是重要性权重，用来纠正历史分布与真实分布之间的差异。通过贝叶斯定理，我们可以将上式推导为：

$$\mathcal{L} = \int \frac{\mathcal{B}(x|y)}{\mathcal{B}'(x|y)} \cdot \frac{\mathcal{B}(y)}{\mathcal{B}'(y)} \cdot \ell(x,y)\, \mathrm{d}\mathcal{B}'(x,y) \qquad (9)$$

其中，$\mathcal{B}(x|y)$ 和 $\mathcal{B}'(x|y)$ 代表条件输入分布（conditional input distribution），$\mathcal{B}(y)$ 和 $\mathcal{B}'(y)$ 代表标签分布。很显让，我们无法直接使用公式（9）来微调模型。因为模型需要服务当天，而我们无法在当天一开始就获得 $\mathcal{B}(y)$、$\mathcal{B}(x|y)$ 和 $\mathcal{B}(x,y)$。为此，我们通过大量的数据分析，做出了两个假设来保证纠偏模块的可用：

- 假设 1：我们认为历史数据与真实数据的分布间符合 label shift 假设，即它们各自的条件输入分布相等：$\mathcal{B}(x|y) = \mathcal{B}'(x|y)$。

为了证明上述假设 1 的合理性，我们对促销活动的条件输入分布进行了大量的数据分析。如下表 5 所示，我们展示了 88 大促与 99 大促爆发日时的一些实例，其对应的条件概率为 $p(\mathrm{U}_i, \mathrm{C}_j|y=0)$ 和 $p(\mathrm{U}_i, \mathrm{C}_j|y=1)$。

User Group × Category	8.8 Sales		9.9 Sales	
	$y=0$	$y=1$	$y=0$	$y=1$
$\mathrm{U}_1, \mathrm{C}_1$	0.9597%	0.3831%	1.1013%	0.3595%
$\mathrm{U}_1, \mathrm{C}_2$	0.0064%	0.0472%	0.0065%	0.0575%
$\mathrm{U}_2, \mathrm{C}_1$	0.3125%	0.1738%	0.3441%	0.1597%
$\mathrm{U}_2, \mathrm{C}_2$	0.1769%	0.0258%	0.1738%	0.0288%

表 5　U1，U2 代表两类用户，C1，C2 代表两种产品品类。表中展示为计算其对应占比来表示条件输出分布

可以观察到，两次大促对应的 $p(x|y)$ 是极为相似的，这也为假设 1 提供了支撑。基于假设 1，我们可得：

$$\mathcal{L} = \int \frac{\mathcal{B}(y)}{\mathcal{B}'(y)} \cdot \ell(x, y)\, \mathrm{d}\mathcal{B}'(x, y) \qquad (10)$$

其中标签分布 $\mathcal{B}(y)$ 和 $\mathcal{B}'(y)$ 是按照自然日的粒度进行估计的。其中 $\mathcal{B}'(y)$ 可以通过历史数据计算得出，而 $\mathcal{B}(y)$ 仍然不可获取。为此，我们进一步利用了分布估计的方法 [4,5] 来预估 $\mathcal{B}(y)$：我们首先引入 $\hat{y}$ 来表示模型对输入 x 的预估，即 $f_\Theta(x)$。对预估值的分布 $\mathcal{B}(\hat{y})$ 可以进行如下分解：

$$\begin{aligned} \mathcal{B}(\hat{y}) &= \sum_{y\in\{0,1\}} \mathcal{B}(\hat{y}|y) \cdot \mathcal{B}(y) \qquad (\text{全概率公式}) \qquad (11) \\ &= \sum_{y\in\{0,1\}} \mathcal{B}'(\hat{y}|y) \cdot \mathcal{B}(y), \qquad (\text{假设 } 1) \end{aligned}$$

如果 $\mathcal{B}(\hat{y})$ 和 $\mathcal{B}(\hat{y}|y)$ 都可得，我们就能计算出 $\mathcal{B}(y)$ 的闭式解。然而，$\mathcal{B}(\hat{y})$ 表示模型对当天全天的预估分布，直到本天结束之前都不可完全获取。因此，我们基于观察提出假设 2，以模型服务延迟为代价，仅使用当天的部分数据进行预估。

假设 2：我们用 $\mathcal{B}'_h(y)$ 表示为使用从 0 点到第 h 小时的数据计算的预估分布。我们假设在一天中，无论使用多少小时的数据，两个促销活动之间的差异都保持相对稳定，即 $\frac{\mathcal{B}(y)}{\mathcal{B}'(y)} = \frac{\mathcal{B}_h(y)}{\mathcal{B}'_h(y)}$。

我们同样通过大量数据分析佐证了 $\frac{\mathcal{B}_h(y)}{\mathcal{B}'_h(y)}$ 的稳定性。如下图所示，我们可以观察到两天间的 $\frac{\mathcal{B}_h(y)}{\mathcal{B}'_h(y)}$ 比例在天内相对稳定，其最大的差异不超过 5%：

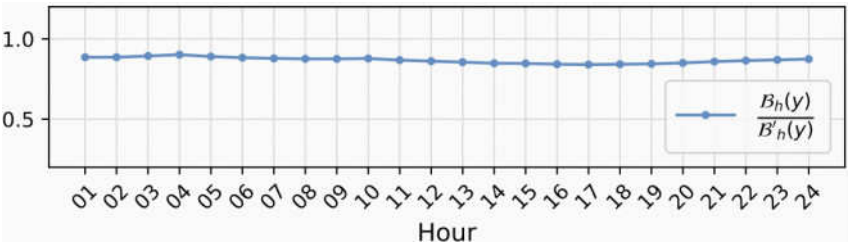


图 8　稳定性示意图

注意：我们系统中的促销节奏通常是相对固定的，因此我们提出了假设 2 来简化问题。当大促节奏变化而导致假设 2 不成立时，可以有以下替代方案：(1) 在实践中，我们观察到用户行为与当前时间和促销开始时间之间的时间差高度相关。因此，我们可以通过将时间间隔与绝对时间点对齐来计算 $\frac{\mathcal{B}_h(y)}{\mathcal{B}'_h(y)}$，此时 h 代表相对时间间隔，而非绝对时间点；(2) 另一种替代方案是增加模型更新频率。例如，我们可以每小时甚至每半小时计算一次 $\frac{\mathcal{B}_h(y)}{\mathcal{B}'_h(y)}$，这样可以更频繁、更准确地捕捉变化。

在生产中，我们设置；因此，模型服务被延迟了 10 个小时 (如 3.1 所述，检索的完成时间点本身就会晚于 10 点，因此这不会导致更大的延迟)。实际情况中，我们发现该延迟不会对在线效果产生太大影响，因为一天的有效转化往往都发生在 10 点之后。基于假设 2，我们将公式 (10) 进一步推导为：

$$\mathcal{L} = \int \frac{\mathcal{B}_h(y)}{\mathcal{B}'_h(y)} \cdot \ell(x, y)\, \mathrm{d}\mathcal{B}'(x, y) \quad (12)$$

已知 $\mathcal{B}'_h(y)$ 可以通过历史数据计算获得。而对于 $\mathcal{B}_h(y)$，我们将 $\mathcal{B}'_h(y)$ 与 $\mathcal{B}_h(y)$ 带入公式 (11) 求解：

$$\mathcal{B}_h(\hat{y}) = \sum_{y \in \{0,1\}} \mathcal{B}'_h(\hat{y}|y) \cdot \mathcal{B}_h(y) \quad (13)$$

在第 $h+1$ 个小时，我们 $\mathcal{B}_h(\hat{y})$ 和 $\mathcal{B}'_h(\hat{y}|y)$ 两个变量均已知，则 $\mathcal{B}_h(y)$ 成为式子内唯一未知变量。因此，我们可以直接求出 $\mathcal{B}_h(y)$ 的闭式解。详细求解过程此处不再赘述，请参考论文附录 A。

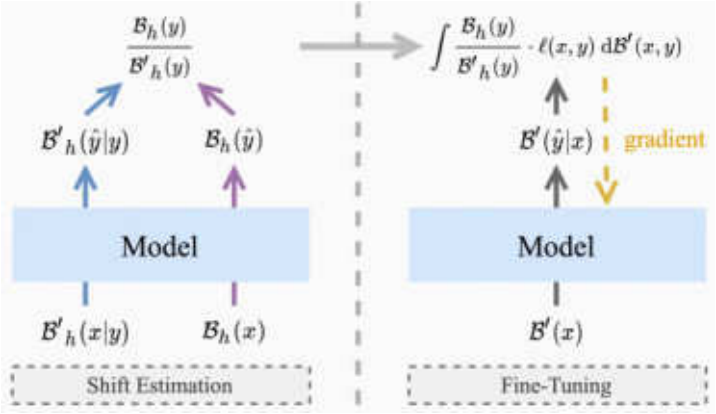


图 9　微调流程

最后，我们总结上述无偏微调流程如图 9 所示。

参考文献

[1] Ma, Xiao and Zhao, Liqin and Huang, Guan and Wang, Zhi and Hu, Zelin and Zhu, Xiaoqiang and Gai, Kun. Entire Space Multi-Task Model: An Effective Approach for Estimating Post-Click Conversion Rate, SIGIR 2018

[2] Gu, Siyu and Sheng, Xiang-Rong and Fan, Ying and Zhou, Guorui and Zhu, Xiaoqiang. Real Negatives Matter: Continuous Training with Real Negatives for Delayed Feedback Modeling, KDD 2021

[3] Olivier Chapelle. Modeling delayed feedback in display advertising. KDD 2014

[4] Ktena, Sofia Ira and Tejani, Alykhan and Theis, Lucas and Myana, Pranay Kumar and Dilipkumar, Deepak and Husz{ ‘a}r, Ferenc and Yoo, Steven and Shi, Wenzhe. Addressing delayed feedback for continuous training with neural networks in CTR prediction. RecSys 2019

[5] Yang, Jia-Qi and Li, Xiang and Han, Shuguang and Zhuang, Tao and Zhan, De-Chuan and Zeng, Xiaoyi and Tong, Bin. Capturing delayed feedback in conversion rate prediction via elapsed-time sampling. AAAI 2021

[6] Chen, Yu and Jin, Jiaqi and Zhao, Hui and Wang, Pengjie and Liu, Guojun and Xu, Jian and Zheng, Bo. Asymptotically Unbiased Estimation for Delayed Feedback Modeling via Label Correction. WWW 2022

[7] Zhang, Zhao-Yu and Sheng, Xiang-Rong and Zhang, Yujing and Jiang, Biye and Han, Shuguang and Deng, Hongbo and Zheng, Bo. Towards Understanding the Overfitting Phenomenon of Deep Click-Through Rate Prediction Models, CIKM 2022

[8] Lipton, Zachary and Wang, Yu-Xiang and Smola, Alexander. Detecting and correcting for label shift with black box predictors. ICML 2018

[9] Azizzadenesheli, Kamyar and Liu, Anqi and Yang, Fanny and Anandkumar, Animashree. Regularized Learning for Domain Adaptation under Label Shifts, ICLR 2019

关于我们

阿里妈妈展示广告核心算法团队负责广告系统的核心召回及排序算法迭代和创新研发，致力于利用人工智能前沿技术打造超大规模体量下的工业级深度学习框架和创新解决方案及相应的基础设施。团队在用户兴趣建模（DIN，DIEN，SIM）、大规模检索技术（TDM、二向箔）等技术持续深耕，并在工业级领域的深度学习框架上有着深厚的积累。近年在 SIGKDD、SIGIR、NIPS、CIKM、AAAI 等学术会议上发表多篇论文。欢迎感兴趣同学加入我们 ~

简历投递邮箱：alimama_tech@service.alibaba.com

基于特征自适应的多场景预估建模

作者：亚岁、添遇、影书、石士

1. 摘要

为了提供更好的搜索体验，满足用户广泛的购物需求，当前手机淘宝支持用户以多种形式来进行搜索。除了常用的文本搜索外，还支持拍照搜索、相似商品搜索以及由投放在外部媒体商品所引导和触发的兴趣搜索。这些场景的特点是 Query 的模态多种多样，我们把除文本搜索外的其余搜索场景统称为多模态搜索场景。多模态搜索的特点是场景多、数据少、差异大，这给点击率预估带来了更大的挑战。针对单场景样本少、行为稀疏的问题，业界常用方法是做多场景联合建模（Multi-Scenario Learning, MSL），MSL 不仅可以减少模型的数量、降低部署成本，还可以利用多场景之间数据的共性来缓解稀疏性问题，同时一定程度上建模各场景的差异性。

相比同模态 Query 下的多场景建模，异构模态 Query 背后反映的是不同用户群体和显著的购物意图差异，因此多模态场景的差异性更大。如何更好的建模这种共性和差异性，是在多模态场景下做 MSL 的核心挑战。针对上述痛点，我们提出一种基于特征自适应学习的多场景联合建模方法 Maria（Multi-Scenario Ranking with Adaptive Feature Learning, Maria）。该方法可以实现端到端的自适应特征学习，具备捕捉多场景差异性知识的能力，解决了现有研究方法普遍存在的效果不平衡问题。基于该项工作整理的论文已被 SIGIR 2023 接收，欢迎阅读交流。

论文：Multi-Scenario Ranking with Adaptive Feature Learning

链接（点击↓阅读原文）: https://arxiv.org/abs/2306.16732

2. 背景

多模态搜索场景支持用户通过不同模态的 Query 来表达多样的搜索需求。如图 1 所示，多模态搜索包含了拍照搜索（Visual Search）、相似商品搜索（Similar Search）和发生在外部媒体的兴趣搜索（Interest Search）三个场景。拍照搜索以用户实拍图作为 Query，相似商品搜索和兴趣搜索以商品信息作为 Query。不同搜索场景和 Query 模态下用户的搜索意图和行为习惯具有明显的差异。阿里妈妈搜索广告团队针对上述异构模态 Query 和差异化显著场景下的点击率（Click Through Rate, CTR）

预估问题进行研究。



图 1　多模态搜索场景描述

由于各场景独立建模存在小场景学习不充分和维护成本高的问题，常用的做法是通过 MSL 技术进行多场景联合的点击率预估建模。而不同场景存在数据分布上的差异，因此目前多场景研究的主要挑战在于如何同时建模场景共性和差异性，并在近几年涌现了不少经典工作，例如 STAR[2]、MMoE[3]、PLE[4]、AEMS2[5] 等。如下图 2 所示，我们将现有 MSL 工作抽象归纳为 Auxiliary Paradigm、Expert Paradigm 和 Multi-Tower Paradigm 三种范式。

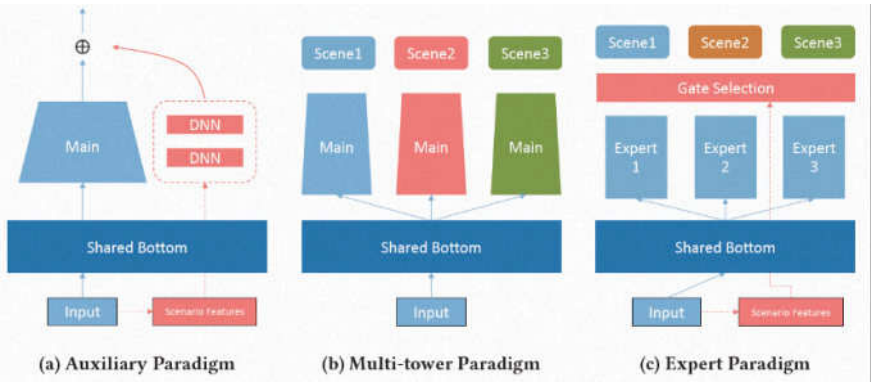


图 2　MSL 的三种技术范式

现有三种范式专注于网络上层模块的优化，但在模型底层都使用简单的 Shared Bottom 策略进行 embedding 特征共享，其背后的假设是不同场景下的底层特征表示是一致的。然而实际上，我们在多场景建模实践中发现上述三种技术范式在复杂跨模态跨媒体的业务场景中无法很好的提取不同场景的差异，导致多场景效果不平衡问题。因此，我们提出了一种基于特征自适应学习的多场景建模方法 Maria，聚焦在模型底层特征的自适应学习，实现多场景差异性知识的捕捉。

3. 方法

简而言之，我们提出基于自适应特征学习的多场景排序模型。如下图 3 所示，不同于现有方法的范式，在该模型的底层，我们设计了三个主要模块来保证有区分力的特征学 习：Feature Scaling（FS）、Feature Refinement（FR）、Feature Correlation Modeling（FCM）。

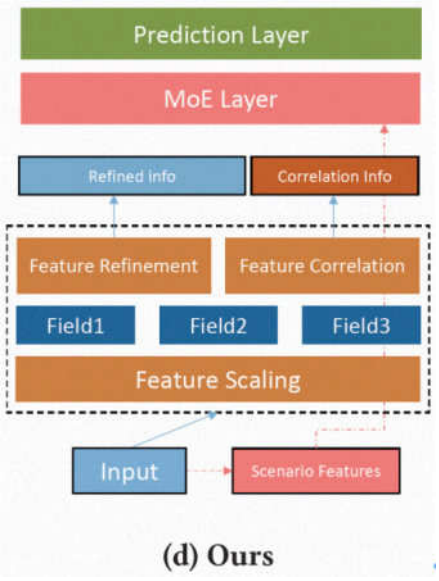


图 3　基于特征自适应学习的新范式

整体的框架图如图 4 所示，我们将按顺序详细介绍 FS、FR、FCM 和网络层预测层等模块。

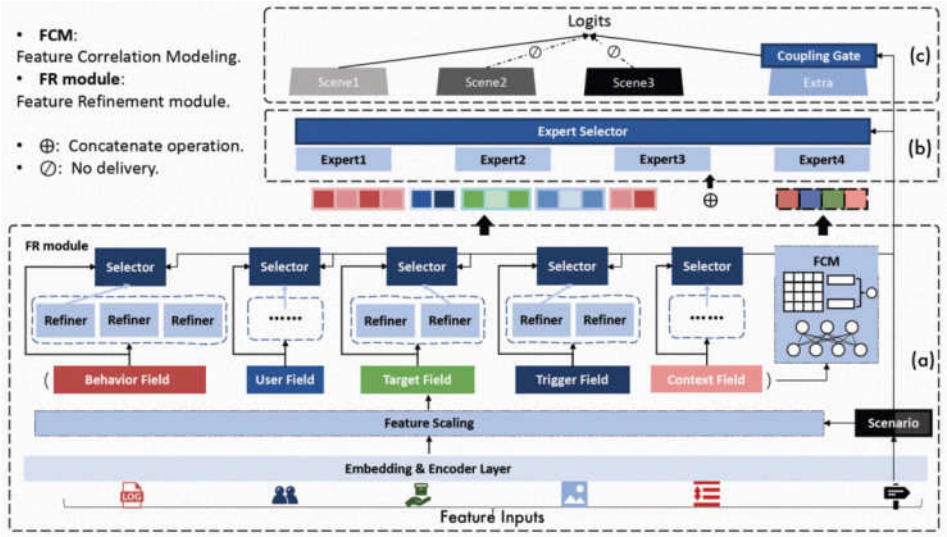


图 4　Maria 整体框架

Feature Scaling

考虑到不同场景的异质性，例如不同的 Query 模态，很明显不同类型的特征在不同场景中具有不同的重要度。例如，视觉特征在拍立淘（拍照搜索）场景中通常比在找相似（商品搜索）场景中发挥更重要的作用。因此，FS 模块被设计为根据场景指示符来缩小或放大每个特征。影响力强的特征被放大，相反不重要的特征被缩小，其中 Scenario 特征将作为场景 Indicator 来指导特征的放缩。我们参考信号调试的方式，将放缩的比例限定在 [0,] 之间，不同场景下各个特征对应的放缩比例是根据场景特征通过 Gate 网络（浅层全连接层网络）计算得到的。

Feature Refinement

考虑到相同特征在不同场景的语义表征存在差异，我们设计了一个特征修正微调模块，该模块利用 Selector 门控网络自动的生成每个 refiner 的权重来支持场景差异化的高级语义编码。具体来说，我们为每个特征字段设置了一组特征微调器（Refiner），每个微调器都是一个浅的完全连接层。如图 5 所示，对于 Refiner 的选择是以场景感知的方式自动进行的。

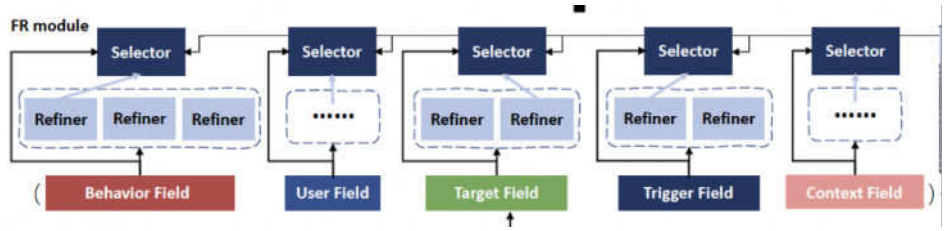


图 5　FR 结构

上图中首先将 Feature Scaling 模块的输出划分为不同的 Field，即意义相近的一组特征。Field 的划分根据特征含义人为划分，例如 Behavior Field 是用户的交互历史序列特征；User Field 是由描述用户信息的特征组成。每一组特征通过 Refiner 的微调输入到下一层。其中微调过程是将 Field 特征向量和不同的 Refiner 做矩阵乘法，每个 Refiner 是一个矩阵。Selector 作用是根据每个 Field 和 Scenario Indicator 生成对 Refiner 的选择权重。具体来说，Selector 是全连接神经网络，将场景特征向量和每个 Field 的特征向量拼接输入到 Selector 中，输出向量每一位作为相应 Refiner 的选择权重。为了使权重的分布相对离散，突出选择使用权重更大的 Refiner，减轻权重更小的 Refiner 的影响，增强场景之间的解耦能力避免不同场景选择相同的 Refiner，我们对 Selector 的输出使用了 Gumbel softmax 处理。

Feature Correlation Modeling

在 FR 中将输入拆分成不同的 Field，这导致 Field 之间的交互信息缺失。于是我们利用 Feature Correlation 模块显式建模 Field 之间的交互信息。如下图 6 中的红框所示：

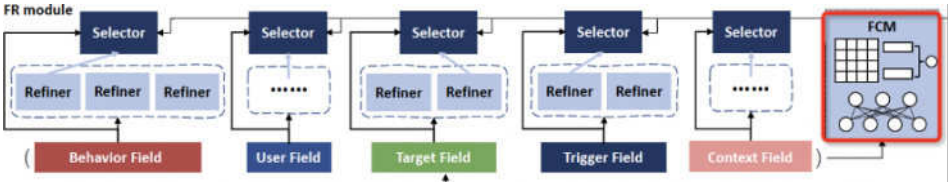


图 6　FCM 模块结构

将不同 Field 通过不共享的全连接神经网络映射为相同长度的向量然后两两之间计算相关性，并将结果输出到下游

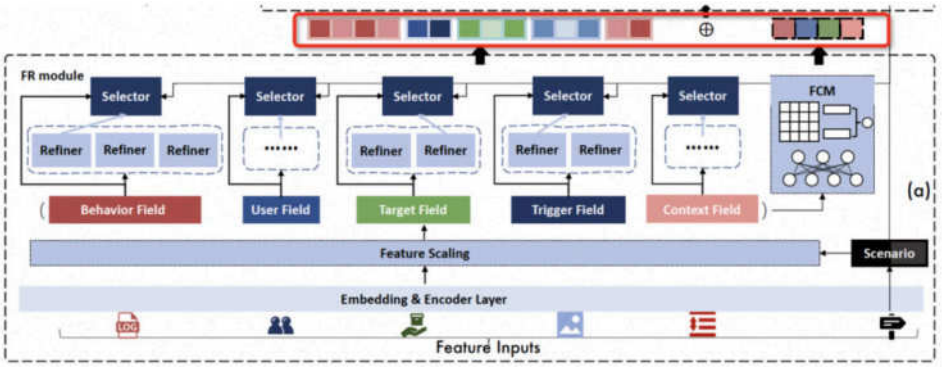


图 7　特征自适应学习输送到下游网络层

如上图 7 所示，红框中的左侧部分是经过 Refiner 微调的输入，而右边则是 Field 之间的 Correlation 信息。然后将 Refiner 微调的输入和 Correlation 信息拼接之后输入到下面一个部分，即网络层。

网络层和预测层

在网络层和预测层，我们分别采用了 MoE 结构如下图 8（b）和多塔结构如下图 8（c）：

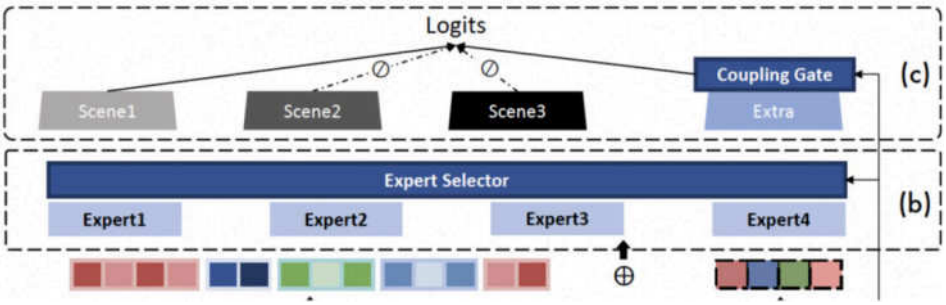


图 8　网络层 & 预测层

每个 Scene Tower 单独服务于对应场景的样本，我们增加一个 Extra Tower 来增加场景间的信息共享，例如用户在 Scene1 中表现的兴趣可以被 Scene2 捕捉。Coupling Gate 用来计算 Extra Tower 输出的权重，计算方式是用当前输入的样本场景特征和其他场景特征计算相似度并取平均，这样就得得到了 Extra Tower 的输出权重。通过增加场景间的信息共享，可以利用其他场景的信息为当前场景的预测提供帮助。进而提升所有场景的预测表现。另外，由于 Refiner 本身可以起到一个降维

的作用，可以大大减少了网络层和预测层的参数量。和线上单模型对比，我们的参数量增加了 7% 左右。模型训练的损失函数采用的是简单的交叉熵损失。

4. 实验分析

为了证明该方法的有效性，我们在多模态搜索广告多场景数据集和公开 Ali-CCP 数据集进行了实验，Ali-CCP 数据集的统计结果如表 1 所示。而 Alimama 数据集选择的是真实业务数据 30 天的日志，包含 VS（Visual Search）、SS（Similar Search）和 IS（Interest Search）。

Datasets	Ali-CCP			
Scenarios	S1	S2	S3	All
#User	91,488	2,612	154,024	244,397
#Product	535,711	198,651	537,937	538,376
#Impression	32,236,951	639,897	52,439,671	85,316,519
#Click	1,291,063	28,022	1,998,618	3,317,703

表 1　数据集统计信息

实验结果

如表 2 所示，与现有最佳的基线模型进行对比，本文提出的模型在推荐数据集 Ali-CCP 和 Alimama 多模态搜索数据集上均有比较稳定的表现。不仅在 AUC 指标上取得了提升，同时缓解了场景间不平衡的问题，能够较为有效的缓解多场景学习中出现的跷跷板现象。

Dataset	Scenarios	Models							
		Hard Sharing	Shared Bottom	MulANN	MMoE	PLE	STAR	$AEMS^2$	MARIA
Alimama	VS	0.7415*	0.7318*	0.7423*	0.7441*	0.7421*	0.7323*	0.7289*	**0.7473**
	SS	0.6777*	0.6842*	0.6792*	0.6779*	0.6840*	0.6724*	0.6703*	**0.6927**
	IS	0.7131*	0.7159*	0.7129*	0.7139*	0.7126*	0.6925*	0.6994*	**0.7178**
	Total Impr.	+0.0255	+0.0259	+0.0234	+0.0219	+0.0191	+0.0606	+0.0592	–
Ali-CCP	S1	0.5747*	0.5528*	0.5625*	0.5740*	0.5744*	0.5524*	0.5773*	**0.5869**
	S2	0.5912*	0.5606*	0.5779*	0.5789*	0.5939*	0.5826*	0.5957*	**0.6114**
	S3	0.5888*	0.5710*	0.5754*	0.5768*	0.5927*	0.5975*	0.5916*	**0.6077**
	Total Impr.	+0.0513	+0.1216	+0.0902	+0.0763	+0.0450	+0.0735	+0.0414	–

表 2　实验结果

消融实验

如表 3 所示，我们通过详细消融实验进一步证明了 Maria 中的各个模块的均能带来

正向效果。其中各消融模块的含义如下：

（1）ST：预测层中 Shared Tower 结构

（2）FS, FR, FCM：分别对应 Feature Scaling、Feature Refinement、Feature Correlation Modeling

（3）NL：对应网络层 Network Layer

（4）GS：Gumbel Softmax 结构

Models	Alimama			
	VS	SS	IS	Total Gain
w/o ST	0.7461	0.6873	0.7173	-0.0071
w/o FCM	0.7445	0.6922	0.7174	-0.0037
w/o FR	0.7466	0.6903	0.7150	-0.0059
w/o FS	0.7462	0.6909	0.7141	-0.0066
w/o NL	0.7447	0.6877	0.7169	-0.0085
w/o GS	0.7455	0.6912	0.7165	-0.0046
MARIA	**0.7473**	**0.6927**	**0.7178**	–

Models	Ali-CPP			
	S1	S2	S3	Total Gain
w/o ST	0.5792	0.5986	0.5952	-0.0330
w/o FCM	0.5686	0.5893	0.5877	-0.0604
w/o FR	0.5709	0.5891	0.5873	-0.0587
w/o FS	0.5844	0.6005	0.5983	-0.0228
w/o NL	0.5796	0.5989	0.5969	-0.0306
w/o GS	0.5857	0.6053	0.5997	-0.0153
MARIA	**0.5869**	**0.6114**	**0.6077**	–

表 3　消融实验结果

Case Study

我们将 FR 结构中的 Refiner 的权重进行可视化，以进一步探究在多场景建模中场景差异性对不同 Refiner 的自适应选择是否达到我们的预期。我们随机采样 Alimama 数据集中 1000 个样本中 User Field 和 User behavior Field 中的第一个 Refiner 的权重，分别画出三个场景的分布图：

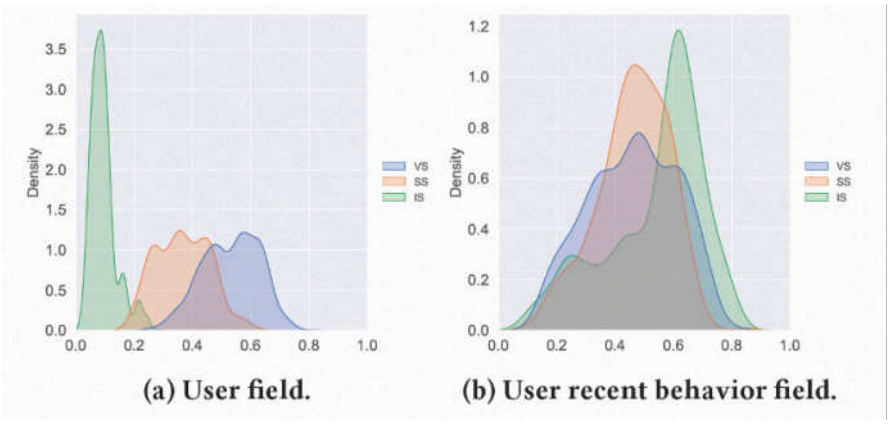


图 9　FR 中 Refiner 的权重分布可视化

从上图 9（a）可以看到，三个场景的分布具有明显的差异，体现了用户侧表征在三个场景中的空间位置存在明显不同。具体来看，SS 和 IS 场景具有一定的相似性，但 VS 场景和另外两个场景存在明显差异，几乎没有交集。结合上文中我们针对 Alimama 三个场景的描述，由于 VS 场景的搜索 Query 和其他场景不同，因此用户的意图和偏好与另外两个场景差异更为明显。而通过 FR 结构，我们可以较好的达到端到端微调的作用，实现特征层面的场景间解耦合。

5. 总结

本文聚焦在多模态搜索广告下的多场景点击率预估建模问题。与现有方法相比，我们提出的 Maria 可以端到端的实现特征的自适应学习。在该模型底层，我们设计了三个主要部分来保证有区分力的特征学习：Feature Scaling（FS）、Feature Refinement（FR）、Feature Correlation Modeling（FCM）。实验结果表明，这种方法在各个场景均有正向表现，解决了现有研究方案的效果不平衡问题。同时 Maria 方法具备通用性，可以有效的迁移到不同的网络结构和建模任务中。

参考文献

[1] Multi-Scenario Ranking with Adaptive Feature Learning, arxiv: https://arxiv.org/abs/2306.16732

[2] Xiang-Rong Sheng, Liqin Zhao, Guorui ZhouGuorui Zhou, , Xinyao Ding, Binding Dai, Qiang Luo, Siran Yang, Jingshan Lv, Chi Zhang, Hongbo Deng, Xiaoqiang Zhu:One Model to Serve All: Star Topology Adaptive Recommender for Multi-Domain CTR Prediction. CIKM 2021

[3] Jiaqi Ma, Zhe Zhao, Xinyang Yi, Jilin Chen, Lichan Hong, Ed H. Chi:Modeling Task Relationships in Multi-task Learning with Multi-gate Mixture-of-Experts. KDD 2018

[4] Hongyan Tang, Junning Liu, Ming Zhao, Xudong Gong:Progressive Layered Extraction (PLE): A Novel Multi-Task Learning (MTL) Model for Personalized Recommendations. RecSys 2020RecSys 2020

[5] Xinyu Zou, Zhi HuZhi Hu, , Yiming Zhao, Xuchu Ding, Chenliang Li, Automatic Expert Selection for Multi-Scenario and Multi-Task Search. SIGIR 2022Xuchu Ding, Chenliang Li, Aixin Sun

HC^2：基于混合对比学习的多场景广告预估建模

作者：曲溪

1. 摘要

多场景广告预估建模旨在利用多场景的数据来训练统一的预估模型，以提高各个场景的效果。尽管现有研究方法在推荐 / 广告领域已取得了不错的提效，但现有的建模方式仍然缺乏跨场景关系的考虑，从而导致模型学习能力的限制和场景间相互关系建模的困难。在本文中，我们提出了一种用于多场景广告预估建模的混合对比学习方法 HC^2。为增强跨场景数据相互关系的建模，我们精心设计了一种混合对比学习方法来协助模型捕获多个场景之间的共性和差异。该方法的核心包括两个精心设计的对比损失，即场景通用对比损失和场景个性化对比损失，其目的分别是捕获场景通用知识和场景特定知识。此外，为了使对比学习适应复杂的多场景预估背景，我们提出了一系列改进方案。实验效果表明 HC^2 方法能有效的提升多场景下广告预估的能力。基于该项工作整理的论文已发表在 CIKM 2023，欢迎阅读交流。

论文： Hybrid Contrastive Constraints for Multi-Scenario Ad Ranking

作者： Shanlei Mu, Penghui Wei, Wayne Xin Zhao, Shaoguo Liu, Liang Wang, Bo Zheng

下载（点击↓阅读原文）： https://dl.acm.org/doi/abs/10.1145/3583780.3614920

2. 引言

多场景推荐 / 广告预估旨在利用多场景的用户行为数据来训练预估模型（例如：CTR 模型、CVR 模型）服务于各个应用场景。作为优势，多场景建模可以利用更加丰富的数据缓解每个场景的数据稀疏问题，进而有效的提升不同场景下的推荐 / 广告投放效果。

多场景广告预估建模的关键在于捕捉不同场景下用户行为的**共性**和**差异**，以更加充分的利用多场景的数据提升建模效果。为此，结合场景共享和场景独有的神经网络结构（例如：SharedBootom[1]、MMoE[2]）被广泛的应用在多场景广告预估建模中。一般来说，场景共享的网络结构使用所有场景的数据进行优化以捕捉多个场景之间的共性，而场景独有的网络结构利用对应单一场景的数据进行优化以学习场景特定

的知识，来建模出不同场景的差异。此外，一些参数动态生成模型也被提出，例如STAR[3]、APG[4]，这些方法使用共享的网络结构来生成场景特有的网络参数达到多场景建模的目的，其也可以被看作结合场景共享和场景独有的神经网络结构。

尽管这些方法已经被证明在推荐 / 广告领域取得不错的提效，但是我们认为多场景建模中的两个关键问题还没有被彻底解决：

- **网络结构学习能力的局限**：通常结合场景共享和场景独有的神经网络结构仅根据多个场景下的特征到标签的映射关系进行优化，这种单纯有监督的信号受到训练数据质量和可用性的高度限制，同时对于集成了多场景结构的复合架构存在优化效率低的问题。
- **场景相互关系建模的困难**：由于我们难以直接获得学习不同场景间关系的直接信号，所以难以精确建模出不同场景下共性和差异。特别是跨场景的数据相互关系非常复杂，在不同的样本或上下文中会显示出不同的关联模式。

为了解决上述问题，我们考虑借用对比学习的思想来提升多场景的广告建模。对比学习旨在通过从不同的视角对比正样本和负样本的表示来增强神经网络的表达能力，在许多领域都取得了巨大的成功。通过利用不同场景下样本间显式或隐式的相关模式，它可以派生出丰富的自监督信号来增强模型学习能力（对应于第一个问题）。同时，它是在样本层面进行的，这样我们就可以设计出更为灵活的的对比损失来分别建模优化不同场景间的共性和差异（对应第二个问题）。结合这两个优势，我们可以开发出更有效的多场景广告建模方法。

基于此，我们在已有的结合场景共享和场景独有的神经网络结构上提出了一种基于混合对比学习的多场景广告预估建模方法（**H**ybrid **C**ontrastive **C**onstraints for Multi-Scenario Ad Ranking，**HC^2**）。HC^2 设计了两种不同的对比损失：**场景通用对比损失**和**场景个性化对比损失**，分别用于帮助模型更好的捕获场景通用的知识和场景独有的知识，同时为了使对比学习适应复杂的多场景广告建模任务，我们设计了一系列具体的策略和技术改进，包括对比样本的选择和扩展，对比损失的加权等等。该方法可看作是现有多场景模型的通用改进，可以被应用到各种结合场景共享和场景独有的神经网络结构中来提升多场景广告预估建模效率。

3. 方法

如图 1 所示，我们的方法建立在结合场景共享和场景独有的神经网络结构上，并引入了两种主要的对比学习损失，即场景通用对比损失和场景个性化对比损失。场景通用

对比损失应用于场景共享的结构上，以捕捉多个场景之间的共性。场景个性化对比损失应用于场景特定的结构上，以学习特定场景的知识。这两种对比损失可以有效地探索多场景相互关系，以提高多场景中的广告建模。

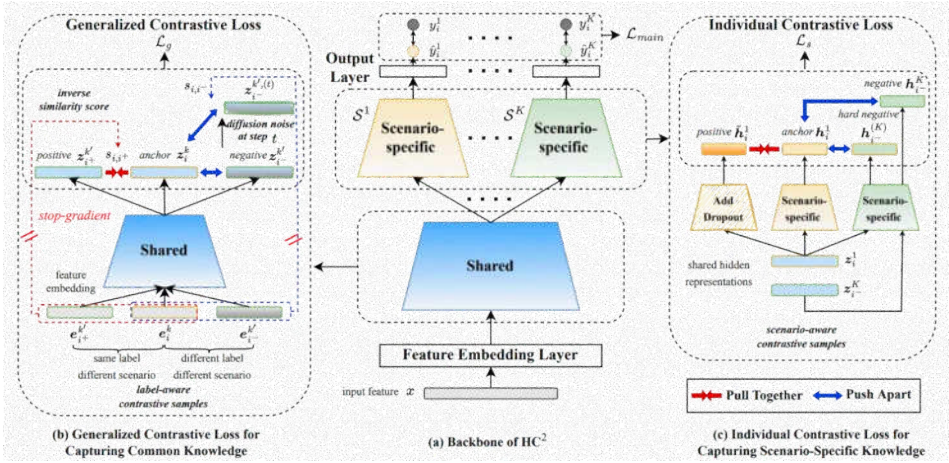


图 1 提出方法的总体结构

3.1 任务定义 & 基础结构

我们首先形式化多场景广告预估任务，让 $\mathcal{X}$ 和 $\mathcal{Y}$ 分别表示特征空间和标签空间，特征空间包含用户特征、广告特征和上下文特征等，标签空间是代表用户行为的二元标签，例如 CTR 预估任务中的 *{点击，未点击}*，CVR 预估任务中的 *{转化，未转化}*。

考虑 K 个不同的广告投放场景 $\{S^1 \cdots S^K\}$，其中场景 k 的样本可以表示为 $S^k = \{(x_i^k, y_i^k)\}_{i=1}^{|S^k|}$，$x_i^k \in \mathcal{X}$ 代表第 i 个样本的特征，$y_i^k \in \{0, 1\}$ 代表样本标签。多场景广告建模旨在利用所有场景的样本来训练一个模型 $\hat{y}_i^k = p_\theta(x_i^k)$ 用于预测标签 y_i^k，通常采用交叉熵损失对参数 θ 进行优化：

$$\mathcal{L}_{main} = \sum_{k=1}^{K} \sum_{i=1}^{|S^k|} \ell(y_i^k, \hat{y}_i^k)$$

本文提出的方法 HC^2 是对结合场景共享和场景独有的神经网络结构用于多场景广告建模的通用提升方法。不失一般性，我们选择代表性的 SharedBottom 结构作为基础结构来介绍本文的方法。其主要包含四个部分：特征编码，场景共享的网络结构，场景独有的网络结构，输出层。其中特征编码主要是将样本的原始输入 x_i^k 映射成特

征向量 $\boldsymbol{e}_i^k$，场景共享的网络结构是 L 层神经网络 $f(\cdot)$ 把特征向量 $\boldsymbol{e}_i^k$ 映射成场景共享的中间层表示 $\boldsymbol{z}_i^k$，场景独有的网络结构是 L' 层神经网络 $g(\cdot)^{(k)}$ 把共享的中间层表示 $\boldsymbol{z}_i^k$ 映射成场景独有的隐层表示 $\boldsymbol{h}_i^k$，最后输出层将场景独有的隐层表示 $\boldsymbol{h}_i^k$ 输出为预测的标签信息 $\hat{y}_i^k$。

3.2 场景通用对比损失 Generalized Contrastive Loss

已有的多场景广告建模预估方法主要依靠场景共享的结构从多个场景中捕获共同知识，但它仅通过端到端的特征到标签映射进行优化。对于这部分共享的结构，在训练过程中缺乏目标指导或约束使得难以精确的捕捉到场景共有的知识。为了解决这个问题，我们设计了一种改进的对比学习方法来帮助共享的结构更好地捕获场景通用知识。

3.2.1 对比损失介绍

我们设计的场景通用对比损失在经典的对比学习范式进行了两点主要的改进：1）对比样例的扩展；2）引入对比权重，形式化如下：

$$\mathcal{L}_g = -\mathbb{E}[\frac{s_{i,i^+}\exp(\boldsymbol{z}_i \cdot \boldsymbol{z}_{i^+}/\tau)}{s_{i,i^+}\exp(\boldsymbol{z}_i \cdot \boldsymbol{z}_{i^+}/\tau) + \sum_{i^- \in \mathcal{N}} s_{i,i^-}\exp(\boldsymbol{z}_i \cdot \boldsymbol{z}_{i^-}/\tau)}]$$

在标准的对比损失中，样本正例通常通过自身数据增强来构造，样本负例通过随机采样构造。为了构建更适合多场景间提取通用知识的对比学习，我们引入了标签感知的对比样本和扩散噪声增强的对比样本，作为另一个提升，原始的对比损失平等地对待每一个对比样例，我们进一步提出逆相似性分数加权机制来减少伪对比样例的影响，接下来我们详细地介绍这两部分。

3.2.2 对比样本构建

样本感知的对比样本：首先我们提出利用标签的信息来为场景通用对比损失构建对比样例。具体地，给定一个来自场景 S^k 的样本 x_i^k，从其他场景 $S^{k'}(k' \neq k)$ 选取和 x_i^k 具有相同标签信息的样本 $x_{i^+}^{k'}$ 作为对比正例，从其他场景选取和 x_i^k 具有相反标签信息的样本 $x_{i^-}^{k'}$ 作为对比负例。通过拉近 x_i^k 和 $x_{i^+}^{k'}$ 的共享中间层表示和拉远 x_i^k 和 $x_{i^-}^{k'}$ 的共享中间层表示，场景通用对比损失鼓励场景共享的结构更好的捕捉不同场景间属于相同标签信息的通用知识。

扩散噪声增强的对比样本：上述提到的对比样本局限在从训练集中局部观测到的候选集合，由于对比学习中存在的各向异性问题，对比负例的中间层表示局限在由共享的

网络结构生成的狭窄表示空间内，并不能反映出整体样本空间的信息。考虑到这个问题，我们受到最近扩散模型进展的启发，提出加入扩散噪声来生成更多的泛化、多样的对比负样本。具体来说，给定负样本 $\boldsymbol{z}_{i^-}$，我们遵循标准的前向扩散过程，通过迭代向样本添加小的高斯噪声，产生一系列噪声样本 $\boldsymbol{z}_{i^-}^{(1)}, \boldsymbol{z}_{i^-}^{(2)}, \cdots, \boldsymbol{z}_{i^-}^{(T)}$。步长由方差表 $\{\beta_t \in (0,1)\}_{t=1}^{T}$ 控制，前向扩散过程可以表示如下：

$$q(\boldsymbol{z}_{i^-}^{(t)}|\boldsymbol{z}_{i^-}^{(t-1)}) = \mathcal{N}(\boldsymbol{z}_{i^-}^{(t)}; \sqrt{1-\beta_t}\boldsymbol{z}_{i^-}^{(t-1)}, \beta_t \mathbf{I})$$

为了生成新的负样本，我们首先对不同的时间步长 t 进行采样，然后使用上述公式来获得相应的样本。这些噪声增强的样本更加多样化，来自更广阔的样本空间，从而提高了底层模型的泛化能力。

3.2.3 逆相似性加权

在对比学习中，构建的对比样本很可能是伪正例或伪负例，从而影响对比学习的效果。在我们的设置中，尽管来自不同场景的两个正样本具有相同的标签，但它们的原始特征可能存在很大差异，这将导致产生伪对比正例问题。类似地，伪对比负例问题也会发生。

为解决这一问题，我们根据样例间的特征相似性设计了一种自适应对比损失加权方法。其基本思想是，正样本对应该具有更大的特征相似性，而负样本对应该具有较小的特征相似性，如果情况相反，对应的样本对可能就是伪正 / 负例。基于这个想法，对于一对对比样本 x_i 和 x_j，我们计算了它们特征 embedding 相似性分数的倒数，并使用它来设置作为场景通用对比损失的权重：

$$s_{i,j} = \frac{1}{\boldsymbol{e}_i \cdot \boldsymbol{e}_j}$$

这种加权策略可以自适应地分配特定的权重，并在训练过程中动态更新这些权重。从最后的场景通用对比损失中可以看出，当出现伪对比正样本时，逆相似性得分 $s_{i,j}$ 会产生一个更大的值，这个值会减小优化这部分损失所对参数造成的影响。相似地，当出现伪对比负样本时，倒数相似性得分 $s_{i,j}$ 将是一个更小的值，因此对应的对比样本对对在优化目标中的贡献也会降低。

3.3 场景个性化对比损失 Individual Contrastive Loss

除了捕捉不同场景间的通用知识外，对于不同场景个性化的差异建模对于提升多场景

广告建模也至关重要。现有的多场景广告建模方案通常基于对应场景的数据来直接优化场景独有的网络结构。为了更好的利用跨场景的信息，我们提出了场景个性化的对比损失来提升对于场景独有网络结构的优化。

3.3.1 场景感知的对比样例

场景个性化对比损失旨在加强场景独有的网络结构对于对应场景信息的捕捉和建模，为了实现这一点，我们选取对场景独有的隐层表示（即 $\boldsymbol{h}_i^k$）进行对比学习，来减少不同场景之间的相关性。接下来我们介绍如何选取对比样本来达到这一目的。

对比正例构建：为了更好的捕捉场景独有的知识，我们考虑通过对原始样本进行数据增强来获取对比正样本 $\tilde{\boldsymbol{h}}_i^k$。本文我们采用了添加 dropout 噪声的方式来对数据进行增强以获取对比正样本。具体来说，我们在场景独有的网络结构的每一层添加 Dropout 层来生成对比正样本 $\tilde{\boldsymbol{h}}_i^k$。

对比负例构建：对于对比负样本，我们首先从其他场景中选取样本，然后生成其对应的场景独有的隐层表示 $\boldsymbol{h}_{i^-}^{k'}$，其中 $k' \neq k$。通过拉远 $\boldsymbol{h}_i^k$ 和 $\boldsymbol{h}_{i^-}^{k'}$，我们希望拉大场景 S^k 和场景 $S^{k'}$ 之间的差异，从而使得场景独有的网络结构专注捕捉对应场景本身的信息。除此之外，为了构建信息量更加充足的对比样本，我们进一步提出跨场景的编码策略来生成对比负例。基本的思想是，我们首先选取和锚定样本 $\boldsymbol{x}_i^k$ 属于同一场景的样本 $\boldsymbol{x}_{i^-}^k$，接着我们将其生成的共享中间层表示 $\boldsymbol{z}_{i^-}^k$ 喂进其他场景的场景独有的网络结构中，依照这种方式来生更强的对比负例：

$$\boldsymbol{h}_{i^-}^{(k')} = g^{(k')}(\boldsymbol{z}_{i^-}^k)$$

其中 k 和 k' 属于两个不同的场景，所以我们称之为跨场景编码。最后我们得到如下的场景个性化对比损失：

$$\mathcal{L}_s = -\mathbb{E}\Big[\frac{\exp\left(\boldsymbol{h}_i^k \cdot \tilde{\boldsymbol{h}}_i^k/\tau\right)}{\exp\left(\boldsymbol{h}_i^k \cdot \tilde{\boldsymbol{h}}_i^k/\tau\right) + \sum_{i^- \in \mathcal{N}}\left[\exp\left(\boldsymbol{h}_i^k \cdot \boldsymbol{h}_{i^-}^{k'}/\tau\right) + \exp\left(\boldsymbol{h}_i^k \cdot \boldsymbol{h}_{i^-}^{(k')}/\tau\right)\right]}\Big]$$

3.4 模型优化

最后我们通过端到端的方式将主损失、场景通用对比损失和场景个性化对比损失联合优化：

$$\mathcal{L} = \mathcal{L}_{main} + \lambda_1 \mathcal{L}_g + \lambda_2 \mathcal{L}_s$$

其中 λ_1 和 λ_2 是调控场景通用对比损失和场景个性化对比损失影响的参数。

4. 实验

4.1 实验设置

数据集：我们选取了两个多场景数据集进行离线实验，其中一个是公开数据集 AliExpress[5]，另一个是我们从阿里妈妈真实广告业务构建的数据集 Ali-ads。

评测任务和指：我们选取了 CTR 预估和 CTCVR 预估两个任务，评测指标为离线 AUC。

4.2 离线实验结果

表 1 记录了两个数据集上我们提出的方法和基线方法的效果对比。可以看到，相较于其他模型，HC^2 在两个数据集上都有更好的效果表现，除此之外，在数据更加稀疏的场景上（例如：AliExpress 的 NL 和 US 场景），HC^2 比其他方法能取得更大的提升，我们判断这是因为通过两种对比损失的约束，我们的方法能更好的利用多场景之间的关系来缓解数据稀疏问题。表 2 记录了消融实验的结果，可以看到设计的各个模块都有助于效果的提升，其中场景通用对比损失对效果提升的贡献最大。

Dataset	Scenario	Metric	BaseDNN	CFM	MultiDNN	SharedBottom	MMoE	HMoE	STAR	M2M	FOREC	CausalInt	HC^2
AliExpress	NL	AUC_{CTR}	0.7229	0.7225	0.7245	0.7251	0.7247	0.7235	0.7223	0.7218	0.7239	0.7235	**0.7398**
		AUC_{CTCVR}	0.8568	0.8469	0.8577	0.8569	0.8490	0.8592	0.8544	0.8577	0.8580	0.8566	**0.8637**
	FR	AUC_{CTR}	0.7246	0.7253	0.7264	0.7276	0.7253	0.7294	0.7258	0.7298	0.7269	0.7271	**0.7380**
		AUC_{CTCVR}	0.8749	0.8742	0.8779	0.8800	0.8797	0.8787	0.8786	0.8766	0.8760	0.8792	**0.8804**
	US	AUC_{CTR}	0.7114	0.7086	0.7112	0.7097	0.7035	0.7080	0.7072	0.7064	0.7119	0.7085	**0.7245**
		AUC_{CTCVR}	0.8673	0.8697	0.8708	0.8703	0.8721	0.8705	0.8701	0.8722	0.8703	0.8711	**0.8734**
	ES	AUC_{CTR}	0.7286	0.7284	0.7293	0.7297	0.7285	0.7301	0.7309	0.7277	0.7282	0.7302	**0.7374**
		AUC_{CTCVR}	0.8871	0.8883	0.8888	0.8893	0.8878	0.8893	0.8869	0.8889	0.8868	0.8891	**0.8936**
	RU	AUC_{CTR}	0.7335	0.7332	0.7324	0.7327	0.7346	0.7331	0.7347	0.7331	0.7336	0.7342	**0.7421**
		AUC_{CTCVR}	0.9086	0.9086	0.9087	0.9050	0.9048	0.9089	**0.9091**	0.9069	0.9086	0.9080	0.9089
Ali-ads	S1	AUC_{CTR}	0.7211	0.7224	0.7226	0.7257	0.7243	0.7219	0.7262	0.7248	0.7249	0.7249	**0.7352**
		AUC_{CTCVR}	0.8722	0.8776	0.8790	0.8801	0.8815	0.8863	0.8827	0.8876	0.8815	0.8812	**0.8979**
	S2	AUC_{CTR}	0.6896	0.6902	0.6898	0.6929	0.6905	0.6918	0.6908	0.6921	0.6915	0.6919	**0.7023**
		AUC_{CTCVR}	0.8530	0.8556	0.8618	0.8633	0.8642	0.8707	0.8650	0.8702	0.8698	0.8699	**0.8824**
	S3	AUC_{CTR}	0.6736	0.6742	0.6779	0.6789	0.6799	0.6807	0.6790	0.6791	0.6817	0.6799	**0.6866**
		AUC_{CTCVR}	0.8685	0.8902	0.8926	0.8894	0.8912	0.8995	0.8930	0.9019	0.8974	0.8999	**0.9131**
	S4	AUC_{CTR}	0.6558	0.6654	0.6667	0.6695	0.6678	0.6669	0.6705	0.6697	0.6696	0.6704	**0.6809**
		AUC_{CTCVR}	0.8335	0.8400	0.8424	0.8428	0.8537	0.8581	0.8552	0.8570	0.8544	0.8529	**0.8832**
	S5	AUC_{CTR}	0.7087	0.7100	0.7075	0.7108	0.7114	0.7081	0.7106	0.7112	0.7124	0.7117	**0.7282**
		AUC_{CTCVR}	0.8661	0.8689	0.8700	0.8719	0.8742	0.8743	0.8745	0.8745	0.8753	0.8746	**0.8901**

表 1　在两个数据集上不同方法的效果对比

Variants	NL	FR	US	ES	RU
HC^2	**0.7398**	**0.7380**	**0.7245**	**0.7374**	**0.7421**
w/o g-loss	0.7257	0.7283	0.7103	0.7301	0.7371
w/o noise	0.7288	0.7368	0.7138	0.7344	0.7356
w/o weight	0.7323	0.7338	0.7175	0.7368	0.7402
w/o s-loss	0.7347	0.7356	0.7183	0.7367	0.7362
SharedBottom	0.7251	0.7276	0.7097	0.7297	0.7327

表 2　消融实验结果

应用 HC^2 到其他结构：我们的方法 HC^2 可以应用于其他多场景广告建模方法中。因此，在这一部分中，我们进行了一个实验来检验我们的方法是否可以为其他网络结构带来效果提升。具体来说，我们将我们的方法应用于多场景广告建模方法 MMoE、HMoE、STAR 和 M2M 上。如图 3 所示，所提出的方法可以持续提高这些网络结构的效果，进一步验证了所提出方法的有效性。

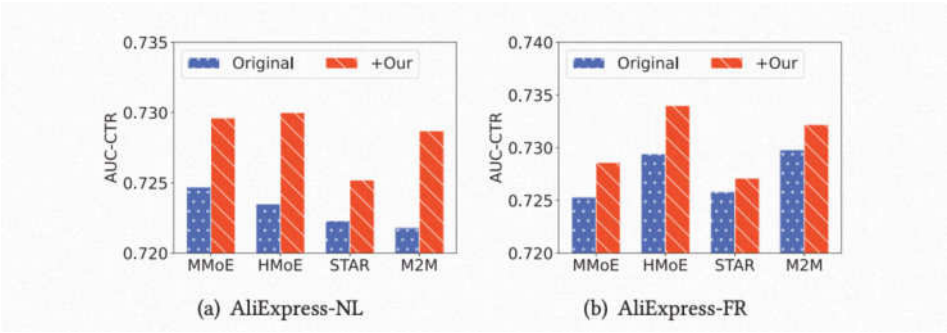


图 2　应用 HC^2 到其他结构效果对比

隐藏层表示可视化：为了更好的理解场景通用对比损失对多场景模型结构的影响，我们在图 3 中可视化了场景共享结构中学习得到的隐藏层表示。具体来说，我们使用 T-SNE 将隐藏层表示的维度减少到二维空间，然后进行绘制。从图 3 我们可以观察到通过 SharedBottom 学习到的表示落在一个狭窄的空间中，而 HC^2 学习到的表示明显表现出更均匀的分布。这可以通过表示学习中的一致性（uniformity）概念来解释，一致性反映了表征保存信息的能力。如果表征大致均匀地分布在平面上，则它们可以保留尽可能多的数据信息。我们推测，所提出的场景通用对比损失可以有效的帮助模型从多个场景中保留更多的通用知识。

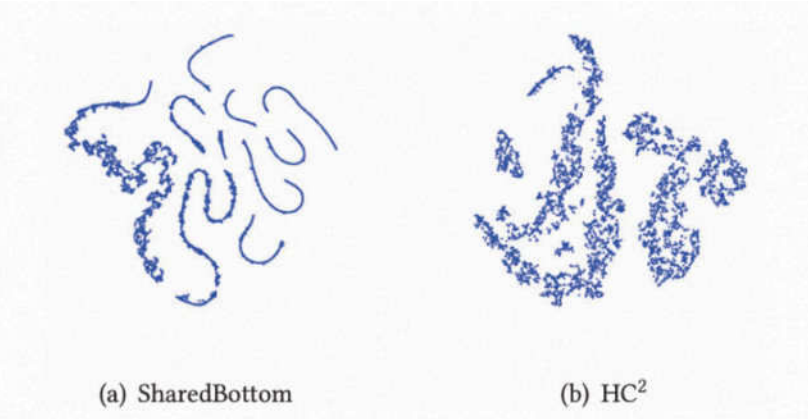


图 3　隐藏层表示可视化

// 更多实验结果请参考论文原文。

4.3 线上实验

我们将提出的 HC^2 方法部署在了阿里妈妈外投广告相关业务进行线上实验，线上整体取得了 CVR +2.51%，GMV +3.72% 的效果提升。

5. 总结

本文提出了一种基于混合对比学习的多场景广告预估建模方法 HC^2。主要的技术贡献在于两个对比损失的设计，即场景通用对比损失和场景个性化对比损失，旨在捕获多场景下的通用知识和特定场景知识。同时为了使对比学习更适应多场景广告建模任务，我们设计了一系列特定策略来改进对比样本和对比权重。通过有效地建模跨场景数据的相互关系，我们的方法可以普遍应用于各种多场景广告预估建模任务。离线评估和在线测试的广泛实验证明了方法的有效性。

参考文献

[1] Rich Caruana. 1998. Multitask Learning. In Learning to Learn. 95 – 133.
[2] Jiaqi Ma, Zhe Zhao, Xinyang Yi, Jilin Chen, Lichan Hong, and Ed H. Chi. 2018. Modeling Task Relationships in Multi-task Learning with Multi-gate Mixture-ofExperts. In KDD. 1930 – 1939.
[3] Xiang-Rong Sheng, Liqin Zhao, Guorui Zhou, Xinyao Ding, Binding Dai, Qiang Luo, Siran Yang, Jingshan Lv, Chi Zhang, Hongbo Deng, and Xiaoqiang Zhu. 2021. One Model to Serve All: Star Topology Adaptive Recommender for Multi-Domain CTR Prediction. In CIKM. 4104 – 4113.

[4] Bencheng Yan, Pengjie Wang, Kai Zhang, Feng Li, Jian Xu, and Bo Zheng. 2022. APG: Adaptive Parameter Generation Network for Click-Through Rate Prediction. CoRR abs/2203.16218 (2022).

[5] Pengcheng Li, Runze Li, Qing Da, Anxiang Zeng, and Lijun Zhang. 2020. Improving Multi-Scenario Learning to Rank in E-commerce by Exploiting Task Relationships in the Label Space. In CIKM. 2605 – 2612.

AdaSparse: 自适应稀疏网络的多场景 CTR 预估建模

作者：宝亘、细麦、烈文

摘要

CTR(Click-through rate) 预估一直是推荐 / 广告领域重要技术之一。近年来，通过统一模型来服务多个场景的预估建模已被证明是一种有效的手段。当前多场景预估技术面临的挑战主要来自两方面：1）跨场景泛化能力：尤其对稀疏场景；2）计算复杂度：在线计算和存储资源有限情况下如何实现多场景模型构建。针对这两个挑战，本文提出 AdaSparse，一种通过自适应学习稀疏网络的方法来实现多场景 CTR 预估建模。该方法为每个场景学习一个子网络结构，在计算成本增加有限的条件下，自适应平衡场景间的共性和特性，提升了模型跨场景的泛化能力。我们设计了一个辅助网络来衡量主网络每一层神经元对每个场景的重要度，并通过网络裁剪技术对不重要 / 冗余的神经元进行裁剪，最终为每个场景生成一个专属的子网络进行预估。此外，我们提供了一种灵活的稀疏正则方法来更好的控制网络稀疏程度，使模型在应用上更具有灵活性。AdaSparse 已在阿里妈妈外投广告系统上线，支撑了多个投放业务的 CTR 预估，整体取得了 CTR+4.63%，CPC-3.82% 的效果（CPC：点击成本，越低越好）。该工作核心部分已收录于 CIKM2022，本文将对该方法的背景及实现进行详细介绍，欢迎阅读交流。

论 文：AdaSparse: Learning Adaptively Sparse Structures for Multi-Domain Click-Through Rate Prediction

下 载：https://dl.acm.org/doi/abs/10.1145/3511808.3557541

1. 背景

传统的 CTR 预估模型往往是单场景的，即假设训练样本服从同一分布。但在工业应用中，这是个很强的假设。一方面，现实中往往面临多个业务需要同时预估的工作。例如在阿里妈妈外投平台，存在多个业务方如天猫 / 饿了么 / 蚂蚁等同时进行广告投放，各业务的投放数据分布往往不同。如图 1(a) 所示，不同的业务方（如 B5/B2）其用户和 CTR 效果差异较大，若合在一起训练模型可能被 B1 和 B5 两个业务主导，

从而导致其他业务效果较差。另一方面，除了不同的业务可以场景划分依据，相同业务下某些特征也可以作为场景划分依据，例如“不同的广告位，用户行为差异较大”、“相同业务不同用户画像，用户行为差异也可能较大”。如图 1(b) 所示，不同广告位下用户和 CTR 分布也有类似的差异现象。为此，我们这里定义一种“广义”的多场景：目标行为差异较大的特征，都可以作为一种多场景划分的依据。这也促使我们不断探索多场景预估技术，以通过捕捉用户在不同场景下的不同偏好来提升模型整体的效果。

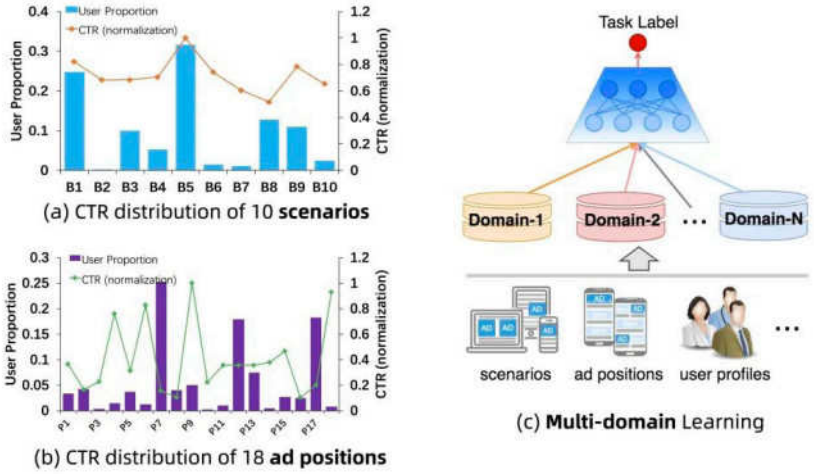


图 1　(a) 基于业务的场景划分；(b) 更广义的场景划分；(c) 多场景预估技术范式

图 1（c）显示了多场景预估建模的抽象范式，主要面临两个挑战：**1）跨场景泛化能力**，为了获得更好的效果，模型需要具备建模场景共性知识和捕捉场景个性知识的能力，同时还需要在稀疏场景有一定的泛化能力。**2）计算成本可控**，实现多场景预估技术往往需要更多参数，但在工业级应用中，计算资源有限，模型需要兼顾效率和效果。因此，我们需要寻找一种多场景预估技术：统一的模型，具备优异的场景迁移能力，且在线性能可控。

近年来涌现了不少多场景预估工作，通过对特征或网络做一些先验设计让场景相关特征在预估中发挥更大的作用。如图 2 所示，我们挑选了其中几个具代表性的工作，基于场景相关特征作用点的不同，归纳出了一条优化路线：场景相关特征作用从底层 embedding 到上层 bias，再到模型核心参数，再到网络结构。其复杂度越来越高，作用力也越来越强。具体地，如图 2 所示，SAML[12] 最开始对底层特征 embedding 做两套，一套用于场景共享，一套是场景特有；SAR-NET[13] 则对模型中间层的表征

做场景共性和特性的平衡操作。这类方法对场景信息的使用是偏底层的，但随着网络层数增加，加之与其他特征的相互作用，最终场景特征对预估产生多少影响是无法确定的。

因此，为了让场景相关特征发挥更大作用，出现了两个方向的思路：a) 信息离目标越近可能作用越强：将场景相关特征也作用在网络上层，诞生如 BiasNet，PPNET 的工作；b) 直接作用于网络中的核心参数上，如参数矩阵、Normlization 的参数等。在实际应用中，有时会将两个方向的方法做结合，比如 STAR[5]，既有上层 Bias 也有核心参数上做场景共性和特性的平衡。这些方法一定程度上解决了场景共性和特性的平衡，但针对稀疏场景的解决可能不足，尤其通过特征划分的广义场景很容易达到成千上万量级，稀疏场景会较多。于是后面出现了一些基于核心参数场景自适应生成的方法（如 M2M[11]、APG[8]），我们的方法主要聚焦这一类，并作用于网络结构。图 2（右）展示了多场景自适应参数生成方法的常见建模范式，这类方法通过设计辅助网络以场景相关特征作为输入，输出场景相关的参数作用于主模型。这类方法容易实现跨场景知识的迁移，同时因对主网络每一层都有作用，场景相关特征对预估的影响也较强，随之而来的挑战是参数爆炸问题，这给模型收敛和存储都带来了困难。因此，我们需要寻找一种方法既可以保持准确度，又能减少参数量。

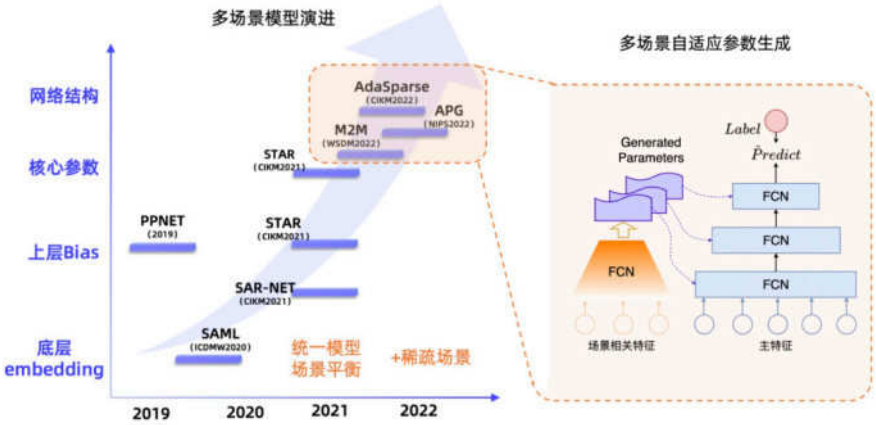


图 2　多场景预估模型技术演进及多场景自适应参数生成建模方法

我们借鉴了网络压缩的技术来实现上述目标。网络压缩技术包括网络裁剪、矩阵低秩分解、知识蒸馏等方法，我们主要采用网络裁剪技术（Network Pruning）[3,4,10]。网络裁剪技术认为神经网络中参数往往是冗余的，从原大模型中能够找到一个准确度不弱于原模型的子网络，通过合理的裁剪便可以得到这个子网络。AdaSparse 则是

将网络裁剪技术和多场景参数生成技术结合，为每个场景寻找一个子网络，各子网络重叠和不重叠部分可视为学习场景的共性和特性。

为兼顾参数量和效果，AdaSparse 将裁剪粒度定为神经元级，为此我们提供了一种场景相关的神经元级的权重因子方法：辅助网络以场景相关特征做输入，为主网络每一层神经元输出权重因子来衡量该神经元在该场景中的重要度。通过裁剪冗余 / 不重要的神经元，最终为每个场景学习到一个专属的子网络。由于神经元的数量远远小于模型参数（连接边），AdaSparse 确保了较低的计算复杂度。

2. AdaSparse

设训练数据集为 $\mathcal{D}=\{(x_j,y_j)\}_{j=1}^{|\mathcal{D}|}$，其中 x_j 和 y_j 分别代表第 j 个样本的特征和点击 label。我们基于场景相关特征将数据集 $\mathcal{D}$ 划分为多个场景，场景 d 相关的数据集可表示为 $\mathcal{D}^d=\left\{\left(x_i^d,x_i^a,y_i\right)\right\}_{i=1}^{|\mathcal{D}^d|}$，这里我们将特征拆解为场景相关特征集 x_i^d 和场景无关的特征集 x_i^a。以图 1（b）为例，x_i^d 如果仅用广告位划分，我们得到 18 个场景，如果加入更多场景划分的特征如业务、广告位、人群、广告样式等等，这将得到成千上万的场景。多场景 CTR 预估的目的便是通过学习一个模型 $\hat{p}CTR=f(x)$ 在所有场景上预估效果都较好。

2.1 模型概览

我们以层全连接 DNN[1] 作为主网络结构 $f(\cdot)$ 来详细介绍我们的方法。我们方法作用于神经元上，因此很容易扩展到其他网络结构例如 DCNv2[6]。所有特征 embedding 后，我们将场景相关 e_d 和无关 e_a 的特征拼接，记为 $[e_d,e_a]$，作为主网络的输入。记 $\mathbf{W}^l$ 为第 l 层网络要学习的参数矩阵，即有 $h^{l+1}=\tanh\left(\mathbf{W}^l h^l\right)$，这里 h^l 为第 l 层隐藏层（神经元）。模型 loss 用常见的交叉熵 loss。最终定义模型优化目标为：

$$\min_{\mathbf{W}}\frac{1}{|\mathcal{D}|}\sum_{d}\sum_{i=1}^{|\mathcal{D}^d|}\mathcal{L}_{CTR}\left(y_i,f\left(x_i^d,x_i^a\right);\{\mathbf{W}^l\}_{1\le l\le L}\right). \qquad (1)$$

AdaSparse 整体的模型流程如下图 3 所示，在主模型每相邻两层网络上做场景自适应。图 3 右侧以相邻两层网络为例详细展示了自适应裁剪的过程，核心在于中间这个场景自适应 Pruner 模块（domain-aware pruner），该模块是一个轻量级 MLP，以该层隐藏层拼接场景特征 embedding（即 $[h^l, e_a]$）作为输入，输出场景相关的神经元维度的权重因子向量（记为 $b\pi^l(d)$，代表场景 d 相关的权重因子向量），并作用在原

网络的这一隐藏层。基于权重因子的不同生成方式，我们进一步细分为三种方法，分别是二值法（Bianarization）、缩放法（Scaling）、二者混合（Fusion）的方法，后续我们将详细介绍这三种方法。此外，AdaSparse 对于每一层的常数项走单独的场景 BiasNet 来生成，因为不想限制常量 bias 的取值范围。同时，为了加速收敛，Pruner 模块梯度反向传播时停止对场景相关特征的更新，即场景相关特征的参数更新还是来源于主网络。

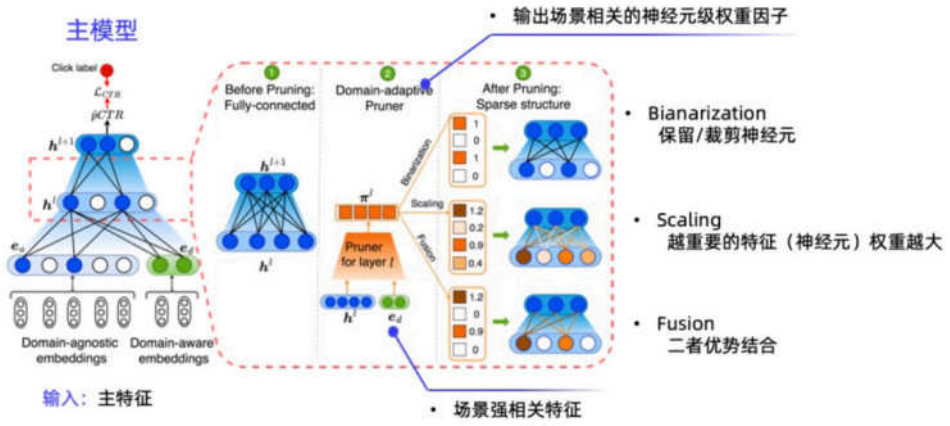


图 3　AdaSparse 建模流程图

2.2 场景自适应 Pruner

如图 4 所示，Pruner 是一个轻量级网络，以 $[h^l, e_d]$ 作为输入，通过 MLP 最上层连接 sigmoid 激活函数和一个软阈值操作函数 S_ϵ，最终得到场景 d 的权重因子 $\pi^l(d)$，然后和原始的隐藏向量做 element-wise 相乘：

$$\boldsymbol{\pi}^l(d) = S_\epsilon\left(\sigma\left(\mathbf{W}_p^l \cdot [\boldsymbol{e}_d; \boldsymbol{h}^l]\right)\right), \qquad (2)$$

$$\boldsymbol{h}^l(d) := \boldsymbol{h}^l \odot \boldsymbol{\pi}^l(d) \qquad (3)$$

这里软阈值操作函数 $S_\epsilon(\cdot)$ 对输出结果小于阈值 ϵ 的值强制变为 0，从而实现对神经元的裁剪。根据 $\sigma(\cdot)$ 和 $S_\epsilon(\cdot)$ 的不同，我们介绍三种权重因子生成的方法：

- **二值法（Binarization）**。我们希望权重因子为 0-1 值，动机是为场景直接剔除冗余 / 噪音信息，保留重要信息会带有效果有益。为避免模型欠拟合初期有裁剪错误的风险，我们希望在训练过程中逐步获得 0-1 值的向量，但激活函数 $y = sigmoid(x)$ 其值域是 (0,1)，要得到 0-1 值其实并不容易，需

要 $|x|$ 值较大，但模型的正则约束又会使 $|x|$ 不会太大。若我们直接按照阈值来获取 0-1，例如 $\epsilon = 0.5$，但这种方式过于直接，存在很大误裁剪的风险。为此，我们从两个方面优化：1）增加个自适应参数 α，激活函数设计为 $\sigma(\cdot) = sigmoid(\alpha x), \alpha > 0$，当 α 较大时，sigmoid 越“陡峭”，越容易得到 0-1 值。由于我们是 End2End 的训练方式，我们希望模型在训练的欠拟合初期不裁剪，α 初始化较小，随着训练增加而增大；2）Pruner 网络初始化设置大一些，这样容易获得输出绝对值较大。

- **缩放法（Scaling）**。该方法动机为相同的特征对不同的场景有不同的作用，我们希望作用越大的信息权重越大。Scaling 方法不做神经元裁剪，而是用一种“软裁剪”方式让冗余 / 噪音信息对预估结果影响较小。这里我们设置超参对权重幅度进行约束。
- **混合法（Fusion）**为上述二种方法的优势结合。

值得注意的是，这里用到的阈值函数其是一种符号函数，符号函数的特点是在断点处不可导，其他地方导数为 0，这个不适合模型反向传播做梯度更新。事实上，对于符号函数求导问题业界解决也比较成熟，这里推荐三种方法：a) stop gradient 方法（我们线上用的该方法）：前向传播正常计算，反向传播把这个函数屏蔽掉，保证权重正常求导；b) $Htanh(x)$ 或 $Hsigmoid(x)$：这是一种妥协的方法，反向传播时把符号函数替换为这个函数，该函数在 x 属于一定范围是能保证梯度；c) 借鉴 L0 正则的松弛技术：一些方法中通过 L0 正则产生稀疏网络，并提供了解决 L0 正则不可导的一种松弛技术，具体可参考文献 [14]。

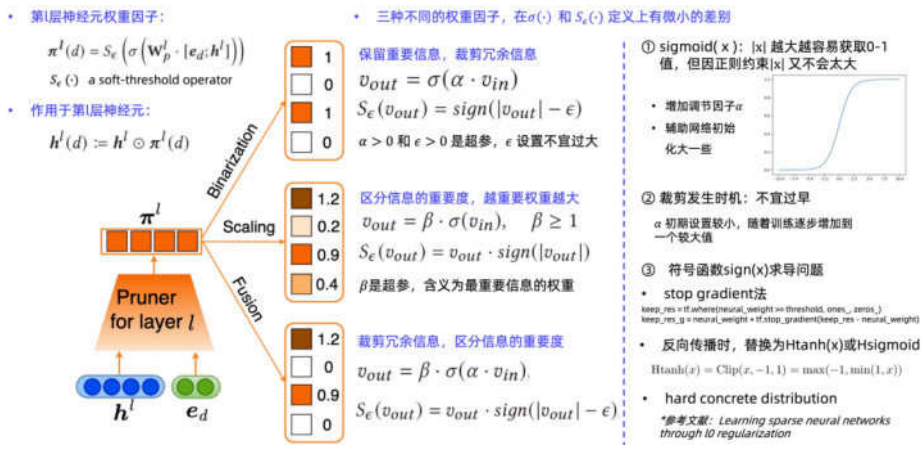


图 4　三种权重因子实现方法

2.3 稀疏度正则

稀疏网络产生过程中一个重要问题是：对于不同场景，稀疏度多少才合适？我们希望模型自动为每个场景学习到最优的稀疏度，但我们也担心自动学习的稀疏度过于异常。为此，我们提供一种灵活的稀疏正则方法，对稀疏度设置个范围，在该范围内各场景模型自动收敛到各自的最优稀疏度。设稀疏度表示为 $r^l = 1 - \frac{||\pi^l||_1}{N_l}$，这里 $||\cdot||_1$ 为 L1 正则，N_l 为 l 层隐藏向量长度。定义稀疏度控制范围为 $[r_{min}, r_{max}]$，稀疏正则公式设计如下：

$$R_s\left(\{\boldsymbol{\pi}^l\}_{1\le l\le L}\right) = \frac{1}{L}\sum_{l=1}^{L}\lambda^l\cdot|r^l - r|^2, \quad r = (r_{min} + r_{max})/2, \tag{4}$$

$$\lambda^l = \begin{cases} 0 & r^l \in [r_{min}, r_{max}] \\ \hat{\lambda}\cdot|r^l - r| & r^l \notin [r_{min}, r_{max}] \end{cases}, \tag{5}$$

这里参数有三个目的：a）如果稀疏度已经在预设范围，则稀疏正则不生效；2）自适应：在训练初期让模型专注拟合 label，稀疏正则作用小一些，后期再增强稀疏控制；3）合理的设置统一与主 loss 的量纲。

3. 实验及效果

我们分别在公开数据集 IAAC[8]（千万训练样本，300 个场景）和 产生数据集 Production（22 亿训练样本，5000+ 个场景）上分别进行实验。主网络结构分别为 DNN[1] 和 DCNv2[6]，AdaSparse 默认参数为 $\beta = 2, \epsilon = 0.25, [r_{min}, r_{max}] = [0.15, 0.25], \alpha$ 初始化为 0.1，上限为 5.0，其他参数可参考论文。Baselines 包括：MAML[9]，STAR[5]，APG[8]。

3.1 离线实验

从表 1（左）可知，“广义”多场景建模可以有效提升预估结果。由于公开数据集偏少，可以看到 DCNv2 效果整体弱于 DNN，并且实验中发现模型不易收敛，可以尝试减少参数量。得益于场景相关特征对预估的增强和场景的冗余 / 噪音信息裁剪，AdaSparse 取得了不错的结果。表 1（右）的消融实验也可以看到，将场景的冗余 / 噪音信息进行个性化裁剪或权重缩放，都能取正向效果，并且二者融合有一定的效果叠加。

多场景预估建模方法比较

Approach	Production Dataset			Public Dataset		
	LogLoss	AUC	GAUC	LogLoss	AUC	GAUC
DNN	0.063630	0.7266	0.6669	0.094886	0.6506	0.6442
+MAML	0.063474	0.7308	0.6715	0.093912	0.6531	0.6463
+STAR	0.063445	0.7313	0.6762	0.093598	0.6541	0.6505
+APG	0.063232	0.7332	0.6826	0.093264	0.6585	0.6553
+AdaSparse	**0.063215**	**0.7359***	**0.6848***	**0.093139**	**0.6607***	**0.6572***
DCNv2	0.063602	0.7282	0.6684	0.095097	0.6497	0.6443
+MAML	0.063619	0.7326	0.6730	0.094024	0.6536	0.6474
+STAR	0.063424	0.7325	0.6794	0.093754	0.6538	0.6501
+APG	0.063192	0.7356	0.6840	0.093375	0.6572	0.6536
+AdaSparse	**0.063187**	**0.7378***	**0.6871***	**0.093292**	**0.6594***	**0.6565***

AdaSparse 三种权重因子比较

Methods	AUC	GAUC
DNN	0.7266	0.6669
+ AdaSparse (Binarization)	0.7355	0.6843
+ AdaSparse (Scaling)	0.7334	0.6817
+ AdaSparse (Fusion)	**0.7359**	**0.6848**

表 1　离线实验

3.2 进一步分析

a) 跨场景泛化可视分析：我们通过 case study 来分析模型跨场景泛化能力。具体的，找一个相似 domain（Da 和 $\widehat{Da}$ 相似）和一个不相似 domain（Da 和 Db 不相似）（相似定义：domain 划分的特征值重合程度），取最后一层 layer 看 0-1 分布。结果如图 5（a）所示，可以看到相似 domain 其网络结构有更多的重叠，即共性学得更多，因此 AdaSparse 能够自动学习场景的共性和特性，实现跨场景迁移。

b) 对稀疏场景的泛化：我们对场景按照曝光数进行等频分桶（5 个），桶 BIN1 代表头部场景，其训练样本较丰富，BIN5 代表长尾场景，其样本较稀疏。图 5（b）显示所有的方法在头部 domain 上提效都更显著，但在长尾 domains 上 AdaSparse 有一定的优势，显示了 AdaSparse 对稀疏场景有不错的泛化能力。

c) 场景数与效果：我们通过逐步增加场景划分的特征，看场景数与模型效果的关系，如图 5（c）显示效果随着 domain 数量增加呈现先增加后降低的现象。该实验说明并非场景划分越多越好，需要一定经验来来设置合理的场景划分。

d) 稀疏度与效果：图 5（d）显示了不同稀疏度控制下效果的变化，可以看到效果随着稀疏度增加先提升后减少。注意到稀疏度为 [0.35,0.4] 时效果和 Base DNN 差不多，这也间接证明模型确实有冗余信息，裁剪冗余信息有潜在的提效空间。

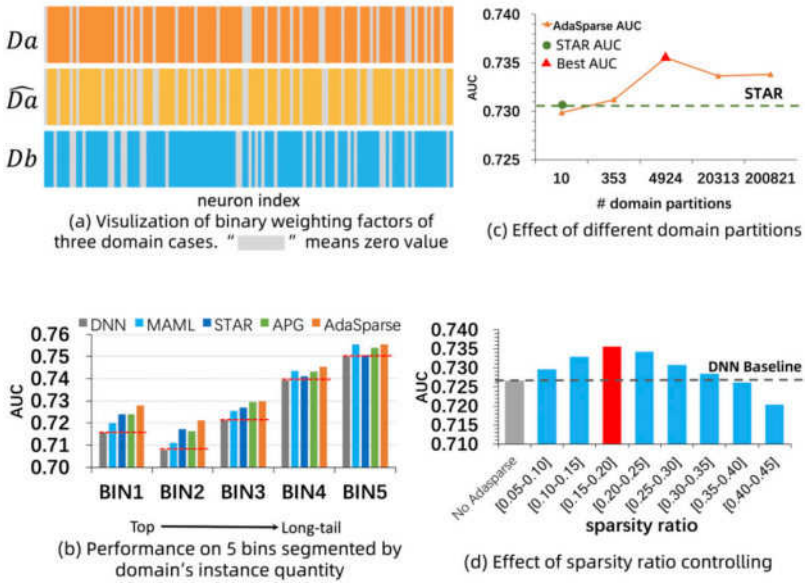


图 5：进一步分析

3.3 在线实验

我们最终采用 Fusion 版本在线实验，评价指标包括 CTR、CPC（点击成本，越低越好），同时我们想评估对各场景的提效情况，评估指标为 UPR：$UPR = \frac{\#\text{ Uplifted domains}}{\#\text{Total domains}}$，UPR 表示有效提效的场景占比多少。最终 AdaSparse 取得了 CTR +4.63%，CPC −3.82% 的正向提升，同时 UPR 取得超过 90% 的效果。

4. 总结与展望

本文分享的多场景 CTR 预估技术 AdaSparse，其结合网络裁剪技术为每个场景自适应学习一个子网络来进行 CTR 预估，方法简单轻量级，也取得了不错的线上效果。在实践中，我们进一步发现辅助网络倾向于被大场景主导，未来我们将进一步探索更有效的缓解场景稀疏的问题，同时小场景对大场景的辅助也是值得探索的方向。

5. Reference

[1] Paul Covington, Jay Adams, and Emre Sargin. 2016. Deep neural networks for youtube recommendations. In Proceedings of the 10th ACM conference on recommender systems. 191 – 198.

[2] Ruining He and Julian McAuley. 2016. Ups and downs: Modeling the visual evolution of fashion trends with one-class collaborative filtering. In proceedings of

the 25th international conference on world wide web. 507 – 517.

[3] Zhuang Liu, Jianguo Li, Zhiqiang Shen, Gao Huang, Shoumeng Yan, and Changshui Zhang. 2017. Learning efficient convolutional networks through network slimming. In Proceedings of the IEEE international conference on computer vision. 2736 – 2744.

[4] Jian-Hao Luo and Jianxin Wu. 2020. Autopruner: An end-to-end trainable filter pruning method for efficient deep model inference. Pattern Recognition 107 (2020), 107461.

[5] Xiang-Rong Sheng, Liqin Zhao, Guorui Zhou, Xinyao Ding, Binding Dai, Qiang Luo, Siran Yang, Jingshan Lv, Chi Zhang, Hongbo Deng, et al. 2021. One Model to Serve All: Star Topology Adaptive Recommender for Multi-Domain CTR Prediction. In Proceedings of the 30th ACM International Conference on Information & Knowledge Management. 4104 – 4113.

[6] RuoxiWang, Rakesh Shivanna, Derek Cheng, Sagar Jain, Dong Lin, Lichan Hong, and Ed Chi. 2021. DCN V2: Improved deep & cross network and practical lessons for web-scale learning to rank systems. In Proceedings of the Web Conference 2021. 1785 – 1797.

[7] Penghui Wei, Weimin Zhang, Zixuan Xu, Shaoguo Liu, Kuang-chih Lee, and Bo Zheng. 2021. AutoHERI: Automated Hierarchical Representation Integration for Post-Click Conversion Rate Estimation. In Proceedings of the 30th ACM International Conference on Information & Knowledge Management. 3528 – 3532.

[8] Bencheng Yan, Pengjie Wang, Kai Zhang, Feng Li, Jian Xu, and Bo Zheng. 2022. APG: Adaptive Parameter Generation Network for Click-Through Rate Prediction. arXiv preprint arXiv:2203.16218 (2022).

[9] Runsheng Yu, Yu Gong, Xu He, Bo An, Yu Zhu, Qingwen Liu, and Wenwu Ou. 2020. Personalized adaptive meta learning for cold-start user preference prediction. arXiv preprint arXiv:2012.11842 (2020). [10] Michael Zhu and Suyog Gupta. 2017. To prune, or not to prune: exploring the efficacy of pruning for model compression. arXiv preprint arXiv:1710.01878 (2017).

[11] Qianqian Zhang, Xinru Liao, Quan Liu, Jian Xu, and Bo Zheng. 2022. Leaving No One Behind: A Multi-Scenario Multi-Task Meta Learning Approach for Advertiser Modeling. arXiv preprint arXiv:2201.06814 (2022).

[12] Yuting Chen, Yanshi Wang, Yabo Ni, An-Xiang Zeng, and Lanfen Lin. 2020. Scenario-aware and Mutual-based approach for Multi-scenario Recommendation in E-Commerce. In 2020 International Conference on Data Mining Workshops (ICDMW). IEEE, 127 – 135.

[13] Qijie Shen, Wanjie Tao, Jing Zhang, Hong Wen, Zulong Chen, and Quan Lu. 2021. SAR-Net: A Scenario-Aware Ranking Network for Personalized Fair Recommendation in Hundreds of Travel Scenarios. In Proceedings of the 30th ACM International Conference on Information & Knowledge Management. 4094 – 4103.

[14] Christos Louizos, Max Welling, and Diederik P Kingma. 2017. Learning sparse neural networks through L_0 regularization. arXiv preprint arXiv:1712.01312 (2017)

贝叶斯分层模型应用之直播场景打分校准

作者：沧睿、北宸

1. 背景

精排预估打分是在线广告系统里非常重要的环节，其中打分准确性会极大地影响客户效果，也会影响用户体验和平台收益。精排预估打分为前链路打分和后链路打分，以 CPC（Cost Per Click）计费模式为例，前链路打分指点击率（Click-Through Rate，CTR），后链路打分指点击后客户获得的效果，这里的效果可以有很多种，常见的有成交量、收藏量等。

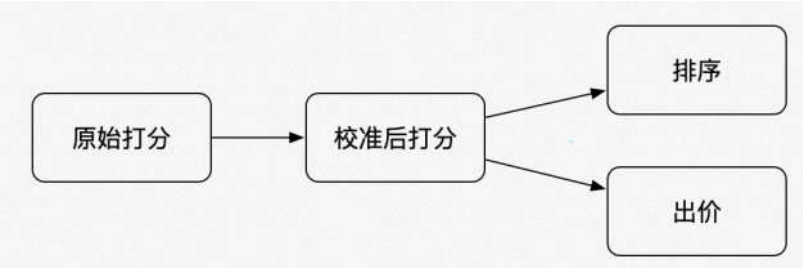
- **前链路打分主要用于广告排序**：特指预估点击率（pCTR，predict CTR），平台按照 pCTR * bid_price 进行排序，pCTR 准确性影响排序的准确性。
- **后链路打分主要用于出价决策**：比如预估成交率（pCVR，predict CVR），OCPX（Optimized Cost Per X）自动出价模式下，出价和后链路打分相关，因此打分的准确性直接影响客户最终投放效果。另外，BCB 以及 MCB 模式下也使用了后链路打分，打分准确是 BCB、MCB 最优性成立的基本条件。

精排预估打分模型主要采用深度模型，对特征、样本、模型结构的挖掘工作持续迭代，经过不断优化，打分精度也不断再创新高，尽管如此，精排预估打分仍然存在一些问题。

- **序准确 or 值准确**：一般而言，精排打分模型多数看的是 AUC 指标，AUC 指标对应序的准确性，对于值的准确性容易被忽视，比如 A 渠道统计 CTR 是 B 渠道的 2 倍，如果值是准确的，那么打分也应该为近似 2 倍的关系。
- **冷启动**：这里的冷启动包括新客户冷启动和新渠道冷启动，这部分流量缺少历史样本，精排预估打分准度受限。
- **维度效果差异巨大**：某些维度下的效果存在很大差异，精排模型精准地学到这些倍数关系非常困难。尤其是流量渠道维度，好渠道的 CVR 可能是差渠道 CTR 的几十倍。
- **样本倾斜**：大多数场景存在长尾效应效应，例如大占比数量的流量渠道的样本量占比较低，由于精排模型统一建模，或多或少存在少样本“迁就”多样本现象。

校准是解决以上问题最直观、最直接的方法，也是业界常用的方法。它的思路很简

单，就是在精排模型打分阶段后对打分进行修改，后续机制策略使用校准后的分，流程图如下：

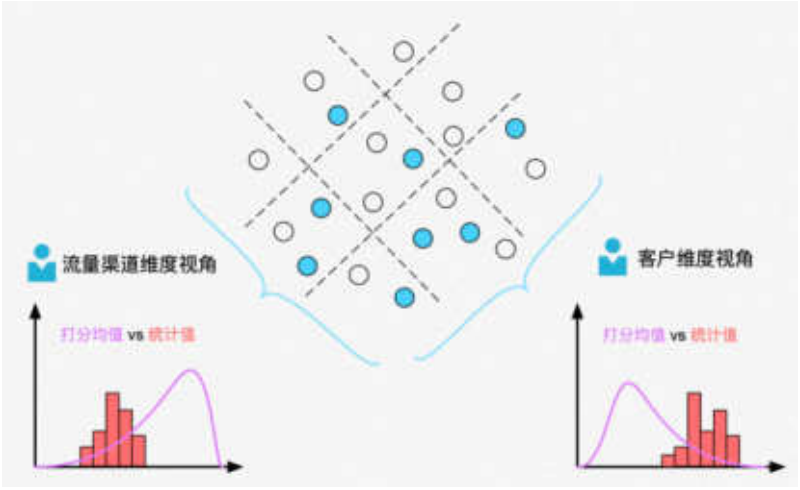


一般而言，校准模块需要要具备以下特点：

- **简单轻量**：根据定位一可知，校准模块不做重度优化，设计理应简单轻量。
- **可解释**：一方面，可解释意味着合情合理，badcase 可能会少，或者出现 badcase 更容易排查，另一方面，校准结果的解释性可为精排模型提供优化思路。
- **鲁棒 & 风险可控**：校准模块承担着为精排模型保驾护航的作用，需要降低校准后效果更差的风险。

2. 问题分析

我们之所以认为精排模型存在值准度问题，大多数情况是因为在做一些数据统计分析的时候，发现某些维度的打分不满足直观的统计特性，如下图所示：



实际角度来看，不管是流量渠道维度和客户维度，统计值和预估值存在较大差异。我们可以通过统计学假设检验来解释这个问题，以 CTR 打分为例，我们挑选了一些打分值都为 9.07% 的实际样本，总样本量为 444587，正样本量为 36267。原假设为「所有样本真实 CTR 都为 9.07%」，这里我们采用正样本量作为统计量，此时正样本量服从 n = 444587，p = 0.0907 的二项分布，我们很容易计算。$\text{p-value} = 4.02 \times 10^{-103} << 0.01$，显而易见我们要拒绝原假设，通俗的解释就是如果原假设成立出现这种样本的几率几乎为 0。原假设被拒绝意味着至少一部分样本 CTR 不是 9.07%。

由于无法确切知道每个样本的真实 CTR，统计推断结论基本只能到这一步，即「我们知道打分有问题，但不知道哪些样本有问题」，此时校准就显得有必要了。

业界常用的校准算法中，非参数方法能做到自适应，但方法复杂一点则缺乏一定的可解释性；参数方法可解释，但目前存在的建模不够复杂，表达能力不足；混合方法看起来能取长补短，但如何结合存在一定的 trick，策略联动没有理论的必然性。因此，这里我们使用了贝叶斯分层模型，相对于参数方法，模型更复杂更具有表达性，相对于混合方法，又能将贝叶斯统计理论贯彻到底，达成一体化建模的目的。

贝叶斯分层模型具有以下特点：

- 纯粹的贝叶斯概率统计模型，显式建模，最终产出是参数后验分布；
- 使用层次参数结构，增加了共性和个性的表达能力，具备一定泛化性，也允许字段缺失；
- 贝叶斯模型产出的参数后验分布，使得具有风险决策的能力。

3. 建模方案

为了让校准工作能够顺利开展，需要做 **bias 一致性假设**，即「在确定的维度值条件下，所有样本打分具有固定的偏差」。概率模型校准的思路比较直接，即通过概率建模，利用有限的样本推断每个维度下的 bias，形式化如下：

$$\ln \text{ctr} = \ln \text{pctr} + \text{bias}$$

这里使用 ln 的原因是使得 bias 具备相对偏差的概念，举个例子，bias = −0.05，意味着 pctr 大概高估了 5%。

这里的维度是指某种样本切分的规则，维度值是指按照这个切分规则下的一个聚簇。例如，「客户 * 渠道 * 打分区间」表达了一种维度，「客户 A、渠道 B、打分

0.01~0.02 之间 | 表达了该维度下的一个值。对于维度的选择需要一定的经验，一般和具体业务相关，如果维度数量过少，基本假设成立的可能性越低，校准效果有限；如果维度数量越多，bias 参数越多，模型越复杂，而且平均每个维度的样本就稀疏，最终也会影响校准精度。实际情况下，一般需要进行折中，即只选取少量重要的维度。

为了方便理解，这里我们由简到繁、循序渐进介绍各种建模方法，如下：

- **全局维度**：所有样本归为一个维度，最简单，效果也有限。
- **独立维度建模**：在全局维度基础上增加分层概念，不同客户的样本 bias 不同，最终模型具有一定的泛化性和解释性，解决了传统机器学习面临的样本倾斜问题；另外，在客户分层维度的基础上，增加独立的流量渠道维度，模型表达能力更高。
- **交叉维度建模**：增加客户、流量渠道交叉维度，解决客户、流量渠道对 bias 的作用不独立的情况。
- **直播广告场景建模实践**：这一小节以直播广告场景为例介绍建模方法。

3.1 全局维度建模

全局维度是最简单的建模方式，即所有样本都归为一个维度，bias 全局唯一，建模如下：

$$\ln y = \ln r + \theta + \varepsilon$$

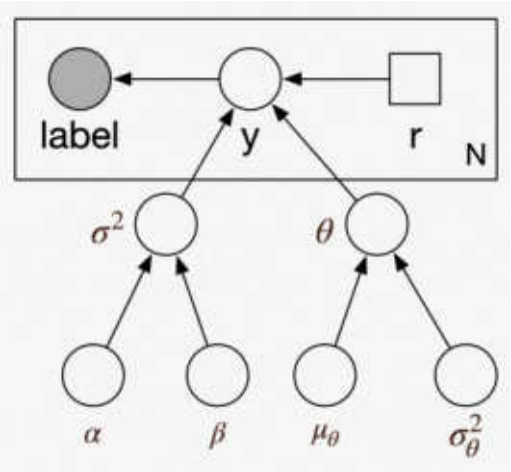
其中 $\varepsilon \sim \text{Normal}(0,\ \sigma^2)$

y 为真实 ctr，r 为 pctr，为 bias

根据贝叶斯理论，我们可以引入先验，即：

$$\theta \sim \text{Normal}(\mu, \sigma_\theta^2)$$
$$\sigma^2 \sim \text{Inv-Gamma}(\alpha,\ \beta)$$

最终概率盘式图（Plate Notation）如下：



其中 N 为样本量，label 为观测到的样本点击与否

由于这是一个线性模型的特例，为了方便表达，这里我们使用线性模型独有的表达方式，如下所示：

$$(\ln y - \ln r) \sim 1$$

解释如下：

- 波浪线左侧为目标部分，右侧为效应部分；
- 效应以符号 + 分隔，目标等于所有效应项之和加上残差项，残差项默认自带，因此可省略不写；
- 效应 = factor * parameter，由于 parameter 默认自带，我们只用写 factor，factor = 1 表示该项为截距项；
- 例如，$y \sim 1 + x_1 + x_2$ 表达的含义为 $y = 1 * \theta_0 + x_1 * \theta_1 + x_2 * \theta_2 + \varepsilon$。

上面表达式表达的含义为 $\ln y - \ln r = 1 * \theta + \varepsilon$，$\theta$ 和 ε 可省略，因此右侧写成 1。

实际我们使用全局维度的意义不大，一方面，维度太少 bias 一致性假设成立可能性低，另一方面，实际总样本量一般非常充足，使用纯规则统计的方式就能得到非常置信的结果，没有必要通过模型来推断。

3.2 独立维度建模

为了增加模型的表达能力，我们需要增加一些维度。以客户维度为例，每个客户应具有不同的 bias，最简单的方法是按照 3.1 小节的方法建 M 个独立模型（M 为客户数）

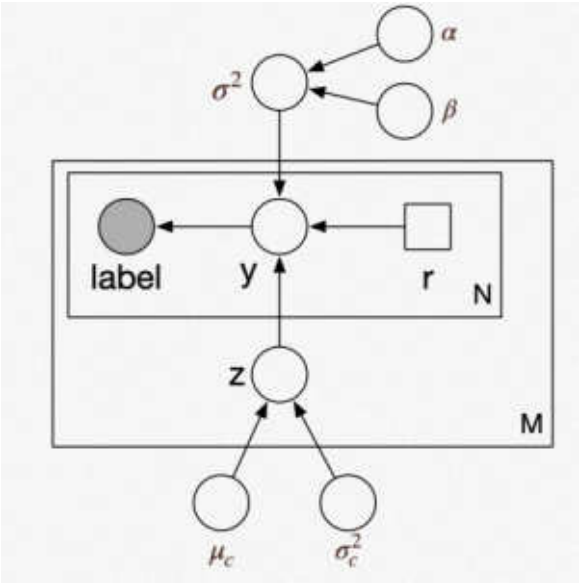
但这种方法存在问题，不同客户样本不一样，有些客户样本比较稀疏，此时模型就会极度依赖先验，那么问题来了，不同客户的先验参数怎么确定。

目前最直接可行的方法就是假设所有的客户先验参数一样，即共用先验参数，这么做也具备很好的泛化性和解释性。例如，某个客户没有样本或者样本很少，此时我们需要大部分参照先验，然而先验是综合所有客户计算出来的，已经能够充分表达所有客户“平均”的情况，合情合理。

既然我们把所有客户的先验参数统一起来，那么最初我们讨论的「M 个独立模型」这个概念就不复存在了，因为它们都共享了先验参数，此时使用一个模型就能表达，即分层模型，具体建模如下：

$$\ln y = \ln r + z_i + \varepsilon$$
$$\varepsilon \sim \text{Normal}(0,\ \sigma^2)$$
$$z_i = \text{Normal}(\mu_c,\ \sigma_c^2)$$
$$\sigma^2 \sim \text{Inv-Gamma}(\alpha,\ \beta)$$

z_i 表示客户 i 的 bias，且这个样本属于客户 i，μ_c 和 σ_c^2 为共享的参数。概率盘式图表达如下：



我们可以把 z_i 代入到目标式子里面，得到另外一种等价的建模，即混合效应模型（**Mixed-Effects Model**，https://en.wikipedia.org/wiki/Mixed_model），如下：

$$\ln y = \ln r + \mu_c + s_i + \varepsilon$$
$$\varepsilon \sim \text{Normal}(0,\ \sigma^2)$$
$$s_i = \text{Normal}(0,\ \sigma_c^2)$$
$$\sigma^2 \sim \text{Inv-Gamma}(\alpha,\ \beta)$$

不同领域对层次模型的称呼习惯不同，比如社会学通常叫分层线性模型（multilevel linear model），生物统计研究经常使用混合效应模型（mixed-effects model）和随机效应模型（rankdom-effects model），计量经济使用随机系数回归模型（random-coefficient regression model），统计学使用协方差成分模型（covariance components model）。

混合效应分为固定效应和随机效应，以上面的表达为例，μ_c 截距项为固定效应，全局唯一，s_i 为随机效应项，和客户有关且满足总体均值为 0 的约束。通过转化为混合效应模型，我们能得到更为简洁的线性模型表达，如下：

$$(\ln y - \ln r) \sim 1 + 1|\text{customer}$$

和 3.1 小节相比，这里多了一个随机效应 ，中间竖线左侧仍然为 factor，竖线右侧表达了分组的概念，即不同的 customer 具有不同的 parameter，且服从总体均值为 0 的约束。

这个模型的统计解释性如下：

- 对于没有样本的新客，随机效应为 0，此时这个客户的打分 bias 均值等于固定效应；
- 对于样本较少的客户，由于随机效应参数有 0 均值约束，结合很少的样本，模型推断的随机效应应该为接近 0 的值，此时这个客户的打分 bias 均值基本接近固定效应；
- 对于样本较多的客户，模型推断的随机效应参数极大参照样本，此时这个客户的打分 bias 基本等于统计 bias（统计 ctr 除以打分均值）。

层次模型可以解决**统计陷阱**和**样本倾斜**的问题。举个例子，某个场景客户分为 3 类，头部客户、腰部客户和底层客户，他们的点击样本信息（构造数据）如下：

客户类型	客户数量	样本数量	统计 ctr
头部客户	100	1000w	0.1
腰部客户	1000	100w	0.03
尾部客户	5000	10w	0.01

那么，问题来了，假设来了一个新客户（无额外信息），他的 ctr 应该是多少？由于我们对这个客户一无所知，通常来看，我们使用大盘平均 ctr 作为新客户的 ctr 合情合理。何为大盘平均 ctr？按照日常统计数据的惯性思维，很容易就想到了**样本统计微平均**，即总样本点击量除以总样本量。以上表为例，大盘 ctr 平均值 = (1000w * 0.1 + 100w * 0.03 + 10w * 0.01) / (1000w + 100w + 10w) = 0.0928。但是，另外存在一种统计方式，即**样本统计宏平均**，这种方法以客户作为统计粒度，即大盘 ctr 平均值 = (100 * 0.1 + 1000 * 0.03 + 5000 * 0.01) / (100 + 1000 + 5000) = 0.0133。这两种统计方式得到的平均值截然不同，哪种统计更合理取决于问题的问法，如果问题是「假如来了一个曝光，它的 ctr 是多少？」，那么微平均合理，如果问题是「假如来了一个客户，它的平均 ctr 是多少？」，那么宏平均合理，很显然原始问题是属于后者。

同样精排模型训练数据使用的是最细粒度的样本，如果处理的不好很容易掉入上面描述的统计陷阱。校准可以针对这个问题设计分层模型结构，不同样本量的客户最终会在顶层参数拉齐（类似上面例子的宏平均统计方法），最终减少样本倾斜带来的影响。

同样，在客户基础上，我们可以增加更多的随机效应，比如流量渠道维度，具体建模如下：

$$(\ln y - \ln r) \sim 1 + 1 \mid \text{customer} + 1|\text{pid}$$

其中 pid 为流量渠道的区分信息。

新建模使得不同 pid 具备不同变化的能力，如果不同的 pid 确实存在较大的 bias 差异，这种建模方式能提升校准效果。当然，这样建模存在假设，即 customer 和 pid 对应的随机效应参数是正交的，即同一个客户不同的 pid bias 分布都是一样的，同理，同一个 pid 不同客户的 bias 分布都是一样的。

另外，从模型结构上来看，增加了 pid 以后，这个模型已经不是严格意义上的分层结构（树状分层）了，但出于习惯问题，我们仍然称呼它为分层模型，这并不影响工作的开展。

3.3 交叉维度建模

有时候 3.2 小节的多维度参数正交的假设并不成立，比如存在某个客户某个 pid 的打分 bias 和其他的明显不同，此时使用独立维度建模就不能表达这种特例了。这里我们可以增加 customer 和 pid 的交叉维度，建模如下：

$$(\ln y - \ln r) \sim 1 + 1 \mid \text{customer} + 1 \mid \text{pid} + 1 | \text{customer,pid}$$

新随机效应项表达的含义为不同的 customer、pid 组合具有不同的参数，这些参数仍然满足 0 均值约束。这么做增加了 customer、pid 组合独立变化的能力，解决了 3.2 小节正交假设不成立问题。

到底假设正交还是假设不正交呢？如果真实情况是正交，实际却增加了交叉维度，这样会导致模型复杂度增加，不会产生更好的效果，但是从直觉上来看，完全正交的假设又不太可能。所以这里需要根据实际情况进行折中，一般而言如果样本足够多，现有模型使用的维度少，那么可以考虑增加交叉维度。

3.4 直播广告场景建模实践

直播广告场景下，采用以下建模方式：

$$(\ln y - \ln r) \sim 1 \mid \{\text{customer, pid, r_level}\}^*$$

$$\text{其中 r_level} = \lfloor \text{scale} * \ln r \rfloor$$

说明：

- 这里 $1 \mid \{\text{customer, pid, r_level}\}^*$ 表达的含义为所有的 customer、pid、r_level 组合组成的效应（包含固定效应），一共 $2^3 = 8$ 个效应项，即：

$$1 + 1 \mid \text{customer} + 1 \mid \text{pid} + 1 \mid \text{r_level} + 1 \mid \textit{customer} + 1 \mid \text{customer,pid} + \ldots$$

- r_level 是 r 的离散化区间，使用 r_level 的原因是根据实际经验，不同打分区间的 bias 存在较大差异，某种意义上来讲，r_level 提供了一种非线性化校准能力。

这样建模其实是综合直播业务特点以及校准定位给出的折中方案，主要体现如下：

- **简单轻量**：仅使用 customer、pid、r_level 三个维度，最终模型仅具有 8 个效应项，满足校准的简单轻量定位，定位问题简单。如果增加一个维度，效应项会翻倍（8 个变成 16 个），训练预测开销翻倍。当然，我们也可以不用维度全组合作为效应项，即挑选部分最有用的组合作为效应项，但这部分仍然在研究探索之中。
- **增加维度边际收益低**：customer、pid、r_level 是肉眼可见 bias 存在较大差异的维度。实际角度而言，增加维度带来的效果并不明显。

4. 模型推断 & 校准流程

众所周知，复杂的贝叶斯模型最大的难点在于后验参数推断，这里我们使用了阿里妈妈开发的 BGLM（Bayes Generalized linear model）框架，使得贝叶斯推断变得简单。

BGLM 采用了【Mean-Field】https://en.wikipedia.org/wiki/Mean-field_theory 变分推断，简单解释为，使用各个参数独立的分布代替复杂的联合后验分布 $p(\mathbf{Z})$，这个分布寻找的准则为新老分布的 KL 散度最小化，公式表达如下：

$$q^* = arg_q min \mathrm{KL}(q \parallel p)$$

q 为近似分布，表达为 $q(\mathbf{Z}) = \prod_i^M q(Z_i)$

p 为真实贝叶斯后验分布

贝叶斯推断存在另外一大类方法，即 MCMC（Markov Chain Monte Carlo），https://en.wikipedia.org/wiki/Markov_chain_Monte_Carlo。变分推断相比 MCMC 最大的优势是收敛迅速（类似 EM 算法），20~30 轮就可收敛，资源占用较少，另外近似分布一般有具体的解析表达，利于进行后续决策推导，当然变分推断也存在一定缺陷，即分布是近似的，收敛后近似后验分布和真实后验分布存在固有 gap，但 MCMC 方法不存在这个问题，只要样本足够多就能无限概率逼近真实后验分布。

经过 BGLM 的变分推断，我们可以得到参数的后验近似分布。有了参数的后验近似分布，我们可以针对预测样本进行后验预测推断。得益于参数后验近似分布有具体形式化表达，预测样本的线性预测子（线性预测子为所有效应项之和）可以很容易推导，最终表达为高斯近似分布如下：

$$\ln y - \ln r \sim \mathrm{Normal}(\mu_{\mathrm{approx}}, \sigma^2_{\mathrm{approx}})$$

有了线性预测子的分布，我们可以很容易得到目标值分布，即：

$$\ln y \sim \mathrm{Normal}(\ln r + \mu_{\mathrm{approx}}, \sigma^2_{\mathrm{approx}})$$

引擎一般只接受一个值，而不是一个分布，因此我们需要额外进行打分决策，根据上面的结论，我们可以简单使用线性预测子的均值作为决策值（风险中性），即：

$$r_d = \exp(\ln r + \mu_{\mathrm{approx}})$$

其中 r_d 为最终校准打分决策

到这里，我们就有了简单的校准流程：

1）构造训练样本，包含 label、打分以及相应特征；

2）使用 BGLM 对模型的参数进行贝叶斯后验推断，参数的变分近似后验分布即为“模型”；

3）在线来了一个预测样本，根据打分、特征调用 BGLM 预测模块，得到真实目标值的后验预测分布；

4）根据打分的后验预测分布进行打分决策。

5. 校准风险决策

根据 4 小节，我们给出了最简单的风险中性校准决策，即使用线性预测子的均值，但是这种方式抛弃了预测方差，对于方差较大的样本，存在校准结果变差的可能，因此这里提出了风险决策。

设观察数据为 D，预测样本为 S，校准打分 $\ln r_d$ 的后验预测概率密度为 $P(\ln r_d \mid D, S)$。

一方面，我们希望决策的校准分尽量后验预测概率大（经验风险），可以表达为概率密度的熵，即：

$$L_e(r_d) = -\ln(P(\ln r_d \mid D))$$

另一方面，我们又希望最终校准后的分离原始分差异不要太大（校准幅度风险），校准幅度风险可表达为平方损失，如下：

$$L_a(r_d) = \frac{1}{2}(\ln r_d - \ln r)^2$$

两个风险函数结合起来组成我们最终的风险函数（λ 为权重因子，$\lambda \geq 0$）。

$$L(r_d) = L_e(r_d) + L_a(r_d)$$

我们的目标为寻找风险值最小的 r，即：

$$r_d^* = \underset{r_d}{argmin}\, L(r_d)$$

根据 6 小节可知，$P(\ln r_d \mid D, S)$ 的形式为高斯分布，我们可以直接求出最优的 r_d^*，即：

$$r_d^* = \exp\left(\frac{(\lambda\sigma_{\text{approx}}^2 + 1)\ln r + \mu_{\text{approx}}}{\lambda\sigma_{\text{approx}}^2 + 1}\right)$$

可以看到：

- 当 $\lambda = 0$ 时，$r_d^* = \exp(\ln r + \mu_{\text{approx}})$，此时等价于风险中性校准，即使用均值作为决策值；
- 当 $\lambda = +\infty$ 时，$r_d^* = r$，等价于不校准；
- 当 σ_{approx}^2 很小时，说明数据充分，$r_d^* \approx \exp(\ln r + \mu_{\text{approx}})$，此时很大程度参照模型推断结果；
- 当 σ_{approx}^2 很大时，说明数据不充分，$r_d^* \approx r$，此时应该选择不校准。

6. 校准效果

这里介绍一下校准的评估指标：

- **AUC**：衡量序的准确性，精排模型关注的重点指标。
- **GAUC（Grouped AUC）**：由于 AUC 评估的是大盘，直播场景不同流量渠道正样本率差异非常大，AUC 经常能达到 0.9 以上，此时 AUC 很难反映流量渠道内部打分的准确性，GAUC 期望解决这类问题。它的计算方式为先按渠道计算 AUC，然后各个渠道的 AUC 再求平均。
- **PCOC**：即统计正样本率除以预估打分均值，PCOC = 1 最好，PCOC 大于 1 或者小于 1 表示打分存在问题。PCOC 指标粒度太粗，不能反应细节，因此不能过低依赖这个指标。
- **ERROR_ 维度值**：首先统计某个维度值下的样本的 PCOC，维度值示例：「流量渠道直播广场」，计算方式：

  $$\text{ERROR}_{\text{维度值}} = \begin{cases} \text{PCOC}_{\text{维度值}} - 1, & \text{if PCOC}_{\text{维度值}} \geq 1 \\ 1/\text{PCOC}_{\text{维度值}} - 1, & \text{if PCOC}_{\text{维度值}} < 1 \end{cases}$$

  可以看到这个值越小越好，最好情况等于 0。

- **ERROR_ 维度**：维度示例「流量渠道」，它包含多个维度值。

  $\text{ERROR}_{\text{维度}} = \frac{1}{n}\sum_{\text{维度值}\in\text{维度}} \text{ERROR}_{\text{维度值}}$。当然某些维度值正样本不足，PCOC 波动较大，这里可以选择不参与计算。

目前来看，直播广告为校准的主要应用场景，前链路打分校准 ERROR 有 66% 的

降幅，流量实验也能带来 5% 的收入提升；后链路打分（成交率、吸粉率等等）校准 ERROR 有 5~25% 不同程度的校准，转化成本有 2-19% 不同程度的降低，显著提升了客户体验。另外，经过调优后的校准风险决策大概能带来 ERROR 1-5% 的降幅。

7. 总结

针对直播场景精排打分值准确性、冷启动等问题，业界常用解决方案是打分校准。为了达成校准的目的，我们使用了基于纯贝叶斯理论的分层模型。一方面，贝叶斯分层模型简单轻量、可解释、可泛化；另一方面，贝叶斯后验分布天然提供了风险决策能力，降低了校准精度风险。实际角度来看，在直播广告场景下，贝叶斯分层模型带来了不错的效果。

召回匹配

代码开源！阿里妈妈展示广告 Match 底层技术架构最新进展

作者：卓立、日涉、谨持

一、背景

大规模信息检索一直是搜推广领域的核心问题之一，而基于任意复杂模型的检索方案无疑是业界重要的迭代方向之一。近年来，阿里妈妈展示广告 Match 团队与预测引擎团队专注于从算法与工程角度推动工业级大规模检索技术的研发，我们在基于任意复杂模型的检索方向上积累了一定经验并取得了不错的业务效果，现整理发布 NANN（Neural Approximate Nearest Neighbor，以下简称 **NANN**）并对外开源，希望通过社区的协同创造力，共同推进该领域的发展。

本文介绍的 **NANN** 源自阿里妈妈展示广告 Match 团队研发的二向箔算法体系，该方案在保留复杂模型召回能力的同时，将索引学习和模型训练解耦，提供了轻量化的任意复杂模型召回解决方案。NANN 基于 Tensorflow，提供了性能 benchmarking 工具以及完整的由模型训练至在线 deployment 的 demo。

该方案由阿里妈妈技术团队自研，已在阿里巴巴集团内部其他业务进行推广上线，在典型的搜索、推荐、广告场景均取得了显著的业务收益。

本文将围绕 NANN 核心功能及算法体系更新进行简要介绍，欢迎大家试用和交流讨论。

相关链接：

开源地址（点击↓阅读原文）： https://github.com/alibaba/nann

相关阅读： TDM 到二向箔：阿里妈妈展示广告 Match 底层技术架构演进

论文下载： CIKM 2022 | Approximate Nearest Neighbor Search under Neural Similarity Metric for Large-Scale Recommendation，https://arxiv.org/pdf/2202.10226.pdf

二、核心功能介绍

NANN 代码库基于原生 TensorFlow 框架实现了高性能的二向箔算法在线检索服务。同时对于 NANN 算法的训练、索引结构构建、离线效果及在线性能评测、部署等流程提供了详细的示范。

该开源代码库的主要优点可以分为模型训练、性能优化和用户友好性三方面：

2.1 模型训练

- **任意复杂模型**：模型训练与索引构建解耦，因此对模型结构几乎没有任何限制，也避免了索引和模型训练绑定带来的高额训练负担。
- **对抗训练**：我们采用对抗训练来保证复杂模型下的优越检索性能。

此外，我们还提供了预训练模型和格式转化后的测试数据集，以方便大家试用及验证。

2.2 性能优化

- **高效检索**：我们使用 TensorFlow Custom Ops 实现了 HNSW 检索过程。就在线检索而言，重写后的 HNSW 检索比 Faiss 原生版本更加高效。
- **运行时优化**：我们支持 GPU Multi-Streaming with Multi-Contexts，这极大地增强了并行性。
- **编译优化**：我们支持 Dynamic Shape 下的 XLA 加速能力，并利用缓存等手段提升了 just-in-time（JIT ）编译效率，最终使得 NANN 整个检索过程可以充分利用 XLA 加速。
- **图级优化**：我们针对推荐、搜索和广告领域中常用的一些常见的模型结构，基于 TensorFlow Grappler 提供了一些图级优化。

2.3 用户友好性

- **原生 TensorFlow**：NANN 的后端服务和前端实现完全基于 TensorFlow 生态系统。
- **模型推理和检索解耦**：训练 – 检索解耦使得用户能专注优化深度模型，无需额外考虑检索流程。
- **性能测试**：我们提供了一个简单的基准测试工具，可用于分析延迟、吞吐量等推理性能。

环境搭建、模型训练、部署等更多细节详见开源库 https://github.com/alibaba/nann。

三、算法体系更新

我们去年在《TDM 到二向箔：阿里妈妈展示广告 Match 底层技术架构演进》这篇文章里介绍了初版的二向箔算法体系。在文章末尾的展望部分，我们也提到图索引构建中采用的 L2 距离度量与在线检索采用的模型打分之间存在不一致，这会导致检索精度会随着模型结构变复杂而大幅降低。为此，我们在训练过程中加入了一个对抗训练辅助任务，大幅减轻了相似性度量之间异质性的负面影响，有效提高了检索精度。

具体来说，如表 1 所示，在内部数据集（Industry）和开源数据集（User Behavior）两个数据集上，随着 User 侧和 Target 侧的交互越来越多，虽然检索 recall（图中 recall-retrieval）提升明显，但同时与全量打分 recall（图中 recall-all）之间的差距（图中）也越来越大。

Dataset	Aux loss	model	recall-retrieval	recall-all	recall-Δ
Industry	w/o	two-sided	28.8%	28.9%	0.35%
		DNN w/o attention	33.9%	34.1%	0.59%
		DNN w/ attention	39.2%	42.8%	8.55%
	w/	two-sided	30.1%	30.2%	**0.33%**
		DNN w/o attention	34.1%	34.2%	**0.29%**
		DNN w/ attention	**42.9%**	**43.2%**	**0.60%**
User Behavior	w/o	two-sided	11.3%	11.4 %	0.74%
		DNN w/o attention.	12.8%	13.1%	2.30%
		DNN w/ attention	23.9%	24.7%	3.48%
	w/	two-sided	12.4%	12.5%	**0.40%**
		DNN w/o attention	13.1%	13.3%	1.20%
		DNN w/attention	**24.9%**	**25.6%**	3.00%

表 1　不同模型结构下的检索 recall、全量打分 recall、检索精度损失结果

我们认为该现象主要是因为模型打分的非线性随模型复杂度的增加而大幅提升。在双塔模型（two-sided）下，模型打分函数与 target embedding $\mathbf{e}_v$ 为线性关系，因此 L2 距离近的 target embedding 对应的模型打分也相近，最终表现为检索损失很低，在 Industry 和 User Behavior 两个数据集上分别为 0.35% 和 0.74%；而在结构最复杂的 attention 模型（DNN w/attention）下，一旦给 target embedding $\mathbf{e}_v$ 加上微小的扰动，模型打分可能会发生剧烈的变化，因此检索精度损失大幅升高。为此，

我们在模型训练中加入了如下辅助函数来控制模型打分的平滑性：

$$\mathcal{L}_{AUX} = \sum_{u} \sum_{v \in \mathcal{Y}_u} s_u(\mathbf{e}_v) \log \frac{s_u(\mathbf{e}_v)}{s_u(\mathbf{e}_v + \Delta)}$$
$$\Delta = \epsilon \, \text{sign}(\nabla_{\mathbf{e}_v} s_u(\mathbf{e}_v))$$

其中 u 为用户，$\mathcal{Y}_u$ 为该用户的正、负样本集，$s_u(\mathbf{e}_v)$ 表示用户 u 和 target v 的模型打分，Δ 为模型打分 $s_u(\mathbf{e}_v)$ 在 target embedding $\mathbf{e}_v$ 处对抗梯度方向的微小扰动。

从表 1 可以看出，加入对抗训练可以在不损失全量打分 recall 的情况下大幅降低复杂模型的检索损失，从而大幅提升检索性能。

四、总结展望

从模型角度看，加入对抗训练的二向箔算法体系真正做到了任意复杂模型下的高精度、高性能大规模 KNN 检索。我们将该方案整理开源，为社区提供了一套轻量级的支持任意复杂模型的大规模召回解决方案，该项工作相关工作论文已发表在 CIKM2022。**期待大家试用和体验，如遇问题，也欢迎与我们讨论。**

五、关于我们

阿里妈妈展示广告 Match 团队多年来致力于大规模检索技术。经过团队同学多年的努力与沉淀，发展并迭代了如 TDM、JTM、BSAT 以及 NANN 等多项核心检索技术。近年来，阿里妈妈展示广告 Match 与阿里妈妈预测引擎团队深度合作，逐步从算法单一视角迭代演变为算法与工程双视角迭代。在**阿里妈妈预测引擎团队**异构计算技术能力的加持下，进一步释放模型迭代的天花板。同时，从业务及算法视角反哺阿里妈妈预测引擎团队，进一步扩展了妈妈预测引擎团队技术能力的通用性。

相关链接：

开源地址（点击↓阅读原文）: https://github.com/alibaba/nann

相关阅读： TDM 到二向箔：阿里妈妈展示广告 Match 底层技术架构演进

论文下载： CIKM 2022 | Approximate Nearest Neighbor Search under Neural Similarity Metric for Large-Scale Recommendation，https://arxiv.org/pdf/2202.10226.pdf

BOMGraph：基于统一图神经网络的电商多场景召回方法

作者：影书

1. 摘要

手机淘宝支持用户以多种形式来进行搜索，除了常用的文本搜索，还支持拍照搜索、相似商品搜索。不同场景之间在数据分布上存在许多共性和差异性。能否利用场景之间的共性来缓解单场景样本稀疏性问题，提升召回效果，同时避免统一建模对于各场景差异化建模的影响。因此，本文提出了一种统一的基于图神经网络的召回方法（**BO**osting **M**ultiscenario E-commerce Search with a unified **Graph** neural network，**BOMGraph**），BOMGraph 包含几个组件来解决上述多场景建模存在的挑战。首先是在节点图卷积的时候通过场景内和场景间的 metapath 来传播跨场景之间的异构信息。其次，我们提出了一个解耦网络来为商品提取场景公共和独有的表示，显式的建模不同场景之间的共性和差异性。最后，通过基于跨场景的样本增强和对比学习，来解决商品在单个场景由于长尾和样本稀疏导致学习不充分的问题。离线评估和在线 A/B 测试均证明了 BOMGraph 的有效性，目前该方案已在搜索广告在线业务中投入使用。

论文：【CIKM 2023】BOMGraph: Boosting Multi-scenario E-commerce Search with a Unified Graph Neural Network

作者： Shuai Fan, Jinping Gou, Yang Li, Jiaxing Bai, Chen Lin, Wanxian Guan, Xubin Li, Hongbo Deng, Jian Xu, Bo Zheng

下载（点击↓阅读原文）： https://dl.acm.org/doi/10.1145/3583780.3614794

2. 背景



图 1　多模态多场景联合搜索示例

如图 1 所示，用户可以在首页进行文本搜索和拍照搜索，也可以在搜索结果页长按商品触发相似商品搜索。不同搜索方式从不同的入口发起，并且接受不同模态的查询输入，包括文本、图片和商品。

此外，用户还会交替使用不同的搜索方式。例如图 1(b) 中，用户想要购买一件连衣裙，从拍照搜索切换到文本搜索和相似商品搜索。由于每个搜索场景会侧重不同的模态信息，因此在交替搜索的过程中，能够更全面的表达用户的偏好（即“粉色、法式碎花、吊带裙、奢华面料”）。

这种多个搜索场景之间的关联性为多场景建模提供了机会。最新研究也表明，利用多场景数据联合建模可以提高单个场景的性能，缓解单场景的样本稀疏问题。本文核心目标是为手淘搜索的多个场景设计一套统一的图神经网络召回方法，其中主要面临以下 3 个挑战：1）如何捕捉不同场景的异构信息传递；2）如何学习场景鲁棒的商品表示；3）如何缓解单场景长尾商品样本稀疏问题。

3. 方法

BOMGraph 使用多个场景数据来构建查询、触发商品和商品的大规模异构图，并定义元路径来更新图中节点的信息，异构图通过多场景图编码器进行编码，该编码器通过元路径引导的信息传播来获得节点嵌入，捕捉不同场景之间的异构信息传递。然后将经过图编码的商品节点嵌入通过解耦表示模块进行解耦，学习场景鲁棒的商品表

示。最后，通过在训练过程中加入跨场景数据扩充和对比学习损失缓解单场景长尾商品点击不足学习不充分的问题。接下来介绍统一图神经网络召回框架的细节。

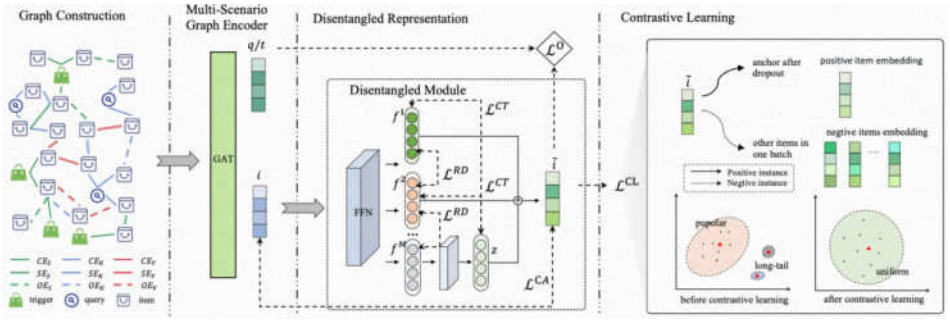


图 2　BOMGraph 整体框架

3.1 异构图构建

我们首先构造了一个异构图 $\mathcal{G}=\{\mathcal{V},\mathcal{E}\}$. $\mathcal{V}$ 是节点集合，每个节点都用一个 D 维的向量表示 . $\mathcal{G}$ 中有三种类型的节点，t 表示触发商品节点，q 表示查询节点，i 表示商品节点。

我们在 3 个场景中分别建模了点击和共现两种关系，其中 S 表示相似商品搜索，K 表示文本搜索，V 表示拍照搜索，总共有七种类型的边 $\{CE,\{OE_o\},\{SE_o\}\mid o\in\{S,K,V\}\}$。其中 CE 是点击边，连接触发商品 / 查询和商品。OE_o，$o\in S$，K，V 是共现边，表示同一个场景的同一次搜索会话下两个被同时点击的商品。SE_o，$o\in\{S$，K，$V\}$ 是相似边，表示同一个场景下两个展现商品图文内容相似。

3.2 基于场景的元路径 (meta-path)

然后我们分别定义了场景内元路径以及场景间元路径去对当前节点游走产生子图，通过在子图上使用图编码器来得到融合不同场景信息的表示。

场景内元路径是在每个搜索场景中定义的，动机是沿着场景内元路径收集特定场景的协作反馈，并允许信息传输到相关查询和商品。例如在相似商品搜索场景 S 和文本搜索场景中 K，我们定义了连接查询（例如触发商品 t 和文本查询 q）和商品（即 i）的元路径，沿着点击边 CE 构建的场景内元路径：

$$
\begin{aligned}
M_t^S &= t \xrightarrow{CE} i \xrightarrow{CE} t', \\
M_i^S &= i \xrightarrow{CE} t \xrightarrow{CE}, i', \\
M_q^K &= q \xrightarrow{CE} i \xrightarrow{CE} q', \\
M_i^K &= i \xrightarrow{CE} q \xrightarrow{CE} i'.
\end{aligned}
\quad （1）
$$

与现有的基于异构网络的检索模型不同，除了场景内元路径外，我们还定义了场景间元路径来捕获跨场景的信息。动机是通过相似性边将不同场景中内容相似的商品连接起来，这样信息就可以跨场景流动，并利用场景间的共性。具体的，我们在三个场景中以不同的顺序定义了三个元路径：

$$
\begin{aligned}
M_i^{S-K-V} &= i \xrightarrow{SE_K} i' \xrightarrow{SE_V} i'', \\
M_i^{K-V-S} &= i \xrightarrow{SE_V} i' \xrightarrow{SE_S} i'', \\
M_i^{V-S-K} &= i \xrightarrow{SE_S} i' \xrightarrow{SE_K} i''.
\end{aligned}
\quad （2）
$$

3.3 异构图编码器

对于每个节点 $v \in \mathcal{V}$，其中 v 可以是触发商品 / 查询 / 商品节点，我们沿着每个场景的场景内元路径进行采样，该路径从 v 开始形成场景内子图 $G_v^{intra,o}|o \in \{ S， K， V \}$。我们还形成了场景间子图 $G_v^{inter,o}|o \in \{ S， K， V \}$，方法是沿着从场景 o 开始的每个场景间元路径选择节点。具体来说，我们分别对一跳邻居中的三个节点和二跳邻居中三个节点进行随机采样。为了防止误采样产生偏差，如果采样的邻居在小批量中构成一对正对，我们将其丢弃并重新采样，直到达到样本大小。

然后我们使用异构图编码器依据采样的子图对查询节点和商品节点进行编码，先前的工作指出，GAT（图注意力网络）的聚合效果优于 GCN，因此我们使用 GAT 作为图编码器的主网络。我们定义 $\mathbf{v} = GAT(\mathbf{v}^0, G_v)$ 为节点 v 通过 GAT 在子图 G_v 上的嵌入，其中 $\mathbf{v}^0$ 是初始化的嵌入向量。

为了初始化节点嵌入 $\mathbf{i}^0$ 和触发节点嵌入 $\mathbf{t}^0$，我们将节点 ID 特征的 One-Hot 嵌入、图像特征嵌入和标题文本的特征嵌入连接起来，并通过全连接层传递它们。初始查询节点嵌入是通过连接 ID 嵌入 $\mathbf{q}^0$ 和文本特征嵌入获得的。

对于查询 q 或触发节点 t，由于它们的表示仅用于特定场景（即 q 用于文本搜索 K 和 t 用于相似商品搜索 S），我们应该关注场景内信息。因此，它们的嵌入是以特定场景的方式在场景内元路径引导的子图上获得的，如下所示：

$$\begin{aligned}\mathbf{q} &= W * GAT^{intra}(\mathbf{q}^0, G_q^{intra,K}),\\ \mathbf{t} &= W * GAT^{intra}(\mathbf{t}^0, G_t^{intra,S}).\end{aligned} \quad （3）$$

由于三个场景共享相同的商品，在获取商品表示时，我们考虑了场景内和场景间的信息。我们首先获取每个场景下的商品表示。我们使用三个 GAT 编码器，来组合来自场景内、场景间和场景共享信息。

$$\begin{aligned}\mathbf{i}^{intra,o} &= GAT^{intra,o}(\mathbf{i}^0, G_i^{intra,o}) + GAT^{shared}(\mathbf{i}^0, G_i^{intra,o}),\\ \mathbf{i}^{inter,o} &= GAT^{inter,o}(\mathbf{i}^0, G_i^{inter,o}),\\ \mathbf{i} &= W[\mathbf{i}^{inter,o}||\mathbf{i}^{intra,o}], o \in \{S,K,V\}.\end{aligned} \quad （4）$$

在异构图编码模块中，我们首先通过场景内和场景间的元路径聚合邻居信息，得到查询节点和商品节点的融合多个场景的粗粒度表示，之后再通过解耦表示学习模块来学习更加鲁棒的表示。

3.4 解耦表示学习

我们保持异构图编码器输出的查询嵌入和触发商品嵌入不变，并将所得到的商品嵌入送到解耦表示模块，该模块由三部分组成。

我们认为，更鲁棒的商品表示必须反映场景之间的共性和细微差异，而不是简单地将信息融合到各个场景中。

因此，我们通过将图学习得到的的商品表示解耦为 M 个独立的对应项来微调它。这些对应项可以对应于不同场景所关注的特定特征，解耦表示如下：

$$\mathbf{f}^m = \frac{\sigma(W^m\mathbf{i} + b^m)}{\|\sigma(W^m\mathbf{i} + b^m)\|_2}, m = 1, 2, \ldots, M, \quad （5）$$

然后，我们使用对应项来推导场景之间的共性。这可以通过转换一个对应物并对其进行调整以反映共同知识来实现。在不失一般性的情况下，我们可以将变换应用于最后一个正交特征空间 $\mathbf{f}^M$。也就是说，我们向前馈层输入 $\mathbf{f}^M$，如下所示：

$$\mathbf{z} = W\mathbf{f}^M + b, \quad （6）$$

由于 $\mathbf{z}$ 反映了不同场景之间的共同信息，因此它应该在 $\mathbf{f}^m$ 中捕获共同信息。受域对齐损失的启发，我们调节 $\mathbf{z}$ 来捕获其他对应项的元素方差：

$$\mathcal{L}^{CT} = \frac{1}{M-1}\frac{1}{F^2}\sum_{\mathbf{i}\in\mathcal{B}}\sum_{m=1}^{M-1}\sum_{(a,b)}(\mathbf{f}_a^m \cdot \mathbf{f}_b^m - \mathbf{z}_a \cdot \mathbf{z}_b), \quad (7)$$

我们通过合并解耦的对应表示 $\mathbf{f}^m$ 和公共信息 $\mathbf{z}$ 来微调商品嵌入：

$$\bar{\mathbf{i}} = \sum_{m=1}^{M-1}\mathbf{f}^m + \mathbf{z}. \quad (8)$$

为了获得更稳定的性能，我们可以鼓励细化的商品嵌入 $\bar{\mathbf{i}}$ 的质心与原始商品嵌入 $\mathbf{i}$ 的质心对齐，避免 $\bar{\mathbf{i}}$ 偏离到遥远的区域：

$$\mathcal{L}^{CA} = \sum_{\mathbf{i}\in\mathcal{B}} \| \frac{1}{|\mathcal{B}|}\sum_{\mathbf{i}\in\mathcal{B}}\mathbf{i} - \frac{1}{|\mathcal{B}|}\sum_{\mathbf{i}\in\mathcal{B}}\bar{\mathbf{i}} \|_2. \quad (9)$$

最终解耦阶段的 loss 组成为解耦损失以及 CT 损失还有 CA 损失。

3.5 对比学习

不同场景下的长尾商品不同，当前场景的长尾商品可能在另一场景下行为充分，因此可以联合其他场景的数据来补充行为数据。因此，我们在数据级别实现了跨场景扩充，使用场景之间相似度高的商品来补充当前场景的训练样本。在当前场景中，如果查询商品对 $(q,\ i)$，连接到点击次数少于三次的商品 i，则我们使用相似性边 SE 来查找与其他场景中的 i 高度相似的商品 i'。我们根据它们的相似性选择前五个商品 i'。这前 5 项被构造为新的查询项对 $(q,\ i')$。

我们优化了成对监督损失。对于查询 q 或触发商品 t，在查询下单击的商品 i^+（或通过如上所述的跨场景数据扩充构建的商品）被视为正样本。我们随机抽取同类目下 M 个其他商品 i^- 作为负样本。监督学习中的 InfoNCE 损失定义如下：

$$\mathcal{L}^{O} = \sum_{\mathbf{x},\bar{\mathbf{i}}^+\in\mathcal{B}} -\log\frac{\exp(\mathrm{sim}(\mathbf{x},\bar{\mathbf{i}}^+)/\tau)}{\sum_{\bar{\mathbf{i}}^-\in\mathcal{N}^-}\exp(\mathrm{sim}(\mathbf{x},\bar{\mathbf{i}}^-)/\tau) + \exp(\mathrm{sim}(\mathbf{x},\bar{\mathbf{i}}^+)/\tau)}, \quad (10)$$

GNN 在表示一致性方面的性能较差，并且由于更加流行的商品可能聚集在一起，因此很难检索长尾商品。受此启发，我们还利用对比学习来解决长尾问题。对比学习的动机是缩短锚样本和正样本之间的距离，同时增加锚样本与负样本之间的间距。通过使用对比学习，样本在特征空间中的分布更加均匀。形式上，我们定义对比损失：

$$\mathcal{L}^{CL} = \sum_{\bar{\mathbf{i}} \in \mathcal{B}} -\log \frac{\exp(\text{sim}(\bar{\mathbf{i}}, \bar{\mathbf{i}}^{+})/\tau)}{\sum_{\bar{\mathbf{i}}^{-} \in \mathcal{B}/\{\bar{\mathbf{i}}\}} \exp(\text{sim}(\bar{\mathbf{i}}, \bar{\mathbf{i}}^{-})/\tau) + \exp(\text{sim}(\bar{\mathbf{i}}, \bar{\mathbf{i}}^{+})/\tau)}, \quad (11)$$

4. 实验分析

我们通过计算后一天点击商品的(NDCG)@100/200，(HR)@100/200，和(MRR)@100/200来评估在多个场景的实际召回效果。

离线主实验部分，我们分别在三个场景对比相应的baseline模型：

1. BOMGraph-SKV在所有搜索场景中始终取得最佳结果，这证明了多场景搜索联合建模的优越性。
2. 我们观察到结合更多的搜索场景可以提高BOMGraph的性能。例如，组合了三个场景的BOMGraph-SKV优于在每个数据集上组合两个场景的模型。然而，即使两个场景数据，BOMGraph也明显优于最佳竞争对手LasGNN。这一观察结果验证了我们的假设，即在统一的框架中多场景联合学习可以提高每个场景的性能。
3. 此外，我们观察到，单一场景的LasGNN优于结合三种场景的MS-Graph-SKV。这表明仅仅融合多个场景是不够的，也不一定比单一场景学习产生更好的结果。同时，LasGNN的性能优于GraphSage，根本原因是基于元路径采样的子图比随机采样的子图更有效地聚合节点信息。

Scenario	Model	HR@100	HR@200	MRR@100	MRR@200	NDCG@100	NDCG@200
Similar product search (S)	GraphSage	0.4359	0.5134	0.0777	0.0782	0.1465	0.1573
	AdaptiveGCN	0.4434	0.5138	0.825	0.0830	0.1531	0.1630
	LasGNN	0.4805	0.5466	0.1103	0.1107	0.1831	0.1924
	MSGraph-SKV	0.4713	0.5382	0.1003	0.1005	0.1715	0.1809
	BOMGraph-SK	0.5314	0.5938	0.1231	0.1235	0.2042	0.2129
	BOMGraph-SV	0.5234	0.5872	0.1207	0.1212	0.2005	0.2094
	BOMGraph-SKV	**0.5363**	**0.5976**	**0.1262**	**0.1266**	**0.2077**	**0.2163**
Textual search (K)	GraphSage	0.3967	0.4887	0.0739	0.0747	0.1279	0.1401
	AdaptiveGCN	0.4027	0.4907	0.0742	0.0749	0.1357	0.1481
	LasGNN	0.4141	0.5019	0.0746	0.0791	0.1359	0.1479
	MSGraph-SKV	0.4089	0.4979	0.0721	0.0776	0.1332	0.1457
	BOMGraph-SK	0.4344	0.5212	0.0851	0.0858	0.1513	0.1634
	BOMGraph-SKV	**0.4382**	**0.5237**	**0.0871**	**0.0877**	**0.1538**	**0.1657**
Visual search (V)	GraphSage	0.2633	0.3181	0.0511	0.0517	0.0897	0.0913
	AdaptiveGCN	0.2721	0.3275	0.0521	0.0528	0.0942	0.0964
	LasGNN	0.2906	0.3434	0.0574	0.0577	0.1021	0.1095
	MSGraph-SKV	0.2887	0.3409	0.0553	0.0561	0.1001	0.1079
	BOMGraph-SV	0.3526	0.4066	0.0691	0.0695	0.1246	0.1331
	BOMGraph-SKV	**0.3581**	**0.4189**	**0.0733**	**0.0737**	**0.1275**	**0.1351**

表1　离线主实验结果

线上 AB 实验部分，我们进行了在线 A/B 测试，将我们提出的模型 BOMGraph 与线上解决方案（单场景模型 LasGNN）的性能进行比较。在为期七天的实验中，我们只修改了候选商品生成步骤，即在相似商品搜索场景中，使用不同的模型来召回每个触发商品查询的前 200 个商品，保持所有其他排序预估的步骤不变。我们比较了在线实际点击率 (CTR)、转化率 (CVR) 和每千次点击扣费收入 (RPM)，以确定 BOMGraph 召回的商品是否更有可能被用户点击和购买。

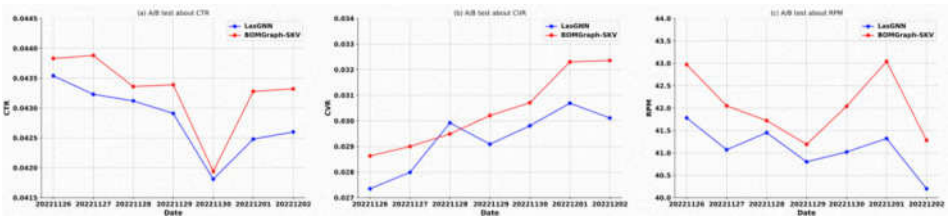


图 3　在线 AB 实验结果

如图所示，在相似商品搜索场景中，我们提出的 BOMGraph-SKV 在 CTR 和 RPM 方面始终优于 LasGNN。就 CVR 而言，它只落后于 LasGNN 一天。这可能是因为，虽然我们的模型能够每天提高点击率，但并不一定总是能提高转化率，总体而言，我们的 CVR 仍有提升效果。

在论文中我们还分别验证了不同模块的消融实验，以及相关超参数实验，具体细节可以参考我们的论文。

5. 总结

在本文中，我们提出一种基于电商搜索多场景联合建模的统一图神经网络召回框架，用于文本搜索、拍照搜索、相似商品搜索三个场景，利用来自多个场景的数据并联合优化整体性能，实验证明我们的方法在多个场景都能够取得效果上的提升。未来工作也将持续关注电商搜索场景下的多场景联合学习。

参考文献

[1] Xinyu Zou, Zhi Hu, Yiming Zhao, Xuchu Ding, Zhongyi Liu, Chenliang Li, and Aixin Sun. 2022. Automatic Expert Selection for Multi-Scenario and Multi-Task Search. In SIGIR. 1535 - 1544.

[2] Qijie Shen, Wanjie Tao, Jing Zhang, Hong Wen, Zulong Chen, and Quan Lu. 2021. SAR-Net: A Scenario-Aware Ranking Network for Personalized Fair Recommendation in Hundreds of Travel Scenarios. In CIKM. 4094 - 4103.

[3] Yuting Chen, Yanshi Wang, Yabo Ni, Anxiang Zeng, and Lanfen Lin. 2020. Scenario-aware and Mutual-based approach for Multi-scenario Recommendation in E-Commerce. In ICDM (Workshops). 127 - 135.
[4] Shaohua Fan, Junxiong Zhu, Xiaotian Han, Chuan Shi, Linmei Hu, Biyu Ma, and Yongliang Li. 2019. Metapath-Guided Heterogeneous Graph Neural Network for Intent Recommendation. 2478 - 2486.
[5] Yichao Wang, Huifeng Guo, Bo Chen, Weiwen Liu, Zhirong Liu, Qi Zhang, Zhicheng He, Hongkun Zheng, Weiwei Yao, Muyu Zhang, Zhenhua Dong, and Ruiming Tang. 2022. CausalInt: Causal Inspired Intervention for Multi-Scenario Recommendation. In KDD. 4090 - 4099.
[6] Jiangxia Cao, Xixun Lin, Xin Cong, Jing Ya, Tingwen Liu, and Bin Wang. 2022. DisenCDR: Learning Disentangled Representations for Cross-Domain Recommendation. In SIGIR. 267 - 277.
[7] Menghan Wang, Yujie Lin, Guli Lin, Keping Yang, and Xiao-Ming Wu. 2020. M2GRL: A Multi-task Multi-view Graph Representation Learning Framework for Web-scale Recommender Systems. In KDD. 2349 - 2358.
[8] Xiang Wang, Hongye Jin, An Zhang, Xiangnan He, Tong Xu, and Tat-Seng Chua. Disentangled Graph Collaborative Filtering. In SIGIR. 1001 - 1010.
[9] Jianxin Ma, Chang Zhou, Hongxia Yang, Peng Cui, Xin Wang, and Wenwu Zhu. 2020. Disentangled Self-Supervision in Sequential Recommenders. In KDD. 483 - 491.
[10] Jiancan Wu, Xiang Wang, Fuli Feng, Xiangnan He, Liang Chen, Jianxun Lian, and Xing Xie. 2021. Self-supervised Graph Learning for Recommendation. In SIGIR. 726 - 735.
[11] Xiangnan He, Zhankui He, Xiaoyu Du, and Tat-Seng Chua. 2018. Adversarial Personalized Ranking for Recommendation. In SIGIR. 355 - 364.
[12] Junliang Yu, Hongzhi Yin, Xin Xia, Tong Chen, Lizhen Cui, and Quoc Viet Hung Nguyen. 2022. Are Graph Augmentations Necessary?: Simple Graph Contrastive Learning for Recommendation. In SIGIR. 1294 - 1303.
[13] Ruoyu Li, Sheng Wang, Feiyun Zhu, and Junzhou Huang. 2018. Adaptive Graph Convolutional Neural Networks. In AAAI. 3546 - 3553.

CC-GNN：基于内容协同图神经网络的电商召回方法

作者：影书

1. 摘要

在电商搜索系统中，普遍流行用图神经网络来做商品召回。这些模型效果虽然很好，但仍存在以下不足：1）没有充分利用商品的图文内容特征；2）在工业级大规模稀疏图结构上的训练效率不高；3）对于长尾查询和冷启动商品的预测不够准确。为了解决这些问题，本文提出了一种新型的基于内容协同的图神经网络（Content Collaborative Graph Neural Network，以下简称 CC-GNN）。首先，CC-GNN 从商品内容中抽取文本短语并将其显式的用于图传播，来捕捉商品之间的语义关系和流行趋势。其次，CC-GNN 提出了一个可扩展的图学习框架，来实现更高效的图神经网络训练，包括高效的图构建、基于 MetaPath 的消息传递机制和基于样本难度加噪的图对比学习方法。此外，CC-GNN 在监督学习和自监督学习中还采用反事实数据补充来解决长尾 / 冷启动问题。我们在上亿级别节点规模的真实电商数据集上进行了充分实验，实验结果表明，相比线上最新的图学习模型，CC-GNN 在整体效果、长尾查询和冷启商品召回上都有显著的效果提升。基于该项工作整理的论文已发表在 KDD 2023，欢迎阅读交流。

论文：E-commerce Search via Content Collaborative Graph Neural Network

下载（点击↓阅读原文）: https://dl.acm.org/doi/abs/10.1145/3580305.3599320

2. 背景

向量化检索是一种常用的商品召回方法，通常需要先把查询和商品编码为低维向量后再进行检索。而越来越多的模型使用图神经网络（GNN）来做编码。基于 GNN 的检索模型有两个优点：1）方便融合不同模态关系的数据。例如电商搜索下，用户既可以用文本关键词搜索，也可以用图片、商品搜索。无论是哪一种模态的查询，都可以通过在图中定义一种新的节点和边来描述。2）对于行为反馈稀疏的冷启查询和长尾商品有更好的性能。由于 GNN 会通过多层卷积不断汇聚相邻节点的信息，因此在节点编码中包含了与其他非冷门商品的近似关系。

然而，电商召回长期以来一直面临着以下三个挑战，这些挑战尚未完全被基于 GNN 的模型解决，特别是在面对工业级规模的数据时。

挑战一：商品内容的语义表示

学习商品内容（例如标题）的语义表示非常重要，因为它是衡量查询和商品相关性的重要依据。标题中词组的含义通常是由商家及其售卖的商品所决定的，例如词组“2022 新款”的确切含义取决于它用于描述哪种商品。此外，词组的含义可能会发生变化（即语义漂移），并且与之相关的商品会受到显著影响。例如，“星光”的原始含义是来自星星的光，如果 MacBook Air 使用“星光”来描述银色，其他银色的电子产品也会倾向于使用“星光”来描述自己，”星光”的含义就变成了银色，发生了语义漂移。大多数现有研究（如图 1(a) 所示）都将内容特征作为节点属性，并且没有随着训练数据的变化来更新其语义表示，从而导致搜索性能的下降。

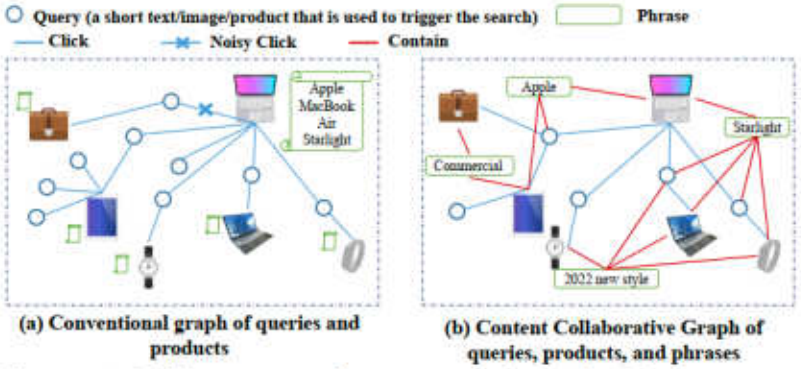


Figure 1: Different graph constructions for E-commerce search. (a) Graph built solely from user feedback, where product contents are treated as node attributes [24, 48]. (b) Content Collaborative Graph is presented in this paper, where carefully extracted phrases are nodes that explicitly involve in graph propagation.

图 1

挑战二：面向工业规模图的学习效率。

我们主要关注两个瓶颈问题：1）对于拥有大规模节点和边的图，每个节点只能与有限的邻居进行信息交换。先前的研究并未考虑到边的性质，因此其学习效率受到了噪声邻居（如图 1(a) 所示的无关邻居）的影响。2）将有监督学习与自监督对比学习相结合能够很好的增强检索模型性能。但是由于图增强来构建正样本需要很多计算量，在工

业级规模的图上是不可行的。

挑战三：长尾查询和冷启动商品。

如图 1(a) 展示的，大多数节点的度数偏低，而只有少数节点的度数较高。低度数的节点通常包括搜索用户较少的长尾查询和新上市的冷启动商品。对于这些低度节点，由于无法获得足够的边以进行有效的邻域聚合，传统的 GNNs 无法对其进行准确的表示。这一问题不仅阻碍了召回模型的整体性能，还导致了系统公平性问题。

3. 方法

我们提出了一种新颖的内容协同图神经网络（Content Collaborative Graph Neural Network，CC-GNN），用于同时解决上述三个问题。CC-GNN 构建了一个内容协同图（图 1(b)），同时建模了商品内容和用户反馈，让词组节点显式地参与图信息传播，以获得更准确的语义表达。同时，CC-GNN 提出了基于 MetaPath 的消息传递机制，帮助 GNN 捕捉每个节点更稳健的拓扑特征。在学习范式方面，CC-GNN 结合了监督学习和对比学习。在监督学习中，CC-GNN 为长尾查询和冷启动商品引入了反事实补充样本。而在对比学习中，为了提高训练效率，CC-GNN 提出了一种基于样本难度加噪的图对比学习方法，该方法较当前的 SOTA 图对比学习方法有着更低的计算复杂度。最后，CC-GNN 通过反事实对比学习进一步增强了长尾查询和冷启动商品的训练。

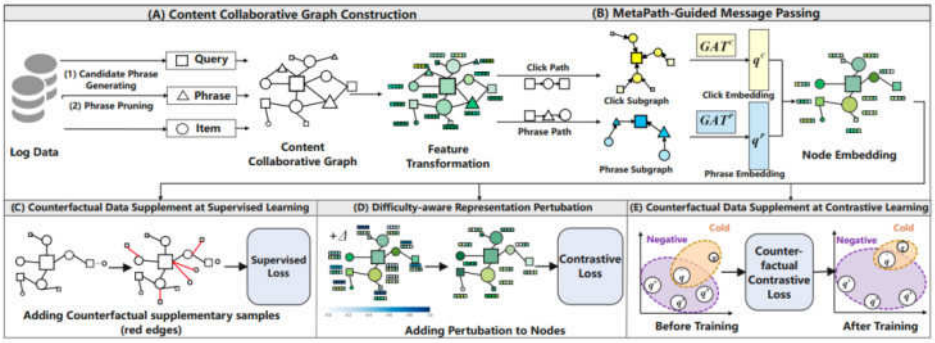


Figure 2: Overall framework of CC-GNN

图 2

3.1 内容协同图的构建

为了构建内容协同图 (CC-GNN)，我们需要添加内容节点。出于计算效率和语义信

息量的考量，我们从商品标题中提取内容词组。一方面，仅使用单词作为节点是有问题的，因为一些流行词（例如“新款”）的使用非常频繁，使得该节点与过多的邻居节点相连，导致过高的计算和存储开销。另一方面，我们并不需要列举所有的词组，有些信息不全、语义不清晰的词组反而会阻碍信息的传播。因此，我们提取了不定长的内容词组，具体词组的产生过程包括候选词组生成和词组剪枝两个步骤。

候选词组生成

直观地说，相似的商品通常会使用相同且有信息量的词组来描述某个商品。因此，我们选择基于历史交互日志来生成候选词组。首先，我们收集了查询 – 商品对、共同点击的查询对以及共同点击的商品对等历史交互对。我们从频次超过 3 次的交互对中提取共现的 N-gram 词组。为了增加覆盖率（即找到尽可能多的有信息量的词组），我们不限制词组的长度。为了避免冗余，我们对词组进行了重写，并按字母顺序重新排列词组中的单词。

词组剪枝

为了进一步减小词组的规模并加快图计算速度，我们会筛选掉那些过于模糊且与商品特征无关的词组。首先，我们使用一个电子商务命名实体识别（NER）工具来识别每个候选词组中的实体。这些实体大致包括类别词、风格词、材料词等。然后，我们根据一个人工定义的实体类型评分表对词组中每个识别出的实体进行评分。如果词组的总分低于预定义的阈值，则该词组将被删除。

3.2 基于 MetaPath 的消息传递

我们定义了两条元路径：点击路径和词组路径。对于查询节点 q，点击路径是 $q-t-q$，词组路径是 $q-p-t$。对于商品节点 t，点击路径是 $t-q-t$，词组路径是 $t-p-q$。点击路径和词组路径引导了不同的采样过程。我们将查询节点 $q \in Q$ 在点击路径上采样的子图表示为 $\mathcal{G}^{q,c}$，在词组路径上采样的子图表示为 $\mathcal{G}^{q,p}$。两个子图的节点大小相等，即对于路径上的每一跳，我们沿相应路径采样个 K 邻居节点。这保证了基于内容的查询 / 商品在图节点信息传播中发挥同等重要的作用。

我们采用两个图注意力网络（GAT）作为采样子图的聚合器。最终的查询表示是通过合并子图的节点表示获得的。

$$\mathbf{q}^c = GAT^c(\mathcal{G}^{q,c}), \quad \mathbf{q}^p = GAT^p(\mathcal{G}^{q,p}), \quad \mathbf{q} = MeanPooling(\mathbf{q}^c, \mathbf{q}^p),$$

其中 $GAT(\cdot)$ 表示使用 $LeakyReLU$ 激活函数的 L 层 GAT 的输出，$\mathbf{q^c}$ 代表基于点击子图 $\mathcal{G}^{q,c}$ 的查询节点表示，$\mathbf{p}^c$ 代表基于词组子图 $\mathcal{G}^{q,p}$ 的查询节点表示。相应的，我们可以获得商品节点的两个子图，以及对应的商品节点表示。我们没有对词组节点进行子图采样，但词组节点的表示会随着查询和商品节点在 GAT 计算中更新。

3.3 监督学习中的反事实数据补充

在监督学习下，长尾查询节点和冷启动商品节点由于缺乏有监督正样本而学得不好。为了解决这个问题，我们生成反事实补充样本。对于长尾查询节点，我们首先根据内容特征的余弦相似度检索 M 个相似商品。然后，我们根据点击次数从中抽取 $\lceil M/10 \rceil$ 个商品。由此，我们添加了最有可能被点击的商品，生成小批次 $\tilde{\mathcal{B}}$。同样，我们首先选择 M 个相似查询并根据点击次数采样 $\lceil M/10 \rceil$ 个查询 ，将查询添加到冷启动商品的小批次 $\hat{\mathcal{B}}$ 中。

反事实数据补充的结果如图 2(A) 所示。对于图中较小的节点（即长尾查询和冷启动商品），反事实数据将补充更多节点与长尾查询和冷启动商品（红色边）相连接。由于反事实补充样本不是真实点击，我们在监督损失中加入了置信度来控制每个样本的影响：

$$\mathcal{L}_{sup} = \sum_{(q,t)\in\mathcal{B}\bigcup\tilde{\mathcal{B}}\bigcup\hat{\mathcal{B}}} -\theta_{q,t} log \frac{exp\big(sim(\mathbf{q},\mathbf{t})/\tau\big)}{exp\big(sim(\mathbf{q},\mathbf{t})/\tau\big) + \sum_{t'} exp\big(sim(\mathbf{q},\mathbf{t}')/\tau\big)},$$

其中 (q,t) 是真正的正对或反事实补充对，t' 是查询 q 的负样本，τ 是温度，$sim(\mathbf{q},\mathbf{t})$ 是节点表示的余弦相似度，θ 表示置信度。

$$\theta_{q,t} = \begin{cases} 1, & \text{if } (q,t) \in \mathcal{B} \\ cos(\mathbf{f}^q, \mathbf{f}^t), & \text{if } (q,t) \in \tilde{\mathcal{B}}\bigcup\hat{\mathcal{B}}. \end{cases}$$

3.4 基于样本难度加噪的图对比学习

为了提升图对比学习的效率，我们提出了基于样本难度加噪的图对比学习方法（DARP）。对于每个 anchor 查询 $\mathbf{q}$，我们通过在 $\mathbf{q}$ 上添加一些扰动来获得正样本。扰动的强度取决于节点的度数，即归一化点击次数 $NC(q)$ 和归一化短语数 $NP(q)$。

$$\bar{\mathbf{q}} = \frac{MeanPooling\big(\mathbf{q} + \Delta_q^c \times NC(q), \mathbf{q} + \Delta_q^p \times NP(q)\big)}{||MeanPooling\big(\mathbf{q} + \Delta_q^c \times NC(q), \mathbf{q} + \Delta_q^p \times NP(q)\big)||_2^2},$$

其中，$NC(q)=\rho_q^c/\max_q \rho_q^c$，$NP(q)=\rho_q^p/\max_q \rho_q^p$，$\rho_q$ 是将点击或短语连接映射到间隔的分段函数。$\bar{\mathbf{q}}$ 使用了 L_2 归一化。扰动向量生成如下：

$$\bar{\Delta^x} \sim \mathbb{U}(0,1), \quad \Delta_q^x = \bar{\Delta^x} \odot sign(\mathbf{q}), \quad s.t. \|\Delta_q^x\|_2 = \epsilon.$$

其中 $\bar{\Delta^x}$ 是一个随机采样的 D 维向量，$\odot$ 是逐元素乘法，$sign(\mathbf{q})$ 表示 $\mathbf{q}$ 中每个元素的 sign 值。该过程如图 2(D) 所示，生成增强数据并添加到节点嵌入中。较小的节点获得较大的扰动（即小节点的颜色在扰动后变化更显著）。我们删除小批次中的冗余查询并形成一组不同的查询 $\bar{\mathcal{B}}^q$。在集合中查询 q 的对比损失如下：

$$\mathcal{L}_{cl}^q = -log\frac{exp(\bar{\mathbf{q}}^T\bar{\mathbf{q}}^{'}/\tau)}{\sum_{g\in\bar{\mathcal{B}}^q} exp(\bar{\mathbf{q}}^T\bar{\mathbf{g}}/\tau)}$$

其中 $\bar{\mathbf{q}}$ 和 $\bar{\mathbf{q}}^{'}$ 是对同一查询 q 的两次不同增强，$\bar{\mathbf{g}}$ 是不同查询的增强，作为负样本。对比损失最大化同一查询不同增强表示之间的一致性，同时最小化不同查询之间的一致性。类似的，从小批次中删除冗余商品可以得到一个不同的商品集合 $\bar{\mathcal{B}}^t$。我们对商品 t 进行类似的扩充。

$$\bar{\mathbf{t}} = \frac{MeanPooling\big(\mathbf{t}+\Delta_t^c \times NC(t), \mathbf{t}+\Delta_t^p \times NP(t)\big)}{||MeanPooling\big(\mathbf{t}+\Delta_t^c \times NC(t), \mathbf{t}+\Delta_t^p \times NP(t)\big)||_2^2},$$

其中，$NC(t)=\rho_t^c/\max_t \rho_t^c$，$NP(t)=\rho_t^p/\max_t \rho_t^p$。在集合中商品的对比损失如下：

$$\mathcal{L}_{cl}^t = -log\frac{exp(\bar{\mathbf{t}}^T\bar{\mathbf{t}}^{'}/\tau)}{\sum_{h\in\bar{\mathcal{B}}^t} exp(\bar{\mathbf{t}}^T\bar{\mathbf{h}}/\tau)},$$

其中 $\bar{\mathbf{t}}$ 和 $\bar{\mathbf{t}}^{'}$ 是对同一商品 t 的两次不同增强，$\bar{\mathbf{h}}$ 是不同商品的增强，作为负样本。最终基于加噪的自监督对比损失如下：

$$\mathcal{L}_{cl} = \sum_{q\in\mathcal{B},t\in\mathcal{B}} \big(\mathcal{L}_{cl}^q + \mathcal{L}_{cl}^t\big).$$

复杂度分析

Table 1: Computational complexity of existing graph contrastive learning methods, E: number of edges, D: embedding size, L: number of convolutional layers and R: dropout rate.

Augmentation Type	Model	Computational Complexity	
		Adjacency Matrix	Graph Convolution
Augmentation on Structure	SGL	$\Theta(2E+4RE)$	$\Theta((2+4R)ELD)$
	GCA	$\Theta(2E+4RE)$	$\Theta((2+4R)ELD)$
Augmentation on Embedding	SimGCL	$\Theta(2E)$	$\Theta(6ELD)$
	COSTA	$\Theta(2E)$	$\Theta(4ELD)$
	DARP	$\Theta(2E)$	$\Theta(2ELD)$

表 1

图对比学习的计算复杂度主要受两部分影响，计算邻接矩阵和图卷积，如表 1 所示。在计算邻接矩阵时，难度感知表示扰动（DARP）与其他 embedding 增强方法具有相同的计算复杂度，比结构增强方法具有更小的复杂度。基于 embedding 增强方法（即 SimGCL、COSTA 和 DARP）不会改变原始的图结构，因此它们只需要在输入图卷积之前对邻接矩阵进行归一化，其复杂度与边的数量成比例。对于结构方法的增强（即 SGL 和 GCA），它们需要额外的计算通过丢弃率来随机丢弃边，因此计算复杂度较大。

在图卷积阶段，DARP 的计算复杂度最小。图卷积的复杂度与卷积层数、边数、嵌入大小和增强图数有关。每个图都有 $2ELD$ 复杂度。(1) 对于 SGL 和 GCA，它们有一个原始图和两个增广图，因此复杂度为 $(2+4R)ELD$。(2) SimGCL 向每个图卷积层中的 embedding 添加噪声，这意味着它将卷积计算增加了三倍 $6ELD$。(3) COSTA 使用保持协方差的特征空间增强来构建单视图的正样本，这意味着它会使卷积计算翻倍 $4ELD$。(4) DARP 在图卷积之后增强表示，不改变卷积的复杂度 $2ELD$。总的来说，DARP 比现有的图对比学习方法具有更小的计算复杂度。

3.5 对比学习中的反事实数据补充

使用对比学习中反事实数据补充（CDS-CL）的动机是从真假相关性的角度处理长尾和冷启动问题。我们认为，真实相关性对应于与内容相关的商品，用户将在同一查询下点击这些商品。虚假相关性对应于不相关的商品，但模型基于反馈的虚假相关性做出错误的预测。

以前的工作主要集中在学习真实的相关性，很少考虑虚假的相关性。我们在对比学

习中使用反事实数据补充（CDS-CL）来同时学习真实相关性并消除虚假相关性的影响。（1）我们在表示空间中拉近成对的 < 头部商品，尾部商品 > 以学习真实相关性。（2）我们将具有相同点击次数水平的不同商品的表示分开，以消除虚假相关性的影响。

对于每个冷启动商品 $\hat{t}$，我们首先检索相似的热门商品 $\dot{t}$，得到一个热门商品集。然后，我们随机采样一个相似的热门商品作为正样本，$\hat{t}$ 应该与 $\dot{t}$ 在表示空间上相似。对于每个热门商品 $\dot{t}$，我们采样了 N 个负样本 t'，它们有相同的点击频次，但内容不相似。商品的反事实对比损失是：

$$\mathcal{L}_{cl_c}^{t} = -log\frac{exp(sim(\hat{\mathbf{t}}, \dot{\mathbf{t}})/\tau)}{exp(sim(\hat{\mathbf{t}}, \dot{\mathbf{t}})/\tau) + \sum_{t'} exp\big(sim(\dot{\mathbf{t}}, \mathbf{t}')/\tau\big)},$$

其中，$\hat{t}, \dot{t}$ 互为正样本，t' 是负样本，τ 是温度，sim() 是节点表示的余弦相似度。

同样，对于每个长尾查询 $\tilde{q}$，我们检索相似的头部查询并形成一个头部查询集，并随机抽取一个头部查询 $\ddot{q}$。我们采样了 N 个负样本 q'，它们具有相同的点击区间，但内容不相似。查询的反事实对比损失是：

$$\mathcal{L}_{cl_c}^{q} = -log\frac{exp(sim(\tilde{\mathbf{q}}, \ddot{\mathbf{q}})/\tau)}{exp(sim(\tilde{\mathbf{q}}, \ddot{\mathbf{q}})/\tau) + \sum_{q'} exp\big(sim(\ddot{\mathbf{q}}, \mathbf{q}')/\tau\big)},$$

其中，$\tilde{q}, \ddot{q}$ 互为正样本，q' 为负样本。反事实对比损失是：

$$\mathcal{L}_{cl_c} = \sum_{q \in \mathcal{B}, t \in \mathcal{B}} \big(\mathcal{L}_{cl_c}^{q} + \mathcal{L}_{cl_c}^{t}\big).$$

CC-GNN 最终的损失包括监督损失、对比损失和反事实对比损失三部分：

$$\mathcal{L} = \mathcal{L}_{sup} + \alpha\mathcal{L}_{cl} + \beta\mathcal{L}_{cl_c},$$

其中 α, β 会将几种损失调整到相同的量级。

4. 实验

4.1 数据集

我们使用了一个工业商品查询搜索数据集（Industryscale Product Query Search，简称 IPQS）。该数据集是通过收集电子商务平台上 91 天的日志数据构建而成。IPQS 数据集中的每条记录对应一个商品查询和一个被点击的商品，包含有关查询和商品的必要信息，包括商品 ID、类别、价格、销量、图片、标题等。我们从前 90 天

的日志数据中随机采样了 3500 万个查询、8700 万个商品和 7.097 亿个交互作为训练数据，使用最后一天的日志数据进行测试。我们移除在训练集中没有出现的商品。

4.2 整体召回性能评估

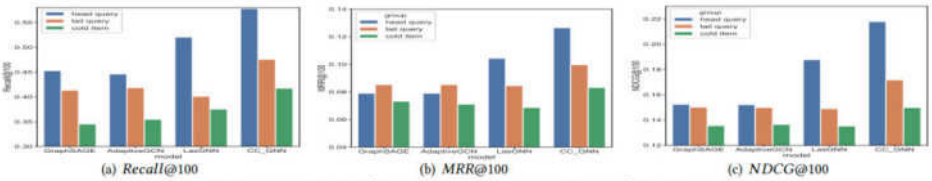
Table 2: Overall ranking results on IPQS

Model	Recall@100	MRR@100	NDCG@100
GraphSAGE	0.4441	0.0819	0.1528
AdaptiveGCN	0.4434	0.0825	0.1531
LasGNN	0.4805	0.1038	0.1759
CC-GNN	**0.5450**	**0.1189**	**0.2046**

表 2

根据表 2 所示，CC-GNN 在所有评估指标上明显对比方法。CC-GNN 相对于最佳基线（即 LasGNN），在整体的 $Recall@100$、$MRR@100$ 和 $NDCG@100$ 方面分别提高了 11.8%、14.5% 和 16.3%。

4.3 长尾和冷启动问题



(a) *Recall*@100 (b) *MRR*@100 (c) *NDCG*@100

Figure 3: Comparative performance on different groups of queries and items

图 3

1. 根据图 3 的观察结果，我们可以看到所有模型在头部查询和长尾查询之间存在着较大的性能差距。所有基线模型在长尾查询上的表现非常接近。然而，CC-GNN 显著提升了长尾查询的性能。具体而言，CC-GNN 在长尾查询上的性能优于 AdaptiveGCN 在头部查询上的性能。相比最佳基线，CC-GNN 在长尾查询上的 $Recall@100$、$MRR@100$ 和 $NDCG@100$ 分别提高了 13.7%、16.7% 和 14.2%。
2. 根据图 3 的观察结果，CC-GNN 极大地提升了冷启动商品的性能。相比于其他模型，CC-GNN 在冷启动商品上的 $Recall@100$、$MRR@100$ 和 $NDCG@100$ 分别提高了 11.1%，13.5%，9.8%。

4.4 词组节点的语义表示

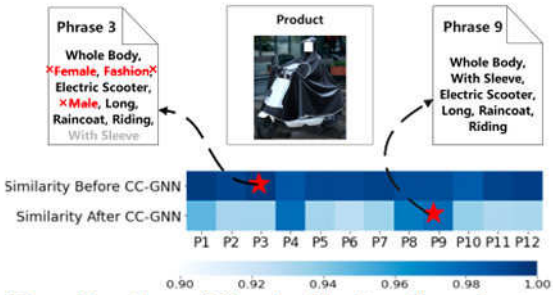


Figure 4: Visualization of the similarity of product and phrase embeddings (P1 to P12) before and after CC-GNN.

图 4

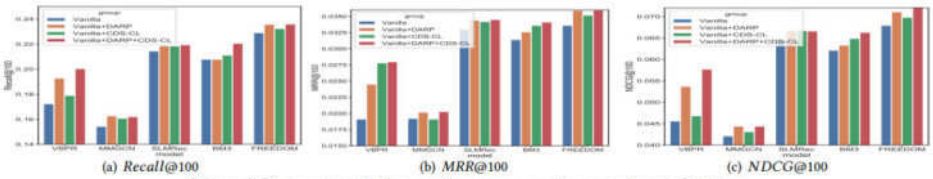
我们提供一个 case study，展示 CC-GNN 能够捕捉内容词组的语义。首先，我们随机从 IPQS 数据集中选择一个商品，并收集与该商品在内容协同图中连接的 12 个词组（从 P1 到 P12）。然后，我们提取这些词组在 CC-GNN 之前和之后的 embedding。接下来，我们计算商品 embedding 与 CC-GNN 之前和之后的词组 embedding 之间的余弦相似度。

如图所示，在 CC-GNN 之前，所有词组与商品的相似度都非常高（即大于 0.99）。需要注意的是，由于所有现有方法都不会更新词组的 embedding，这意味着所有现有方法都无法区分这些词组。相反，在 CC-GNN 之后，词组之间的差异变得更加明显。这表明 CC-GNN 可以根据电商训练数据调整词组的语义表达。在 CC-GNN 之前，与商品的相似度最高的词组是 Phrase3，在 CC-GNN 之后，相似度最高的是 Phrase9。通过分析这些词组，我们可以看到 Phrase3 包含一个非常普遍、噪声较大的词语“fashion”，它并不能描述商品的属性，并且缺少一个重要的特征词“with sleeve”，这是消费者购买雨衣时会考虑的一个重要特征。因此，Phrase3 确实是有噪声的，Phrase9 比 Phrase3 更好。CC-GNN 可以通过在图传播中学习词组语义来正确识别描述商品的最相关词组。

4.5 推荐模型上的性能评估

CC-GNN 包含多个组件，可以叠加在其他模型上，或应用于其他任务。我们提出的难度感知表示扰动（DARP）和对比学习中的反事实数据补充（CDS-CL）可以方便的插入到各种模型中，不限于图学习方法。此外，尽管 CC-GNN 的主要目的是解

决大规模的商品召回问题，但 DARP 和 CDS-CL 可以应用于其他任务。我们在亚马逊数据集上进行了多模态推荐系统的评估。选用的 Amazon Sports 数据集包含了 3.6 万个用户、1.8 万个商品和 29.6 万个交互记录，是一个广泛使用的推荐数据集。



(a) *Recall*@100 (b) *MRR*@100 (c) *NDCG*@100

Figure 6: Recommendation performance on Amazon Sports Dataset

图 5

我们将 DARP 和 CDS-CL 应用于基础推荐模型。为了更全面的对比，我们使用了五个具有不同模型架构和学习范式的代表性基础模型。我们观察到：

1. DARP 对比不同模型都有稳定的性能提升。特别是在监督模型（如 VBPR、MMGCN 和 FREEDOM）上，DARP 的改进更为显著，而在对比学习模型（如 SLMRec 和 BM3）上的改进较小。一个可能的原因是，DARP 旨在区分物品表示以提取额外的监督信号。
2. CDS-CL 也稳定提高了不同模型的性能。特别是，CDS-CL 在对比学习方法（如 SLMRec 和 BM3）上获得了更大的改进。对比学习关注物品本身及其增强方式，长尾 / 冷启动物品在数据集中的表示较少，并且它们的表示学习对其增强的效果不太敏感。相反，对比学习中的反事实数据补充旨在将知识从头部 / 热门物品转移到长尾 / 冷启动物品，因此可以进一步改善对比学习方法的性能。
3. 当结合 DARP 和 CDS-CL 时，在所有基础模型上可以获得最佳结果。这一现象表明，DARP 和 CDS-CL 可以组合应用在各种模型上，从不同角度获得最大的性能提升。

5. 总结

本文提出了一种高效的图学习方法 CC-GNN，在电商召回中表现出优越的性能，它捕获了内容语义迁移现象，缓解了长尾查询和冷启动商品训练和召回效果不佳的问题。CC-GNN 提出的模块，如内容协同图和反事实数据补充，可以应用到不同的 GNN 基线模型中，帮助其提升召回效果。此外，在公开数据集上的实验证明，我们的方法应用于推荐模型也可以获得性能提升。这些结果都表明，我们提出的方法有进

一步增强搜索和推荐系统性能的潜力。

参考文献

[1] Inductive Representation Learning on Large Graphs. NeurIPS 2017.
[2] DC-GNN: Decoupled Graph Neural Networks for Improving and Accelerating Large-Scale E-commerce Retrieval. WWW 2022.
[3] Self-supervised Graph Learning for Recommendation. SIGIR 2021.
[4] Are Graph Augmentations Necessary? Simple Graph Contrastive Learning for Recommendation. SIGIR 2022.
[5] COSTA: Covariance-Preserving Feature Augmentation for Graph Contrastive Learning. KDD 2022.
[6] Graph Contrastive Learning with Adaptive Augmentation. WWW 2021.

RGIB：对抗双边图噪声的鲁棒图学习

作者：零刻、周展科

摘要

链接预测 [1,2] 是图学习的一种基础任务，用于判断图中的两个节点是否可能相连，被广泛应用于药物发现、知识图谱补全和在线问答等实际场景。尽管图神经网络（Graph Neural Network，GNN）在该问题的性能上取得了显著进步，但在图结构噪声下的差强人意的鲁棒性仍是当前深度图模型的实际瓶颈。

在鲁棒图学习方面，早期工作探索了通过邻近节点的平滑效果来提高 GNN 在节点标签噪声下的鲁棒性，其他方法通过随机移除边或主动选择有信息量的节点或边来达到类似的效果。然而，当将这些抗噪声方法应用于带有噪声的链接预测时，只能取得非常有限的增益。其原因在于，**不同于标签噪声，这里的图结构噪声是双向的：它会自然地同时扰动输入图的拓扑结构和输出端目标边的标签，即同时存在 noisy inputs 和 noisy labels（如下图 1 所示）**，且这种双向噪声在现实世界的图数据中是常见的 [3]，如点击率预测、商品推荐等场景。

于是，我们提出一个新的挑战：如何处理双边噪声以实现鲁棒的链接预测?

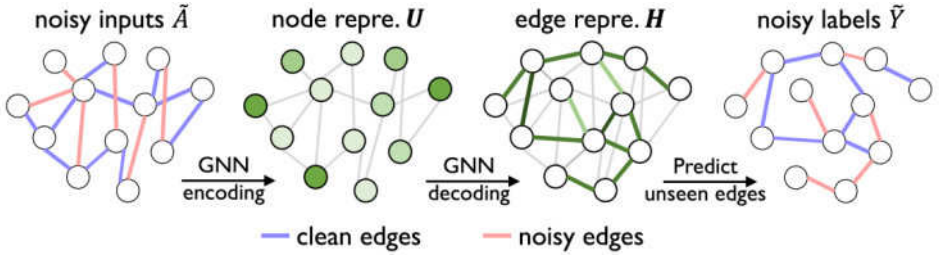


图 1　双边图噪声下的链接预测问题 .

首先，我们进行了一个实证研究，揭示了图结构噪声如何双向干扰输入拓扑结构和目标标签，导致性能严重下降和表征坍缩。为此，我们提出了一个信息论指导原则，即鲁棒图信息瓶颈（Robust Graph Information Bottleneck，RGIB），以提取可靠的监督信号并避免表征坍缩。**与基本的信息瓶颈 GIB[4,5] 不同的是，RGIB 进一步解耦并平衡了图拓扑、图标签和图表征之间的相互依赖性，为抵抗双边噪声的鲁棒表征构**

建了新的学习目标。此外，我们探索了两种实例，RGIB-SSL 和 RGIB-REP，利用自监督学习和数据重参数化方法的优势，分别进行隐式和显式的去噪学习。

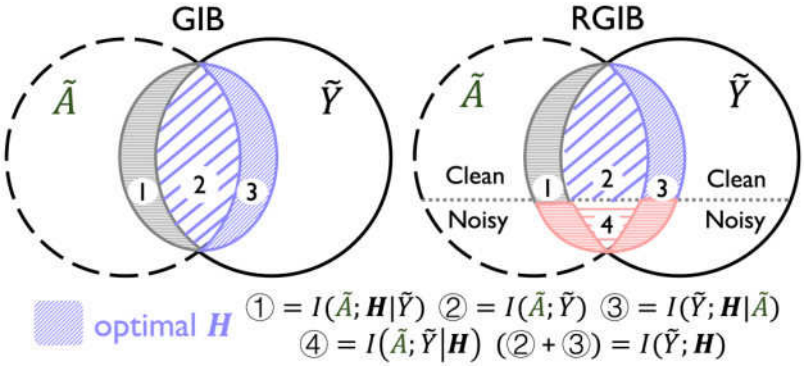


图 2　基本 GIB 和本文提出的 RGIB（其中 A 是图结构，Y 是边标签，H 是图表征，I 为互信息）.

简言之，在本项工作中：

- 我们发现双边噪声会导致严重的表征坍缩和性能下降，并且这种负面影响对常见数据集和图神经网络来说是普遍存在的。据我们所知，我们是最早研究在双边噪声下链接预测鲁棒性问题的。
- 我们提出了一个通用学习框架 RGIB，设计了新的表征学习目标以提高图神经网络的鲁棒性。我们基于不同的方法论提出了两种实现方式，即 RGIB-SSL 和 RGIB-REP，并提出了适应性的设计和理论的分析。
- RGIB 在不修改 GNN 架构的情况下，在 3 种常用 GNN 和 6 个常见数据集上达到了最有效果，各种噪声场景下的 AUC 提升了高达 12.9%，模型学到的表征分布显著恢复，并且对双边噪声更加鲁棒。

接下来，将简要地向大家分享我们近期发表在 NeurIPS 2023 上的有关双边噪声下链接预测鲁棒性的研究结果。

本项研究结果是淘天集团阿里妈妈展示外投团队与香港浸会大学韩波老师研究团队自 2022 年 8 月开始通过阿里巴巴创新研究计划（AIR），共同参与“针对大规模在线广告的可信赖深度学习”项目的研究工作。

论文标题：Combating Bilateral Edge Noise for Robust Link Prediction

论文下载：https://openreview.net/pdf?id=ePkLqJh5kw

代码链接：https://github.com/tmlr-group/RGIB

1. 问题定义

为了定量研究双边图结构噪声的影响，我们在一系列 GNN 基准数据集上合理地模拟不同程度的扰动，详细说明见如下定义 3.1。需要注意的是，目前最常采用的数据划分方式是随机地将部分边分为观测部分和预测目标部分，因此在训练集中，噪声边会被划分到输入和标签中。

双边噪声的生成（定义 3.1）：假设存在一组干净的训练数据，即观察到的图 $\mathcal{G} = (A, X)$，以及查询边的标签 Y。通过向原始邻接矩阵 A 添加边噪声，同时保持节点特征 X 不变，生成了噪声邻接矩阵 $\tilde{A}$。类似地，通过向标签 Y 添加边噪声生成了噪声标签 $\tilde{Y}$。具体而言，给定噪声比例 ε_a，噪声边 A'（$\tilde{A} = A + A'$）通过将 A 中的零元素以概率 ε_a 翻转为一来生成。满足 $A' \odot A = O$ 和 $\varepsilon_a = \frac{|\text{nonzero}(\tilde{A})| - |\text{nonzero}(A)|}{|\text{nonzero}(A)|}$。类似地，可生成噪声标签并添加到原始标签中，其中 $\varepsilon_y = \frac{|\text{nonzero}(\tilde{Y})| - |\text{nonzero}(Y)|}{|\text{nonzero}(Y)|}$。

基于此定义，我们进行实验并发现，双边图结构噪声导致 GNN 的性能显著下降（见图 4），而更大的噪声比率 ϵ 通常导致更严重的性能退化。这意味着，经过标准训练的 GNN 容易受到双边图结构噪声的影响，表现出严重的鲁棒性问题。此外，双边噪声带来的性能下降远远大于单边输入噪声或标签噪声的影响。

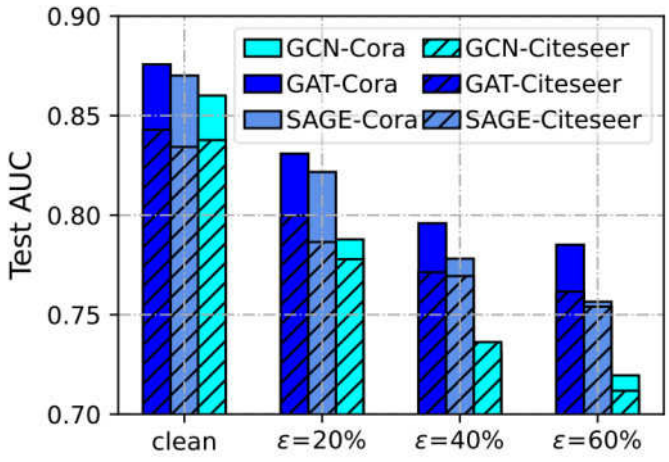


图 3　双边噪声导致显著的性能下降 .

接着，我们检查 GNN 学习得到的表征。从图 5 的 uniformity 分布可以看出，表征在双边噪声的作用下严重坍缩，由原本较为均匀的环状分布逐步退化成了几个单点，且更高的噪声率会导致更严重的坍缩程度，这反映了噪声对于图学习的负面影响，也是

最终性能下降的重要原因。

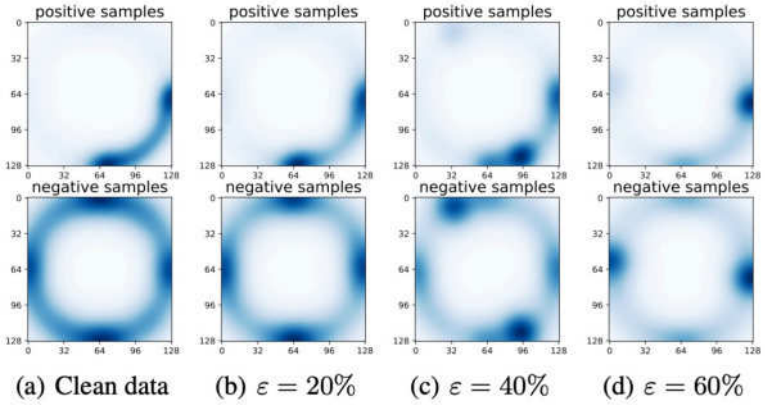


图 4　双边噪声造成严重的表征坍缩 .

2. 解决方案

2.1 GIB 的固有缺陷

为了增强图表征的鲁棒性并避免严重的表征坍缩，我们可以利用图信息瓶颈（Graph Information Bottleneck，GIB）[4,5] 的信息约束作为图表征优化 H 的目标，即：

$$\min \mathrm{GIB} \triangleq -I(H;\tilde{Y}),\ \text{s.t. } I(H;\tilde{A}) < \gamma.$$

其中，超参数 γ 用于限制互信息项，以避免表征 H 过多捕获来自 $\tilde{A}$ 的与任务无关的信息。基本的 GIB 可以有效地防御输入扰动，然而，它在本质上容易受到标签噪声 $\tilde{Y}$ 的影响，因为它完全地保留了标签噪声的监督，所以基本的 GIB 不能够解决双边噪声问题。

2.2 RGIB 优化目标设计

在本工作中，我们尝试对 GIB 进行分析和改进。注意到，基本的 GIB 通过直接约束 $I(H;\tilde{A})$ 来降低 $I(H;\tilde{A}|\tilde{Y})$，以处理输入噪声。同样地，标签噪声可以隐藏在 $I(H;\tilde{Y}|\tilde{A})$ 中，但是简单地约束 $I(H;\tilde{Y})$ 来正则化 $I(H;\tilde{Y}|\tilde{A})$ 并不理想，因为它与 GIB 原始方程冲突，并且也无法处理 $I(\tilde{A};\tilde{Y})$ 内的噪声。因此，进一步解耦 $\tilde{A}$、$\tilde{Y}$ 和 H 之间的依赖关系至关重要。

注意到，噪声可以存在于 $I(H;\tilde{Y}|\tilde{A})$、$I(H;\tilde{A}|\tilde{Y})$ 和 $I(\tilde{A};\tilde{Y}|H)$ 这几个区域。分析

上，我们知道：

$$I(\tilde{A};\tilde{Y}|H) = I(\tilde{A};\tilde{Y}) + I(H;\tilde{Y}|\tilde{A}) + I(H;\tilde{A}|\tilde{Y}) - H(H) + H(H|A,\tilde{Y}).$$

其中 $I(\tilde{A};\tilde{Y})$ 是一个常数，冗余 $H(H|A,\tilde{Y})$ 可以被最小化。因此，$I(\tilde{A};\tilde{Y}|H)$ 可以近似拆解为 $H(H)$，$I(H;\tilde{Y}|\tilde{A})$ 和 $I(H;\tilde{A}|\tilde{Y})$，这三个信息项的平衡可以构成双边图结构噪声问题的解决方案。

基于上述分析，我们提出了 RGIB（Robust Graph Information Bottleneck），一个新的表征 H 学习目标来平衡 $\tilde{A}$、$\tilde{Y}$ 两方面的监督信息，即：

$$\min \text{RGIB} \triangleq -I(H;\tilde{Y}),\ \text{s.t.}\ \gamma_H^- < H(H) < \gamma_H^+, I(H;\tilde{Y}|\tilde{A}) < \gamma_Y,\ I(H;\tilde{A}|\tilde{Y}) < \gamma_A.$$

其中对 $H(H)$ 的约束鼓励更有信息量的表征以防止坍缩（$> \gamma_H^-$），并限制其容量（$< \gamma_H^+$）以避免过拟合。另外两个互信息项 $I(H;\tilde{Y}|\tilde{A})$ 和 $I(H;\tilde{A}|\tilde{Y})$，相互约束后验信息以减轻双边噪声对的负面影响。

需要注意的是，互信息项如 $I(H;\tilde{A}|\tilde{Y})$ 通常是难以精确计算的。因此，我们基于不同的方法论，来给出两种实际的 RGIB 实现，即 RGIB-SSL 和 RGIB-REP。其中，RGIB-SSL 通过自监督正则化显式地优化表征，而 RGIB-REP 通过重参数化隐式地优化表征，详细设计如下。

2.3 RGIB 实例化

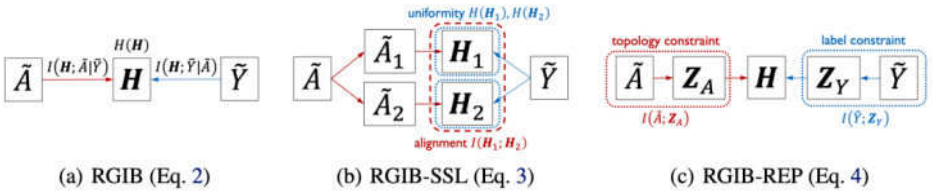


图 5　RGIB 及其实例 RGIB-SSL、RGIB-REP 的示意 .

RGIB-SSL：图表征在监督学习范式下已经退化，自然地，我们将其修改为自监督学习的范式，通过 uniformity 项鼓励表征 H 提高信息量来缓解坍缩，并配合 alignment 项隐式地捕捉含噪变量之间的可靠关系（见图 6b），即：

$$\min \text{RGIB-SSL} \triangleq \underbrace{-\lambda_s(I(H_1;\tilde{Y})+I(H_2;\tilde{Y}))}_{\text{supervision}} \underbrace{-\lambda_u(H(H_1)+H(H_2))}_{\text{uniformity}} \underbrace{-\lambda_a I(H_1;H_2)}_{\text{alignment}}.$$

其中 $\lambda_s, \lambda_u, \lambda_a$ 用于平衡一个监督和两个自监督正则化项，当 $\lambda_s \equiv 1, \lambda_u \equiv 0$ 时，RGIB-SSL 可退化为基本的 GIB。H_1 和 H_2 是两个增强图 $\tilde{A}_1$ 和 $\tilde{A}_2$ 的表征。

RGIB-REP：另一种实现方式是，通过重新参数化拓扑空间和标签空间的信息，保留干净的信息并丢弃噪声部分。为此，我们通过构建隐变量 Z，显式地建模 $\tilde{A}$ 和 $\tilde{Y}$ 的可靠性，以学习一个抗噪声的 H（见图 6c），即：

$$\min \text{RGIB-REP} \triangleq -\underbrace{\lambda_s I(H;Z_Y)}_{\text{supervision}} + \underbrace{\lambda_A I(Z_A;\tilde{A})}_{\text{topology constraint}} + \underbrace{\lambda_Y I(Z_Y;\tilde{Y})}_{\text{label constraint}}.$$

其中，隐变量 Z_Y 和 Z_A 是从含噪的 $\tilde{Y}$ 和 $\tilde{A}$ 中提取的干净信号。它们的补充部分 $Z_{Y'}$ 和 $Z_{A'}$ 被视为噪声，满足 $\tilde{Y} = Z_Y + Z_{Y'}$ 和 $\tilde{A} = Z_A + Z_{A'}$。当 $Z_Y \equiv \tilde{Y}$ 和 $Z_A \equiv \tilde{A}$ 时，RGIB-REP 可退化为基本的 GIB。此外，$I(H;Z_Y)$ 测量了选择样本 Z_Y 的监督信号，其中分类器以 Z_A 作为输入而不是原始的 $\tilde{A}$，即 $H = f_w(Z_A, X)$。

更多技术细节请见正文。

3. 实验结果

我们提供了多维度的实验结果，以验证和理解所提的 RGIB 方法。

3.1 主要性能对比

如表 1 所示，RGIB 在所有 6 个数据集上，在不同噪声比例下，都取得了最佳结果，特别是在 Cora 和 Citeseer 数据集上，与次佳方法相比，RGIB 带来的 AUC 提升达 12.9%。

method	Cora			Citeseer			Pubmed			Facebook			Chameleon			Squirrel		
	20%	40%	60%	20%	40%	60%	20%	40%	60%	20%	40%	60%	20%	40%	60%	20%	40%	60%
Standard	.8111	.7419	.6970	.7864	.7380	.7085	.8870	.8748	.8641	.9829	.9520	.9438	.9616	.9496	.9274	.9432	.9406	.9386
DropEdge	.8017	.7423	.7303	.7635	.7393	.7094	.8711	.8482	.8354	.9811	.9682	.9473	.9568	.9548	.9407	.9439	.9377	.9365
NeuralSparse	.8190	.7318	.7293	.7765	.7397	.7148	.8908	.8733	.8630	.9825	.9638	.9456	.9599	.9497	.9402	.9494	.9309	.9297
PTDNet	.8047	.7559	.7388	.7795	.7423	.7283	.8872	.8733	.8623	.9725	.9674	.9485	.9607	.9514	.9424	.9485	.9326	.9304
Co-teaching	.8197	.7479	.7030	.7533	.7238	.7131	.8943	.8760	.8638	.9820	.9526	.9480	.9595	.9516	.9483	.9461	.9352	.9374
Peer loss	.8185	.7468	.7018	.7423	.7345	.7104	.8961	.8815	.8566	.9807	.9536	.9430	.9543	.9533	.9267	.9457	.9345	.9286
Jaccard	.8143	.7498	.7024	.7473	.7324	.7107	.8872	.8803	.8512	.9794	.9579	.9428	.9503	.9538	.9344	.9443	.9327	.9244
GIB	.8198	.7485	.7148	.7509	.7388	.7121	.8899	.8729	.8544	.9773	.9608	.9417	.9554	.9561	.9321	.9472	.9329	.9302
VIB	.8208	.7810	.7218	.7701	.8120	.7185	.8927	.8825	.8501	.9697	.9637	.9500	.9529	.9561	.9487	.9431	.9399	.9288
PRI	.7976	.7330	.6981	.7567	.7452	.7018	.8898	.8801	.8487	.9601	.9619	.9507	.9513	.9499	.9490	.9382	.9413	.9301
SupCon	.8240	.7819	.7490	.7554	.7458	.7299	.8853	.8718	.8525	.9588	.9508	.9297	.9561	.9531	.9467	.9473	.9348	.9301
GRACE	.7872	.6940	.6929	.7632	.7242	.6844	.8922	.8749	.8588	.8899	.8865	.8315	.8978	.8987	.8949	.9394	.9380	.9363
RGIB-REP	.8313	.7966	.7591	.7875	.7519	.7312	.9017	.8834	.8652	**.9832**	**.9770**	.9519	**.9723**	**.9621**	**.9519**	**.9509**	**.9455**	**.9434**
RGIB-SSL	**.8930**	**.8554**	**.8339**	**.8694**	**.8427**	**.8137**	**.9225**	**.8918**	**.8697**	.9829	.9711	**.9643**	.9655	.9592	.9500	.9499	.9426	.9425

表 1　双边噪声下实验结果展示 .

表 2 中展示了单边噪声的实验结果。无论是针对单边输入噪声还是标签噪声，RGIB

仍然超越了所有的基准方法。实验表明，双边图结构噪声可以通过统一的学习框架来建模和解决，而此前的去噪方法只能用于特定的噪声模式。

input noise	Cora			Citeseer			Pubmed			Facebook			Chameleon			Squirrel		
	20%	40%	60%	20%	40%	60%	20%	40%	60%	20%	40%	60%	20%	40%	60%	20%	40%	60%
Standard	.8027	.7856	.7490	.8054	.7708	.7583	.8854	.8759	.8651	.9819	.9668	.9622	.9608	.9433	.9368	.9416	.9395	.9411
DropEdge	.8338	.7826	.7454	.8025	.7730	.7473	.8682	.8456	.8376	.9803	.9685	.9531	.9567	.9433	.9432	.9426	.9376	.9358
NeuralSparse	.8534	.7794	.7637	.8093	.7809	.7468	.8931	.8720	.8649	.9712	.9691	.9583	.9609	.9540	.9348	.9469	.9403	.9417
PTDNet	.8433	.8214	.7770	.8119	.7811	.7638	.8903	.8776	.8609	.9725	.9668	.9493	.9610	.9457	.9360	.9469	.9400	.9379
Jaccard	.8200	.7838	.7617	.8176	.7776	.7725	.8987	.8764	.8639	.9784	.9702	.9638	.9507	.9436	.9364	.9388	.9345	.9240
GIB	.8002	.8099	.7741	.8070	.7717	.7798	.8932	.8808	.8618	.9796	.9647	.9650	.9605	.9521	.9416	.9390	.9406	.9397
VIB	.8603	**.8590**	.8008	.8497	.8399	.7703	.8910	.8829	.8519	.9800	.9710	.9536	.9499	.9558	.9312	.9416	.9402	.9297
PRI	.8307	.7990	.7736	.8105	.7852	.7558	.8801	.8790	.8551	.9691	.9698	.9529	.9528	.9501	.9435	.9397	.9419	.9318
SupCon	.8349	.8301	.8025	.8076	.7767	.7655	.8867	.8739	.8558	.9647	.9517	.9401	.9606	.9536	.9468	.9372	.9343	.9305
GRACE	.7877	.7107	.6975	.7615	.7151	.6830	.8810	.8795	.8593	.9015	.8833	.8395	.8994	.9007	.8964	.9392	.9378	.9363
RGIB-REP	.8624	.8313	.8158	.8299	.7996	.7771	.9008	.8822	.8687	**.9833**	**.9723**	**.9682**	**.9705**	**.9604**	.9480	**.9495**	**.9432**	.9405
RGIB-SSL	**.9024**	.8577	**.8421**	**.8747**	**.8461**	**.8245**	**.9126**	**.8889**	**.8693**	.9821	.9707	.9668	.9658	.9570	**.9486**	.9479	.9429	**.9429**
label noise	Cora			Citeseer			Pubmed			Facebook			Chameleon			Squirrel		
	20%	40%	60%	20%	40%	60%	20%	40%	60%	20%	40%	60%	20%	40%	60%	20%	40%	60%
Standard	.8281	.8054	.8060	.7965	.7850	.7659	.9030	.9039	.9070	.9882	.9880	.9886	.9686	.9580	.9362	.9720	.9720	.9710
Co-teaching	.8446	.8209	.8157	.7974	.7877	.7913	.9315	.9291	.9319	.9762	.9797	.9638	.9642	.9650	.9533	.9675	.9641	.9655
Peer loss	.8325	.8036	.8069	.7991	.7990	.7751	.9126	.9101	.9210	.9769	.9750	.9734	.9621	.9501	.9569	.9636	.9694	.9696
Jaccard	.8289	.8064	.8148	.8061	.7887	.7689	.9098	.9135	.9096	.9702	.9725	.9758	.9603	.9659	.9557	.9529	.9512	.9501
GIB	.8337	.8137	.8157	.7986	.7852	.7649	.9037	.9114	.9064	.9742	.9703	.9771	.9651	.9582	.9489	.9641	.9628	.9601
VIB	.8406	.8296	.8036	.8068	.8034	.7739	.9088	.9042	.8981	.9705	.9790	.9713	.9610	.9629	.9518	.9568	.9722	.9690
PRI	.8298	.8139	.7960	.7902	.7881	.7725	.8939	.8917	.8832	.9733	.9802	.9620	.9414	.9593	.9599	.9658	.9698	.9533
SupCon	.8491	.8275	.8256	.8024	.7983	.7807	.9131	.9108	.9162	.9647	.9567	.9553	.9584	.9580	.9477	.9516	.9595	.9511
GRACE	.8531	.8237	.8193	.7909	.7630	.7737	.9234	.9252	.9255	.8913	.8972	.8887	.9053	.9074	.9075	.9171	.9174	.9166
RGIB-REP	.8554	.8318	.8297	.8083	.7846	.7945	.9357	.9343	.9332	**.9884**	**.9883**	**.9889**	**.9785**	**.9797**	**.9785**	**.9735**	**.9733**	**.9737**
RGIB-SSL	**.9314**	**.9224**	**.9241**	**.9204**	**.9218**	**.9250**	**.9594**	**.9604**	**.9613**	.9857	.9881	.9857	.9730	.9752	.9744	.9727	.9729	.9726

表 2　单边噪声下实验结果展示 .

3.2 多方面的消融实验及深入讨论

我们进一步进行了诸多消融实验，深入探讨了所提方法在不同角度下的表现。

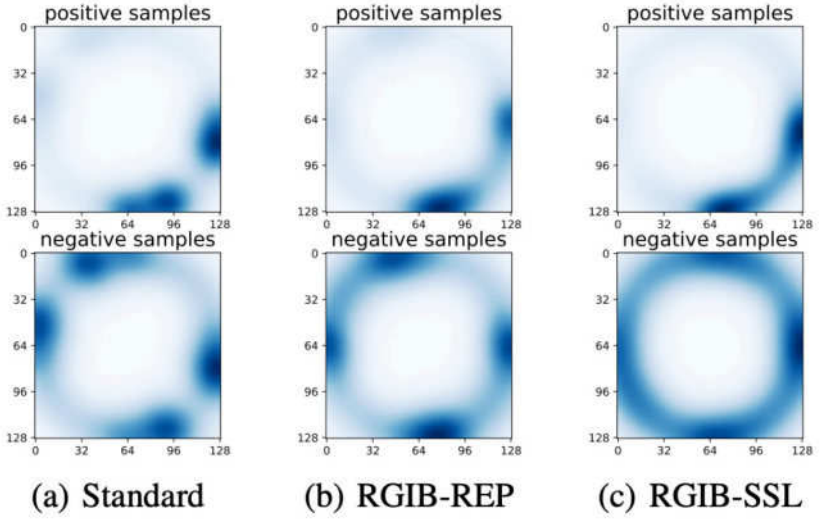


图 6　RGIB 能显著改善表征分布，降低坍缩程度 .

dataset	Cora		Citeseer		Pubmed	
method	SSL	REP	SSL	REP	SSL	REP
$constant$	.8398	**.7927**	**.8227**	**.7742**	.8596	**.8416**
$linear(\cdot)$	.8427	.7653	.8167	.7559	**.8645**	.8239
$sin(\cdot)$	**.8436**	.7924	.8132	.7680	.8637	.8275
$cos(\cdot)$	.8334	.7833	.8088	.7647	.8579	.8372
$exp(\cdot)$	.8381	.7815	.8085	.7569	.8617	.8177

表 3　RGIB 在不同超参 schedule 下的表现 .

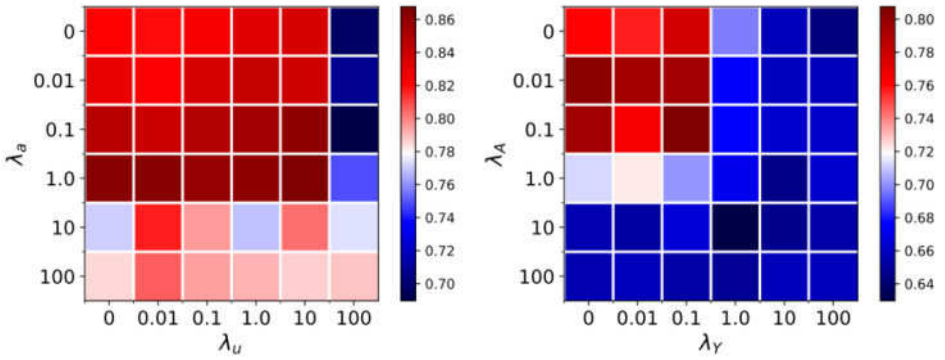


图 7　RGIB 的超参数搜索结果热力图 .

adversarial attacks	Cora clean	Cora $\epsilon_{adv}=20\%$	Cora $\epsilon_{adv}=40\%$	Cora $\epsilon_{adv}=60\%$	Citeseer clean	Citeseer $\epsilon_{adv}=20\%$	Citeseer $\epsilon_{adv}=40\%$	Citeseer $\epsilon_{adv}=60\%$
Standard	.8686	.7971	.7671	.7014	.8317	.8139	.7736	.7481
RGIB-SSL	**.9260**	.8296	**.8095**	**.8052**	**.9148**	**.8656**	**.8347**	**.8022**
RGIB-REP	.8758	**.8408**	.7918	.7611	.8415	.8382	.8107	.7893

表 4　RGIB 在对抗扰动下的实验结果 .

variant	Cora $\epsilon=60\%$	Cora $\epsilon_a=60\%$	Cora $\epsilon_y=60\%$	Chameleon $\epsilon=60\%$	Chameleon $\epsilon_a=60\%$	Chameleon $\epsilon_y=60\%$
RGIB-SSL (full)	.8596	.8730	.8994	.9663	.9758	.9762
- w/o hybrid augmentation	.8150 (5.1%↓)	.8604 (1.4%↓)	.8757 (2.6%↓)	.9528 (1.3%↓)	.9746 (0.1%↓)	.9695 (0.6%↓)
- w/o self-adversarial	.8410 (2.1%↓)	.8705 (0.2%↓)	.8927 (0.7%↓)	.9655 (0.1%↓)	.9732 (0.2%↓)	.9746 (0.1%↓)
- w/o supervision ($\lambda_s=0$)	.7480 (12.9%↓)	.7810 (10.5%↓)	.7820 (13.0%↓)	.8626 (10.7%↓)	.8628 (11.5%↓)	.8512 (12.8%↓)
- w/o alignment ($\lambda_a=0$)	.8194 (4.6%↓)	.8510 (2.5%↓)	.8461 (5.9%↓)	.9613 (0.5%↓)	.9749 (0.1%↓)	.9722 (0.4%↓)
- w/o uniformity ($\lambda_u=0$)	.8355 (2.8%↓)	.8621 (1.2%↓)	.8878 (1.3%↓)	.9652 (0.1%↓)	.9710 (0.4%↓)	.9751 (0.1%↓)
RGIB-REP (full)	.7611	.8487	.8095	.9567	.9706	.9676
- w/o edge selection ($Z_A \equiv \tilde{A}$)	.7515 (1.2%↓)	.8199 (3.3%↓)	.7890 (2.5%↓)	.9554(0.1%↓)	.9704 (0.1%↓)	.9661 (0.1%↓)
- w/o label selection ($Z_Y \equiv \tilde{Y}$)	.7533 (1.0%↓)	.8373 (1.3%↓)	.7847 (3.0%↓)	.9484(0.8%↓)	.9666 (0.4%↓)	.9594 (0.8%↓)
- w/o topology constraint ($\lambda_A=0$)	.7355 (3.3%↓)	.7699 (9.2%↓)	.7969 (1.5%↓)	.9503(0.6%↓)	.9658 (0.4%↓)	.9635 (0.4%↓)
- w/o label constraint ($\lambda_Y=0$)	.7381 (3.0%↓)	.8106 (4.4%↓)	.8032 (0.7%↓)	.9443(1.2%↓)	.9665 (0.4%↓)	.9669 (0.1%↓)

表 5　RGIB 的消融实验 .

除此以外，我们提供了更多的可视化及相关实验结果，感兴趣的读者请移步原文与附录部分。

4. 算法落地

本文提出的 RGIB-SSL 方法，在展示外投业务中进行了算法落地。在该业务中，商家广告被投放于全域互联网媒体流量上。本技术通过在预训练上对用户广告行为特征构图并约束 RGIB，增强了对点击行为的预估鲁棒性，从而提升精排阶段点击率预估的准确性，提升投放广告的精准度与质量与在媒体流量出价上的准确度，使得大盘营收获得约 5% 的提升。该技术全面应用于展示外投的几乎所有媒体流量，覆盖数十家媒体、近百个流量资源位和数亿用户。

5. 总结及展望

本文研究了带有双边图结构噪声的链接预测问题，并发现在这种双边噪声下，GNN 学习得到的表征严重坍缩。基于这一观察，我们引入了鲁棒图信息瓶颈原则 RGIB，旨在通过解耦和平衡输入、标签和表征之间的互信息来提取可靠信号，以增强表征鲁棒性并避免坍缩。展望未来，可将 RGIB 拓展至节点预测（Node Classification）、整图预测（Graph Classification）即知识图谱推理（Knowledge Graph Reasoning）等任务上。此外，正交于本文研究的结构噪声（Structural Noise），图节点特征上的噪声（Feature Noise）同样值得关注。

参考文献

[1] D. Liben-Nowell and J. Kleinberg. The link-prediction problem for social networks. Journal of the American society for information science and technology, 2007.

[2] M. Zhang and Y. Chen. Link prediction based on graph neural networks. In NeurIPS, 2018.

[3] B. Wu, J. Li, C. Hou, G. Fu, Y. Bian, L. Chen, and J. Huang. Recent advances in reliable deep graph learning: Adversarial attack, inherent noise, and distribution shift. arXiv, 2022.

[4] T. Wu, H. Ren, P. Li, and J. Leskovec. Graph information bottleneck. In NeurIPS, 2020.

[5] J. Yu, T. Xu, Y. Rong, Y. Bian, J. Huang, and R. He. Graph information bottleneck for subgraph recognition. arXiv, 2020

团队介绍

阿里妈妈展示外投团队

阿里妈妈展示外投团队是阿里妈妈核心广告技术团队之一，也是阿里妈妈业务增长最快的团队。依托于集团庞大而真实的营销场景，以 AI 技术驱动实现客户商品营销，并承担集团 App 用户增长等业务需求。我们持续探索人工智能，联邦学习，深度学习，强化学习，知识图谱，图学习等前沿技术在外投广告和用增方面的落地应用。在创造业务价值的同时，团队近几年在 ICML、NIPS、WWW、CIKM、SIGIR、KDD、NAACL 等领域知名会议上发表过多篇论文。真诚欢迎对广告算法、推荐系统、NLP 等方向感兴趣的同学加入我们，一起成长！

简历投递邮箱：alimama_tech@service.alibaba.com

香港浸会大学可信机器学习和推理组

香港浸会大学可信机器学习和推理课题组 (TMLR Group) 由多名青年教授、博士后研究员、博士生、访问博士生和研究助理共同组成，课题组隶属于理学院计算机系。课题组专攻可信表征学习、基于因果推理的可信学习、可信基础模型等相关的算法，理论和系统设计以及在自然科学上的应用，具体研究方向和相关成果详见本组 Github (https://github.com/tmlr-group)。课题组由政府科研基金以及工业界科研基金资助，如香港研究资助局杰出青年学者计划，国家自然科学基金面上项目和青年项目，以及国内外企业的科研基金。青年教授和资深研究员手把手带，GPU 计算资源充足，长期招收多名博士后研究员、博士生、研究助理和研究实习生。感兴趣的同学请发送个人简历和初步研究计划到邮箱：bhanml@comp.hkbu.edu.hk。

Memorization Discrepancy：利用模型动态信息发现累积性注毒攻击

作者：零刻

本文分享阿里妈妈外投算法团队与香港浸会大学可信赖机器学习和推理组（HKBU TMLR Group）合作在理论与实践上探索外投广告媒体等复杂场景下应对噪声信号进行模型训练的问题。基于该项工作总结的论文已发表在 ICML 2023，欢迎阅读交流。

摘要

近期研究表明，对抗性注毒攻击（Poisoning attack）对各类机器学习应用会构成巨大威胁 [1,2]。有别于之前研究所关注的线下注毒设定，累积性注毒攻击（accumulative poisoning attack）[3] 是最近提出的一种模拟线上实时数据流设定下进行注毒攻击的方法。通过利用两阶段不同的注毒样本生成投放，累积性注毒攻击能够优化第一阶段中注毒样本的不可辨别性，以此实现更为隐匿的注毒攻击。并且，能够将第一阶段中的注毒效果累积至第二个触发阶段，达到在短时间内大幅降低模型性能的效果。

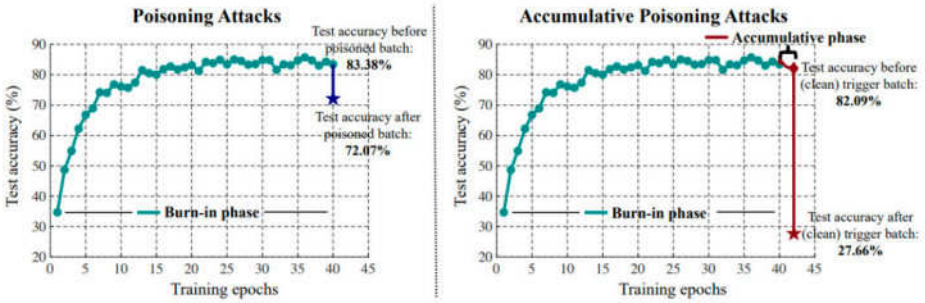


图 1　累积注毒攻击 [3] 的效果示意

考虑到线上模拟实时数据流的学习设定，之前基于线下注毒攻击的防御算法或数据层面的检测方法由于信息的缺失无法应对这种新型的注毒攻击方法。然而，现有可能的防御方法例如对抗训练变体 [4] 及梯度截断 [5] 会导致对注毒数据与干净数据有无差别或过度纠正，进而对机器学习模型的性能有所影响。由于对在线数据流的数据检测相对困难，我们考虑到是否可以从模型的动态变化中挖掘对发现注毒攻击有用的信息来识别这种隐匿的累积注毒攻击。基于此，我们在本项工作中：

- 首次从模型动态变化的角度探索注毒攻击的发现；
- 提出了一种新的信息评价指标，记忆性差异，利用模型动态变化尝试分辨不可见的注毒样本；
- 基于新的信息评价指标提出了针对累积注毒攻击的防御学习算法。

接下来将简要地向大家分享我们近期发表在 ICML 2023 上的有关模型动态及注毒样本发现的研究结果，欢迎阅读交流。

论 文：Exploring Model Dynamics for Accumulative Poisoning Discovery

论 文 作 者：Jianing Zhu, Xiawei Guo, Jiangchao Yao, Chao Du, Li He, Shuo Yuan, Tongliang Liu, Liang Wang, Bo Han

下 载：https://arxiv.org/abs/2306.03726

代码链接：https://github.com/tmlr-group/Memorization-Discrepancy

1. 回溯历史模型

本文沿着累积注毒攻击 [3] 的实验设定，我们考虑在基准数据集上预训练到一定阶段的模型（例如图 1 中的 Burn-in phase），在接下来的训练过程中模拟线上实时数据流的接入与训练，监测模型在基础分类任务上的性能变化。

由于累积注毒样本经过不可分辨的目标优化，在数据层面上的差异通过注毒样本生成的方式已经被缩小。并且，考虑到实时数据流的真实场景，在没有额外的先验知识情况下，我们很难得知从端口新获取的数据样本是为干净样本还是注毒样本。

1.1 从数据层面信息转向模型动态的视角

通过上述分析，对于新获取的数据样本组，通过对比其与同样的干净样本（即生成注毒样本之前的同一批）的单一维度输出信息难以对两种样本进行分辨，因为累积注毒样本的第一阶段目标存在隐藏注毒目标的部分。白盒的攻击设定使得注毒者能够充分掌握当前的模型参数信息进行注毒样本的生成。然而，受攻击模型有一个重要的部分，即历史模型信息，可以为注毒样本的鉴别提供潜在的多维度信息。

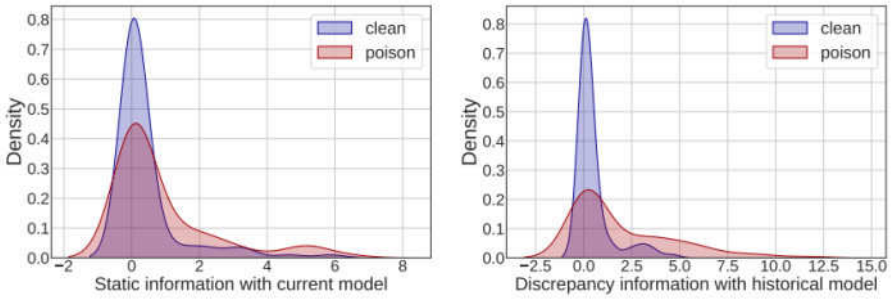


图 2　对比当前状态及考虑不同时刻模型输出信息下的干净样本和注毒样本指标差异

在实验中发现，通过构造模型输出的差异化信息，继而放大回溯的模型间隔，干净样本和注毒样本的差异可以被显著放大。在这里我们猜想，可以通过挖掘模型动态层面的信息发现一些有用的线索用于发现潜在的注毒样本。

2. 新的信息评价指标：记忆性差异

之前的实验结果展示了回溯历史模型来构造差异信息发现注毒样本的潜力，在接下来的部分具体地阐述主要构造方法和新的信息评价指标。

2.1 模型动态信息视角的深入探索

回溯历史模型以构造差异性信息的核心操作十分简单，在获取当前模型以及历史模型对同一批数据的输出信息后可以进行计算。具体做法如下图，在下图底部子图中，通过扩大回溯模型的间隔，干净样本与注毒样本的差异在整体上存在放大的趋势。

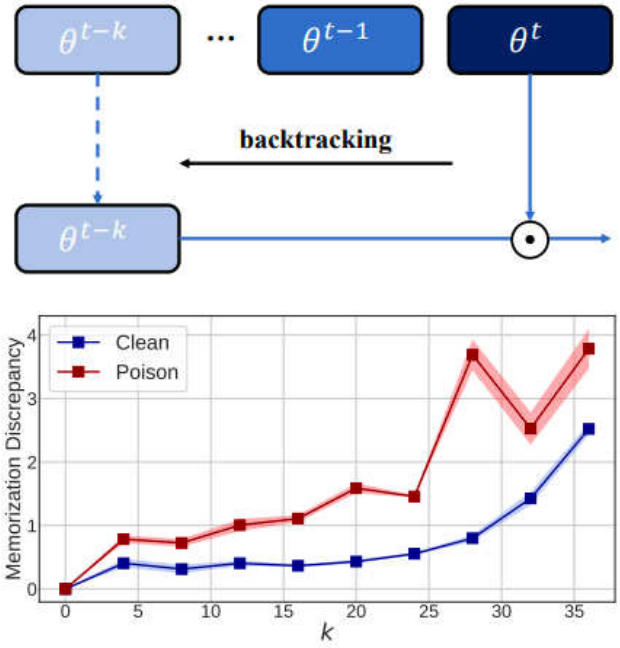


图 3.1　构造残差信息的操作示意及记忆性差异的数值监测

2.2 Memorization Discrepancy

这种构造的差异信息具体定义如下，考虑当前模型 θ^t 及回溯的历史模型 θ^{t-k}，我们定义在当前模型基础上生成的潜在注毒样本 $\hat{x}(\theta^t)$ 的记忆性差异为，

$$\mathbb{D}(f(\hat{x}(\theta^t);\theta^{t-k}), f(\hat{x}(\theta^t);\theta^t))$$

其中，$\mathbb{D}$ 代表通用的信息差异计算方法，在我们的研究工作中，主要使用 KL Divergence 进行计算与实验性探索。

在下图中，我们通过示意图解释模型动态层面差异的底层原理解读。

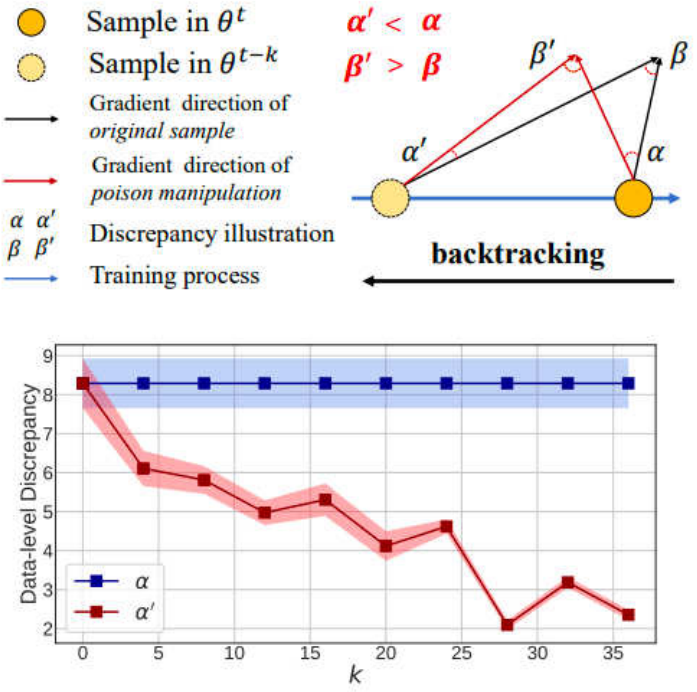
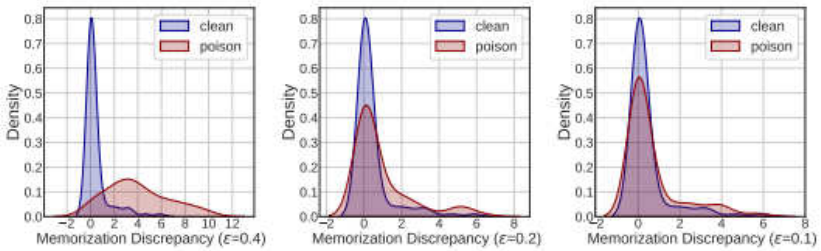


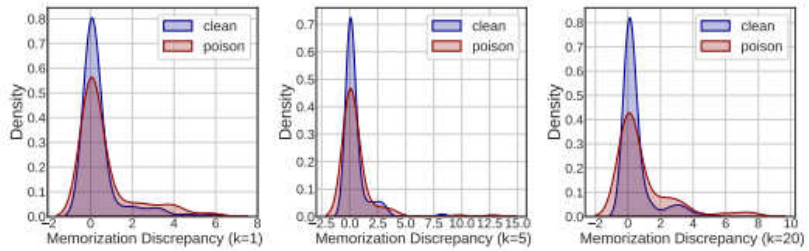
图 3.2　自然学习与注毒攻击目标差异导致的模型动态变化差异

在图 3.2 的顶部子图中，横轴代表模型训练的不同状态，我们对同一批样本在不同 模型状态的输出构造信息差异。整体上，我们可以考虑在干净样本上的标准训练及在注毒样本上进行训练的效果差异，由于注毒样本的目的在于降低模型在同一数据集上的性能表现，模型在注毒数据上进行训练优化的方向相较于干净数据会有所变化。而这种变化会由于模型训练的状态发生差异，这种差异可以通过回溯模型历史状态进行放大。

由此，我们分析了新提出的记忆性差异相关性质，更多的细节可以参考论文相应的部分。在下图中，我们展示了记忆性差异对两种样本的分辨能力随着注毒程度及回溯间隔的相关表现。



(a) Under different poisoning capacities (k=1)

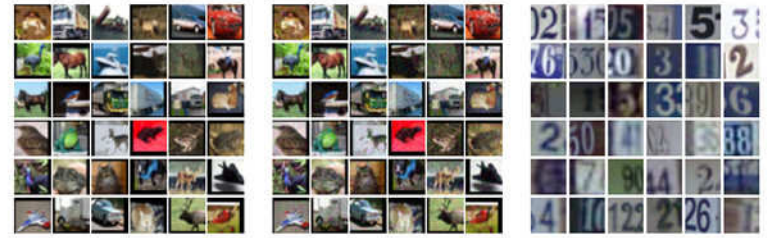


(b) Under different backtracking intervals (ϵ=0.1)

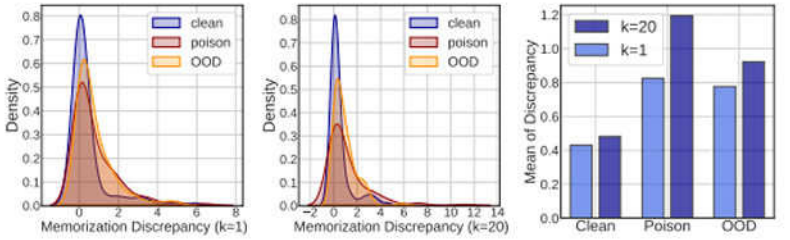
图 4　有关注毒样本攻击程度及模型回溯间隔在记忆性差异上的动态变化

3. 利用模型动态所反映的数据信息

除此之外，我们也考虑了实时数据流可能出现的分布外数据，将干净样本，注毒样本及语义层面差异更大的分布外样本数据在同一训练框架下进行比较，分别计算记忆性差异指标。我们发现，相对于注毒样本，分布外样本与干净样本的差异并不如注毒样本那么大，再次体现了这种差异化信息对此类对模型性能直接影响的注毒样本具有特别的识别潜力。



(a) Visualization of clean, poison, and out-of-distribution data



(b) Comparisons of the above three type of data

图 5　记忆性差异在分辨不同样本的分布比较

3.1 Discrepancy-aware Sample Correction (DSC)

基于前文定义的记忆性差异，我们提出一种差异化样本纠正方法对实时数据流中新获取的数据进行有分辨的学习并防御潜在的注毒攻击。具体的学习优化目标如下，

$$\tilde{x} = \arg\min_{\tilde{x}\in\mathcal{B}[x,\epsilon]} \ell(f(x), y), \quad \text{s.t.}, \mathbb{D}(f(\tilde{x};\theta), f(\tilde{x};\theta^*)) > P,$$

其中 $\tilde{x}$ 是经过纠正的样本，$\mathcal{B}[x,\epsilon] = \{\tilde{x} \mid d_\infty(x,\tilde{x}) \leq \epsilon\}$ 限制了以训练样本为中心 $\epsilon > 0$ 的一个纠正范围，P 是估计得到的纠正阈值，考虑到前文中探索得到的干净样本及注毒样本在记忆性差异上的区别，差异化样本纠正在反向生成对抗样本过程中通过计算记忆性差异将样本纠正至一个安全的范围中，在减弱注毒攻击影响的同时避免了对干净样本的过度优化。

3.2 整体算法流程

Algorithm 1 Discrepancy-aware Sample Correction (DSC)

Input: data streaming $S = \{(x_i, y_i)\}_{i=1}^{n}$, learning rate η, number of epochs T, batch size m, number of batches M, data $x \in \mathcal{X}$, label $y \in \mathcal{Y}$, victim model θ, loss function ℓ, PGD step K, perturbation bound ϵ, step size δ, projection opt. Π, Memorization Discrepancy threshold P, auxiliary historical model θ^*.

Output: model θ^T;

for epoch $= 1, \ldots, T$ **do**
  for mini-batch $= 1, \ldots, M$ **do**
    Sample a mini-batch $\{(x_i, y_i)\}_{i=1}^{m}$ from S
    for $i = 1, \ldots, m$ (in parallel) **do**
      Obtain the corrected sample $\tilde{x}_i$ of x_i:
      $\tilde{x}_i \leftarrow x_i, n = 1$
      while $\mathbb{D}(f(\tilde{x}_i;\theta), f(\tilde{x}_i;\theta^*)) > P$ and $n < K$ **do**
        $\tilde{x}_i \leftarrow \Pi_{\mathcal{B}[x_i,\epsilon]}(\tilde{x}_i - \delta \cdot \text{sign}(\nabla_{\tilde{x}_i}\ell(f(\tilde{x}_i), y)))$
        $n = n + 1$
      end while
    end for
    $\theta \leftarrow \theta - \eta\nabla_\theta \ell(f_\theta(\tilde{x}_i), y_i)$
  end for
end for

算法 1　Discrepancy-aware Sample Correction

整体的算法流程如上所示，带有差异化指标的样本纠正建立在经典的反向生成对抗样本算法基础上。对于每一批从数据流中新抓取的数据，本算法会先用另一历史辅助模型与当前模型对该样本的输出进行记忆性差异指标的计算，同时判断是否记忆性差异指标落入到安全范围之内，并无中毒风险，继而正常训练，反之则持续做反向生成对抗样本的扰动修正，使得可疑样本被修正到可学习的范围空间内。

3.3 方法有效性初探

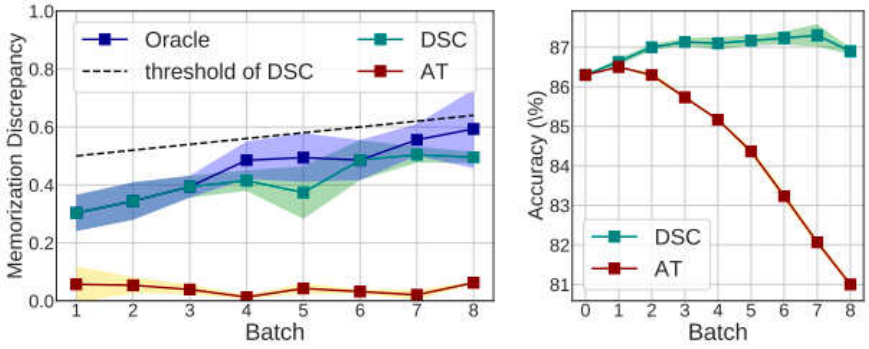


图 6　DSC 对于改善无差别纠正的效果

在上图中，我们比较了不同防御学习算法对记忆性差异的影响及模型性能表现的影响，可以初步发现实验效果验证了我们提出的差异化纠正的算法期望。

4. 实验与讨论

在本项工作中，我们提供了多维度的实验结果以理解我们提出的方法表现。

4.1 主要性能对比

如表 1 所示，主要性能对比在三个基准分类数据集上进行。分别对比了标准训练，利用梯度截断，反向对抗训练，及我们提出的差异化样本纠正的方式在第一个 Burn-in phase 之后进行注毒攻击及完整标准训练两个实验设定上的表现，可以发现，在不同的数据集上我们的方法均取得了令人满意的效果。

CIFAR-10	Defense	Accuracy: Start	Batch	Accuracy: +Poison	Accuracy: + Trigger	Δ
Clean Oracle	ST	86.3	-	84.4	84.4	-
	GC		-	86.2	86.2	-
	AT		-	77.2	77.2	-
	DSC		-	84.7	84.7	-
Accu. Poison	ST		1	75.7±3.33	50.4±5.03	-25.3±4.13
	GC		3	79.7±0.25	75.1±0.05	-4.6±0.26
	AT		3	80.1±0.10	75.3±0.26	-4.7±0.20
	DSC		3	**81.2±0.35**	**77.3±0.58**	**-3.8±0.31**
SVHN	Defense	Accuracy: Start	Batch	Accuracy: +Poison	Accuracy: + Trigger	Δ
Clean Oracle	ST	94.6	-	93.4	93.4	-
	GC		-	94.5	94.5	-
	AT		-	89.9	89.9	-
	DSC		-	94.7	94.7	-
Accu. Poison	ST		3	85.4±3.54	70.4±9.16	-15.4±6.2
	GC		7	89.7±0.06	88.3±0.26	-1.4±0.30
	AT		7	89.6±0.21	88.7±0.20	**-0.9±0.06**
	DSC		9	**89.9±0.01**	**88.8±0.26**	-1.1±0.26
CIFAR-100	Defense	Accuracy: Start	Batch	Accuracy: +Poison	Accuracy: + Trigger	Δ
Clean Oracle	ST	59.0	-	55.8	55.8	-
	GC		-	60.2	60.2	-
	AT		-	49.5	49.5	-
	DSC		-	55.0	55.0	-
Accu. Poison	ST		3	42.9±2.74	32.6±2.84	-10.3±0.29
	GC		4	49.8±0.12	43.8±0.29	-6.1±0.25
	AT		5	47.7±0.25	44.4±0.21	-3.2±0.42
	DSC		5	**48.6±0.91**	**45.4±1.39**	**-3.2±0.65**

表 1　主要实验对比结果展示

4.2 多方面的消融实验及深入讨论

我们进一步进行了诸多消融实验，深入探讨了所提方法在不同角度下的表现。

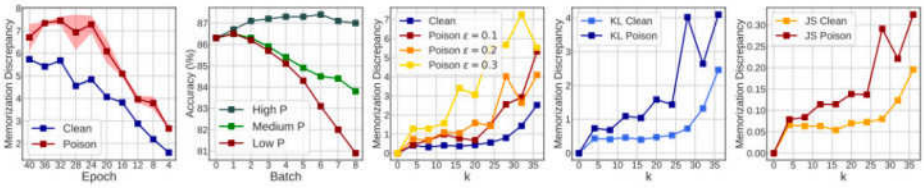


图 7　多样化的消融实验刻画所提出的记忆性差异

Defense/Attack		PGD	KL (TRADES)	CW_∞
DSC	Start	86.3		
	Acc. +Poison	81.4	80.3	81.8
	Acc. +Trigger	78.7	77.3	79.3
	Δ	-2.69	-3.04	-2.54

表 2　不同反对抗纠正目标对 DSC 的影响

Constraint β		0	0.05	0.1	0.2
ST	Start	86.3			
	Acc. +Poison	81.4	80.9	80.9	80.3
	Acc. +Trigger	51.3	59.9	69.8	74.9
	Δ	-30.04	-20.97	-11.12	-5.43

表 3　不同程度约束项对标准训练的影响

Dataset	Model/Interval k	Discrepancy on	4	8	12	16	20	24	28	32	36
CIFAR-10	ResNet	Clean	0.43444	0.40864	0.46287	0.39770	0.47189	0.52683	0.72566	1.31921	2.45922
		Poison	0.73276	0.68203	1.09318	1.03971	1.58336	1.43465	4.01915	2.64194	4.09608
	VGG-11	Clean	0.52409	0.41497	0.51741	0.82906	0.94531	1.32099	1.97665	2.82950	5.69943
		Poison	0.33073	0.44917	0.41988	0.80057	0.89317	1.25389	3.65167	6.73917	14.53571
	SmallCNN	Clean	0.42651	0.67719	0.48234	0.53239	0.46322	0.64273	1.05432	0.65214	1.89398
		Poison	1.04400	1.49517	1.08457	0.94815	1.91275	1.88436	3.73850	3.34128	5.79640

表 4　在不同模型上计算所得记忆性差异的数值比较

除此以外，我们提供了更多的可视化及相关实验结果，感兴趣的读者敬请移步原文与附录部分。

5. 总结及展望

本文思考了从模型动态信息变化的角度挖掘对实时数据流中注毒样本识别的有用信息。通过回溯历史模型，利用当前模型及历史模型信息构造差异化信息指标，我们在实验中发现这种新型的指标有助于捕捉干净样本及注毒样本的差异，进一步提出了针对性的防御学习算法对新获取的数据进行差异化纠正。展望未来，实时数据流中还可能存在分布外样本，数据噪声，随时间变化的动态数据分布漂移等诸多问题，如何分辨并从中学习有价值的信息仍值得我们进一步探索。

参考文献

[1] Li, B., Wang, Y., Singh, A., and Vorobeychik, Y. Data poisoning attacks on

factorization-based collaborative filtering. In NeurIPS, 2016.
[2] Fowl, L. H., Goldblum, M., Chiang, P.-y., Geiping, J., Czaja, W., and Goldstein, T. Adversarial examples make strong poisons. In NeurIPS, 2021.
[3] Pang, T., Yang, X., Dong, Y., Su, H., and Zhu, J. Accumulative poisoning attacks on real-time data. In NeurIPS, 2021.
[4] Tao, L., Feng, L., Yi, J., Huang, S.-J., and Chen, S. Better safe than sorry: Preventing delusive adversaries with adversarial training. In NeurIPS, 2021.
[5] Pascanu, R., Mikolov, T., and Bengio, Y. On the difficulty of training recurrent neural networks. In ICML, 2013.

团队介绍

阿里妈妈效果外投团队

阿里妈妈效果外投团队是阿里妈妈核心广告技术团队之一，也是阿里妈妈业务增长最快的团队。依托于集团庞大而真实的营销场景，以 AI 技术驱动实现客户商品营销，并承担集团 App 用户增长等业务需求。我们持续探索人工智能，联邦学习，深度学习，强化学习，知识图谱，图学习等前沿技术在外投广告和用增方面的落地应用。在创造业务价值的同时，团队近几年在 ICML、NIPS、WWW、CIKM、SIGIR、KDD、NAACL 等领域知名会议上发表过多篇论文。真诚欢迎对广告算法、推荐系统、NLP 等方向感兴趣的同学加入我们，一起成长！

简历投递邮箱：alimama_tech@service.alibaba.com

香港浸会大学可信赖机器学习和推理组

香港浸会大学可信赖机器学习和推理组 (HKBU TMLR Group) 专攻可信赖机器学习和推理，具体研究方向和相关成果详见：https://github.com/tmlr-group；课题组由政府科研基金 RGC Early CAREER Scheme（香港研究资助局早期职业生涯计划），NSFC General Program and Young Scientists Fund（国家自然科学基金面上项目和青年基金）和工业界科研基金资助，资深研究员手把手带，GPU 机器充足，长期招收多名博士后研究员，博士生，研究助理，研究实习生。感兴趣的同学请发送个人 CV 和研究计划到邮箱：bhanml@comp.hkbu.edu.hk。

智能创意

ACM MM'23 | 4 篇论文解析阿里妈妈广告创意算法最新进展

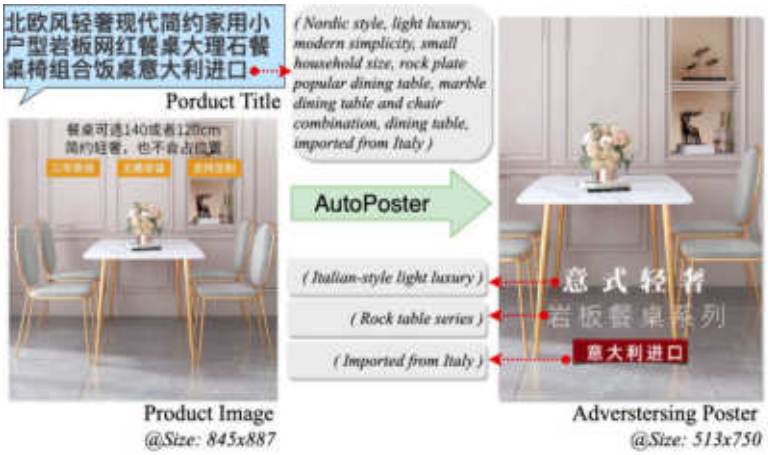
作者：内容平台智能创作

ACM Multimedia 是国际多媒体领域的旗舰会议。今年，阿里妈妈内容平台与智能创作团队总结业务中的创新，共有 4 篇文章入选 ACM MM 2023。内容涵盖新的图文创意制作系统、像素级的图文创意文案生成方案、风格化的图片视频的文案生成系统、视频延展系统，这些有趣的算法都在阿里妈妈创意制作中得到应用。本文通过论文速览为大家介绍这些最新进展，后续我们也将邀请论文作者详细解析论文思路和技术成果，欢迎持续关注～

AutoPoster: A Highly Automatic and Content-aware Design System for Advertising Poster Generation

一种高度自动化和内容感知的广告海报生成设计系统

简介：广告海报是一种结合了视觉和语言模态的信息展示形式。海报的创作过程涉及繁复的步骤，考验设计师的经验和创造力。本研究旨在提出一种高度自动化、内容感知的广告海报生成系统“AutoPoster”。仅需一张商品图和描述作为输入，AutoPoster 通过四个关键步骤自动产出不同尺寸的海报，包括图像重定向、布局生成、图上广告标语生成和视觉属性预测。为了确保海报在视觉上和谐，我们除了引入了两个基于内容感知的生成模型用于布局和标语生成，还创新性地提出了多任务视觉样式属性预测器（SAP）来同时预测视觉样式属性。另外，我们提出囊括超过 76000 张海报图的附有视觉属性标注的用于海报生成的数据集。用户研究和相关实验的定性和定量结果证明了我们系统的有效性，并验证了所生成的海报在美学上相对于其他海报生成方法的优越性。



给定商品图像、商品描述和目标海报尺寸作为输入，AutoPoster 系统可以自动化生成相应尺寸的完整海报创意。图中的斜体英文部分为相应中文文案的翻译

TextPainter: Multimodal Text Image Generation with Visual-harmony and Text-comprehension for Poster Design

TextPainter：基于视觉和谐和文本理解的多模态海报文本图像生成

简介：海报文本图像生成是指在海报背景图像上，根据给定的区域位置和文本内容，像素级地生成渲染文字内容的区域图像，从而确保整体海报在视觉和文字语义上呈现出和谐美观的效果。与学术界自然场景下的文本图像生成任务不同，海报文本图像生成需要考虑文本语义、图像语义以及颜色纹理等多维信息。同时，还需要学习海报中描边、高亮、渐变等更复杂的文字展现形式，可谓是一项具有挑战性的新任务。针对这一任务，我们提出了 TextPainter，一种全新的多模态方法，它充分利用上下文视觉信息和相应的文本语义来生成文本图像。具体而言，TextPainter 以全局 - 局部背景图像作为样式提示，并通过视觉和谐性来引导文本图像的生成。为了实现句子级和词级样式变化，我们还运用了语言模型，并引入了文本理解模块。此外，我们还创建了 PosterT80K 数据集，其中包含约 80K 个带有句子级包围盒和文本内容的海报注释。这一数据集将有助于进一步推动海报文本图像生成的研究。经过定量和定性实验验证，TextPainter 能够为海报生成视觉和语义上和谐的文本图像，这为广告海报（创意）的自动化制作提供了极大的帮助。



如图所示，第一二行分别为模型的输入、输出，在给定海报背景图、文本目标区域和内容的情况下，我们的模型能综合以上信息生成与商品主体颜色和谐、与周围背景颜色对比度高的文本图像，且能根据文本语义内容进行高亮（如第三列中的”1”）。

Visual Captioning at Will: Describing Images and Videos Guided by a Few Stylized Sentences

任意风格的视觉描述：少量风格化语句指导下的图像和视频描述生成

下载： https://arxiv.org/abs/2307.16399

简介： 风格化视觉描述旨在以特定风格生成图像或视频描述，使它们更具吸引力和情感上的适宜性。这一任务的主要挑战是缺乏针对视觉内容的配对风格化描述，因此大多数现有的研究都侧重于不依赖平行数据集的无监督方法。然而，这些方法仍然需要用具有风格标签的样本进行训练，并且生成的描述仅限于预定义的风格。为了解决这些限制，我们探索了少样本风格化视觉描述的问题。该任务旨在仅使用少量样本作为指导，生成任意风格的描述，而无需进一步的训练。我们提出了一个名为 FS-StyleCap 的框架，该框架由一个条件编码器 – 解码器语言模型和一个视觉映射

模块组成。我们的训练方案分为两步：首先，我们训练一个风格提取器，在没有标签的文本语料库上训练生成风格表征。然后，冻结提取器，并使我们的解码器根据提取的风格表征和经过映射后的视觉内容表征生成风格化描述。在推理阶段，我们的模型可以通过用户提供的风格示例，生成所需的风格化描述。根据评估结果，我们的少样本风格化描述方法优于性能最强的方法，并且与在有标签的风格语料库上进行训练过的模型性能相当。

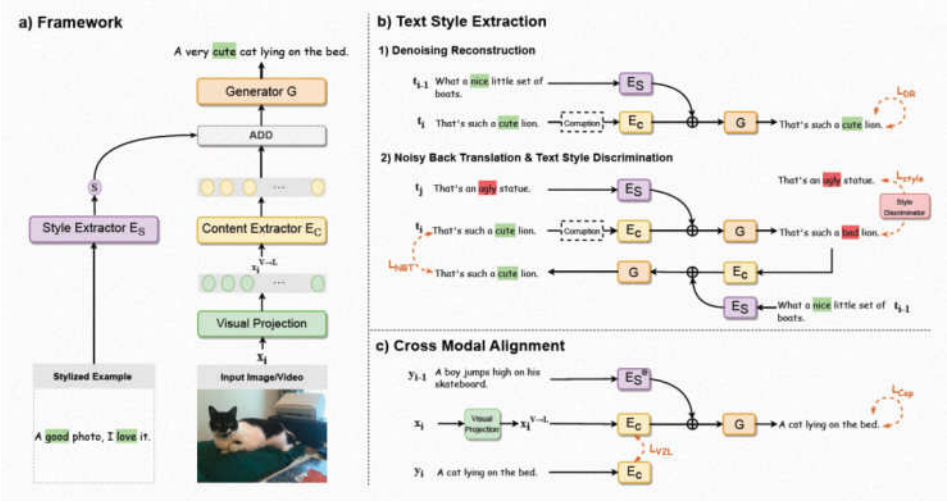


图 a) 展示了整体网络框架，共包括四个关键模块：风格提取器 E_S，内容提取器 E_c，生成器 G 和一个视觉投影模块；图 b) 展示第一个训练阶段，我们仅使用文本语料库训练 E_S。共包括三个训练子任务：去噪重建，带噪回译和文本风格分类；图 c) 展示第二个训练阶段，我们冻结 E_S，并使用视觉描述任务训练其他模块以实现跨模态对齐。同时，为确保能够生成风格化描述，我们还保持 E_S 冻结，使用文本风格注入任务 (如图 b) 所示) 同时训练 E_c 和 G。

Hierarchical Masked 3D Diffusion Model for Video Outpainting

用于视频延展的多级掩码 3D 扩散模型

简介：视频延展 (Video Outpainting) 是对视频的边界进行扩展的任务。在电商场景中，广告主提供的视频素材经常出现与 App 展示区域的尺寸不适配的情况，直接拉伸素材视频展示效果较差，而裁切会丢掉原素材的诸多元素，因此我们可以采用视频

延展算法来使得延展之后的视频的长宽比适配我们的广告展示区域的尺寸。考虑到广告主的视频长度普遍大于 10s，视频延展任务与图像延展任务相比，带来了以下额外的两个挑战：1). 推理阶段需要把视频分成多个片段来预测，如何保证多个片段间的时序一致性？ 2). 长视频延展存在误差累积的问题。为了解决上述的两个挑战，我们提出了以下优化：1). 我们基于 2D 图像扩散模型 (Stable Diffusion) 的参数先验构建了 3D 视频扩散模型，加速视频延展任务的快速收敛；2). 采用引导帧的方式来连接同一长视频的多个片段，为此我们提出了一种新的掩码预测方法来训练 3D 视频扩散模型；3). 为了更好地保持时序一致性，我们在交叉注意力层引入全局帧的信息，使模型在预测当前视频片段时，感知到全局的视频信息，以期望在有空间信息重叠时，模型能够填补更加合理的结果；4). 我们提出了一种混合策略的 Coarse-to-Fine 的推理流水线来缓解长视频误差累积的问题。本文提出的方法已上线至阿里妈妈创意中心，为广告主提供视频尺寸的一键修改。

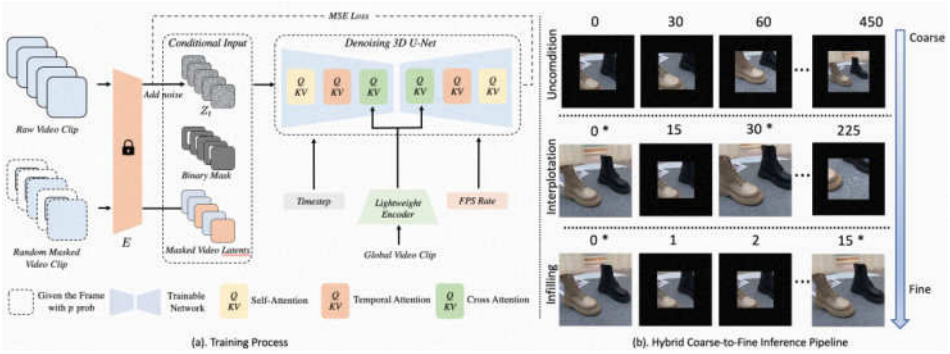


图 (a) 是视频延展扩散模型的训练流程；图 (b) 是视频延展多级化的推理流程

上下文驱动的图上文案生成

本文作者： 持信、弈臻、悟放、积流、孟诸

1. 摘要

为商品图片上特定位置配上装饰性文案来突出重点在广告业务中有着十分广泛的应用前景。然而，现有的图片文案描述生成系统均生成与图片位置关系无关的文案，无法很好地应用到广告业务中。在本文中我们提出了一种新的文案生成任务——图上文案生成，并基于商品图片数据提出了一个大规模图上文案数据集 CapOnImage2M。为了更好的利用上下文以及商品本身的信息来生成更适合特定位置的文案，我们提出了一种基于上下文的多模态图上文案模型，并设计了几种针对位置关系的预训练任务来帮助模型更好的理解位置信息。目前，使用该工作针对业务数据训练的模型，已经应用在淘宝首页焦点位、首页猜你喜欢信息流等广告业务中，并取得了显著的业务收益。该项工作论文已发表在 EMNLP 2022，欢迎阅读交流。

论 文：CapOnImage: Context-driven Dense Captioning on Image

下 载（点击↓阅读原文）：https://arxiv.org/abs/2204.12974

2. 背景

广告主通常会给商品图片配上特定的装饰性文案以突出重点，提升商品的吸引力和信息量，这些文案通常包括产品名、产品介绍、卖点、点击引导、利益点等类型。然而为图片设计合适的图上文案通常需要雇佣专业写手和设计师来完成，成本较高且相对低效。传统的图文创意是基于预设模板的方式，依赖设计师的模板去填充对应的文案，模板的多样性往往不足以匹配图片的多样性，导致模板的适配性不足，同时受限于模板的固定范式，要求我们具有明确指定各种文案类型和特定字数的文案生成能力，不够灵活且适配成本较高。

为此，我们希望提出一种自动化的图文创作方式，在本文中我们提出了图上文案生成，一种新的文案生成任务，利用多模态的文案生成技术，综合考虑图片本身信息（如商品主体、商品主体位置和背景色）、商品文本信息、文本框位置 layout 以及多个框之间的相对位置关系等信息自适应地生成合适的文案。其中文本框位置可以通过其他手段获取，比如 OCR 工具或者 layout 模型生成等。



图 1　图上文案生成任务示意图

3. 数据集

我们提出了 CapOnImage2M 数据集来作为图上文案任务的 benchmark。它包含 50 类共计 210 万业务图片，每张图片包含每个商品标题、属性和图片上不同位置的文案以及对应的坐标。



图 2　CapOnImage2M 数据集示例

4. 方法设计

4.1 概述

现有的图像文案生成通常是对整张图片生成一个整体的叙述性文案，缺乏对图片在空间上与对应文案的交互关系。在这个任务中，我们将在图片上的多个位置生成多个与之对应的文案。为此，**我们提出了一种基于上下文的多模态图上文案的模型，充分利用各模态商品信息去生成合理且多样的图上文案**。以上所有输入信息分别进行信息嵌入后输入一个混合模态的多层 transformer 中，模型通过自回归的方式生成预测的文案 [1,2]。为了更好的帮助模型理解上下文位置关系，我们提出了几种不同层级的位置相关的预训练任务，并利用 progressive training 的策略帮助模型训练。

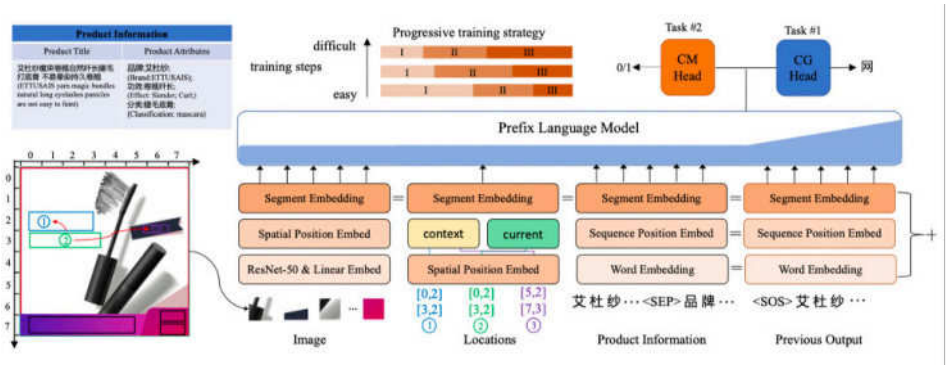


图 3　方法总览图

4.2 模型输入

模型将图片、当前位置框、前后位置框、商品类目、商品标题、商品属性对等作为输入，去生成对应当前位置框的文案。所有位置框的原始坐标为像素坐标，为了方便编码，我们将其进行离散化，具体来说，整张图从横纵两个方向切分为固定数量（示意图中以 7X7 为例）的格子 patch，将位置框的坐标所在的 patch 的横纵坐标作为框的 embedding id。图像部分的输入经过。在模型训练的过程中，由于图片上含有文字（即待生成的），为了避免模型坍塌成识别图上文字的 OCR 任务，我们对原图上的文字部分进行了 mask。

4.3 预训练方案

我们通过预训练微调的方式来训练模型，首先通过 Caption Generation（CG）和 Caption Matching（CM）两个任务对模型进行预训练，在微调阶段仅利用 CG 任务来生成文案。CG 任务我们使用 Prefix LM[1,2] 的方式进行解码；CM 任务与视觉语言预训练工作中常用的 Image-Text Matching[3,4,5] 任务类似，都是构造正负样本让模型来预测图片和 caption 之间是否匹配，在图上文案这个任务当中，为了帮助模型进一步的理解位置关系，我们设计了 3 种不同难度的负样本：

Level-I: Image caption matching. 第一种负样本是我们随机替换正确的 caption 为其它图片的 caption，我们希望模型能通过图片和商品信息来很好的识别这一类负样本。

Level-II: Location caption matching. 第二种负样本是将正确的 caption 随机替换为同一张图片上其它位置的 caption，我们希望模型能通过文字的位置信息更好

的理解文字的关系。

Level-III: Neighbor-location caption matching. 第三种负样本是将正确的 caption 随机替换为同一张图片上相邻（包含前后）位置的 caption，这一种负样本可以看做第二种的一种特殊形式。

因为三种负样本是从易到难，我们进一步提出利用 progressive training 的策略在训练过程中动态的调整样本难易，来进一步帮助模型理解位置关系。

5. 实验

5.1 定量分析

因为图上文案任务是一个新提出的任务，我们将传统的图像文本描述任务的相关模型适配到我们的任务上。从表一中可以看出我们的模型在准确性以及多样性的指标上均取得了最好的结果。



表 1　与基准模型对比

5.2 消融实验

我们进一步进行实验对我们提出的三种预训练 task 以及 progressive training 策略

的有效性进行验证，从表二可以看出，三种不同难度的预训练 task 均可以帮助模型更好的理解位置关系，进而提升模型效果；progressive training 的策略进一步提升了模型的效果。

Row	Pretrain tasks			Pretrain strategy		Validation				Test			
	Level-I	Level-II	Level-III	fixed	progressive	B@1	B@4	M	CIDEr	B@1	B@4	M	CIDEr
1	-	-	-	-	-	37.69	27.98	21.91	313.64	37.02	27.23	21.14	305.50
2	✓	-	-	✓	-	38.87	29.23	22.32	325.74	38.19	28.48	21.69	316.46
3	✓	✓	-	✓	-	40.22	30.36	22.88	339.72	39.36	29.54	22.03	329.77
4	✓	✓	✓	✓	-	40.55	30.71	23.34	347.30	39.40	29.57	22.51	335.49
5	✓	✓	✓	-	✓	**41.77**	**32.20**	**24.52**	**357.03**	**40.95**	**31.45**	**23.49**	**345.41**

表 2　消融实验

5.3 可视化分析

在图四中我们可视化了一些生成的 case，并与 ground-truth 进行了对比，可以看出模型生成的文案很好的理解了多模态信息以及上下文位置关系，生成了与位置相匹配的文案。



图 4　部分生成 case 可视化

6. 总结与展望

本文提出了一种新的文案生成任务——图上文案生成，在广告、社交平台图片等多个场景都有着很好的应用前景，我们基于商品图片提出了一个大规模数据集 CapOn-Image2M 以方便后续工作对任务进行进一步探索。我们相信自动化是图文创作的未来，希望本文工作能对后续广告文案自动化生成有所启发。

后续我们将持续改进图上文案生成的质量，并预期可以不借助于前置文本框预测模型和后置文字渲染模型，做到端到端的文字渲染。

7. 关于我们

我们是阿里妈妈创意 & 内容算法团队，致力于推动广告创意和内容投放产业的 AI 升级，努力推动创意制作、理解、模型预估和广告投放的全栈智能化。得益于阿里巴巴庞大而真实的营销场景，团队在图像技术、视频技术、文案生成、广告投放等领域持续发力和创新，现已构建出图片与短视频创意自动生成，创意个性化投放，智能文案写作，全自动与交互式抠图等特色产品，论文发表于 CVPR、ICCV、AAAI、ACMMM、WWW、EMNLP、CIKM、ICASSP 等领域知名会议。用 AI 赋能现代营销，驱动产业升级。真诚欢迎 CV、NLP 和推荐系统相关领域的同学加入！

投递简历邮箱：

alimama_chuangyi@service.alibaba.com

8. 引用

[1] Raffel C, Shazeer N, Roberts A, et al. Exploring the limits of transfer learning with a unified text-to-text transformer[J]. J. Mach. Learn. Res., 2020, 21(140): 1-67.

[2] Dong L, Yang N, Wang W, et al. Unified language model pre-training for natural language understanding and generation[J]. Advances in Neural Information Processing Systems, 2019, 32.

[3] Chen Y C, Li L, Yu L, et al. Uniter: Universal image-text representation learning[C]//European conference on computer vision. Springer, Cham, 2020: 104-120.

[4] Li G, Duan N, Fang Y, et al. Unicoder-vl: A universal encoder for vision and language by cross-modal pre-training[C]//Proceedings of the AAAI Conference on Artificial Intelligence. 2020, 34(07): 11336-11344.

[5] Zhuge M, Gao D, Fan D P, et al. Kaleido-bert: Vision-language pre-training on fashion domain[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2021: 12647-12657.

基于无监督域自适应方法的海报布局生成

作者：云芑

本文分享阿里妈妈内容平台与智能创作团队在图文广告创意方向上关于元素自动布局的探索与实践，结合无监督域自适应技术，有效提升了基于图像内容的图形布局生成效果。基于该项工作总结的论文已被 CVPR 2023 录用，欢迎阅读交流～

论 文：Unsupervised Domain Adaption with Pixel-level Discriminator for Image-aware Layout Generation

链 接：https://arxiv.org/abs/2303.14377

1. 背景

在广告投放过程中，为了吸引用户，我们需要根据不同的商品制作创意。根据历史实验的结果，创意的视觉美观度与点击效果呈正相关关系。然而，目前业界广泛使用的自动化创意制作方法都是基于固定模板的元素替换或属性更改，导致图形元素的位置不随商品图像的变化而改变。这种方法常常出现遮挡图像主体、视觉融合度不佳等问题，且缺乏独特性，容易引起视觉疲劳。

在学术研究中，已有一些自动生成布局的方法[1-3]，但它们主要侧重于建模布局内部图形元素之间的关系，未充分利用图像内容信息，无法解决上述问题。为了解决这个问题，我们提出了一种感知图像内容的创意布局自动生成方法[4]，可以根据商品图自动生成合理的布局。一种直接的方法是由专业设计师为无图形元素的干净产品图设计元素布局，以获取产品图 – 布局的配对训练数据，但这需要大量人力。为了减轻工作量，我们收集了已渲染好图形元素的海报图像，并标注了图形元素的位置和大小。然后，我们使用图像修复技术（inpainting）去除海报上的图形元素，并结合高斯模糊来减小处理后海报与产品图之间的差异。实验证明这种方法是有效的，训练后的模型可以为普通产品图设计布局，但图像修复可能会引入一些瑕疵或提示，而高斯模糊可能会破坏图像的色彩和纹理细节，导致一些不合理的遮挡或元素位置安排。

于是，我们提出利用无监督域自适应技术，来进一步缩小干净产品图和修复海报图之间的域差异，从而提高生成的图形布局的质量。为此，我们设计了一个带有像素级域判别器（Pixel-level Discriminator，PD）的 GAN，简称为 PDA-GAN，以更细粒

度地完成特征空间的域对齐。像素级域判别器 PD 可以代替 [4] 中的高斯模糊步骤，有助于模型感知图像的细节，进行更精准的布局生成。因为修复区域相对于整个图像通常很小，并且在具有大感受野的深层级别上很难辨别，PD 直接作用于浅层特征图。此外，它仅由三个卷积层构成，网络参数数量不到 [4] 中判别器参数的 2%。实验证明，这些设计使得不仅 PD 在内存和计算成本上都非常轻量，而且能有效提升生成的海报图形布局的视觉质量。

2. 方法

如图 1 所示，我们的网络主要由两个子网络组成：1）布局生成器器，它以图像及其显著性图作为输入，生成图形布局；2）像素级域判别器 PD。

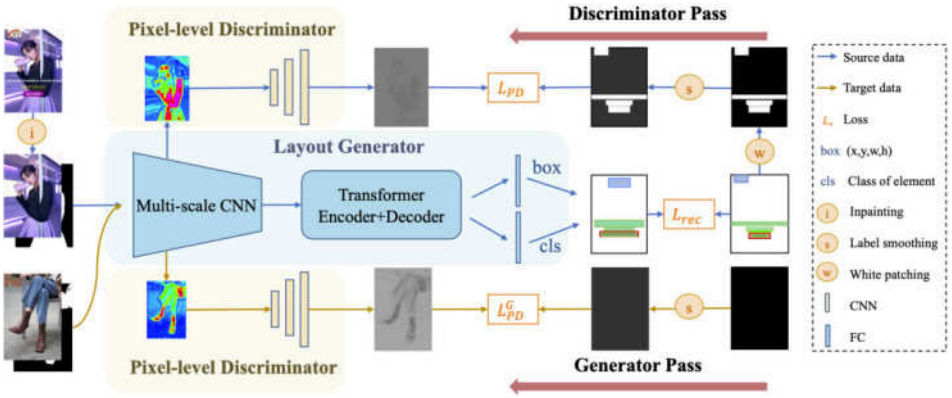


图 1　网络结构图

布局生成器的结构与 [4] 中的生成器相同，共包括三个模块：用于提取图像特征的多尺度卷积神经网络（CNN），接受布局元素查询作为输入的 Transformer 编码器 – 解码器，用于建模布局元素、图像之间的关系，以及两个全连接层，用于使用 Transformer 解码器输出的元素特征预测元素类别和包围盒。

像素级域判别器 PD 则是由三个大小为 3x3、步长为 2 的反卷积层组成，其输入是多尺度 CNN 中第一个残差块的特征图，输出的大小与整体输入图像的尺寸一致，以方便判别器损失的计算，因为修复图像与干净产品图像之间的域差距，主要存在于修复过程中合成的像素上。如图 1 所示，修复图像和干净产品图像的特征都会被送到判别器中。在更新判别器时，我们激励判别器去检测修复图像中的被修复像素。相反，在更新生成器时，我们利用像素级别判别器引导生成器输出浅层特征图来欺骗判别器，这意味着即使对于修复图像计算的特征图，判别器检测修复像素的能力也应该迅速减

弱。通过这种方式，当训练收敛后，源域和目标域图像的特征空间应该对齐。

为了计算损失 L_{PD}，我们可用和输入图像等尺寸的二值图来表示像素是否经过修复（修复为 1，代表源域，未修复为 0，代表目标域），作为判别器的监督。在 GAN 训练中更新判别器时，损失可表示为：

$$L_{PD} = \frac{1}{N_p}\sum_{i=1}^{N_p}(|\mathbf{P}_i^{s,w} - \mathbf{P}_i^{s,o}| * \alpha + \left|\mathbf{P}_i^{t,w} - \mathbf{P}_i^{t,o}\right| * \beta), (1)$$

其中 N_p 表示修复像素的数量，$\mathbf{P}_i$ 表示第 i 个图像的预测或真值。$\mathbf{P}_i$ 的上标表示它来自源域 s 或目标域 t，来自预测 o 或真实 w。两个系数 α 和 β 用于平衡源域和目标域损失之间的关系。

我们使用单侧标签平滑化 [5-6] 来提高训练模型的泛化能力。由于修复区域占据输入图像的一小部分，我们仅对不在修复区域中的像素进行标签平滑化，即图 1 中黑色区域的值为 0.2 而非 0。

布局生成器在 GAN 训练中更新时，我们希望欺骗更新后的判别器，使其无法输出值为 1 的像素。因此，损失 L_{PD} 被修改为惩罚生成器，如果判别器输出值为 1 的像素，则损失增加。因此，得到下式（其中 $\hat{\mathbf{P}}_i^s$ 中像素的值全部设置为 0.2）：

$$L_{PD}^G = \frac{1}{N_p}\sum_{i=1}^{N_p}(\left|\hat{\mathbf{P}}_i^s - \mathbf{P}_i^{s,o}\right| * \alpha + \left|\mathbf{P}_i^{t,w} - \mathbf{P}_i^{t,o}\right| * \beta), (2)$$

布局生成器的训练损失如下：

$$L_G = L_{rec} + \gamma * L_{PD}^G, (3)$$

L_{rec} 为生成图形布局与真实布局之间的重建损失，其计算方式与 [4] 相同。

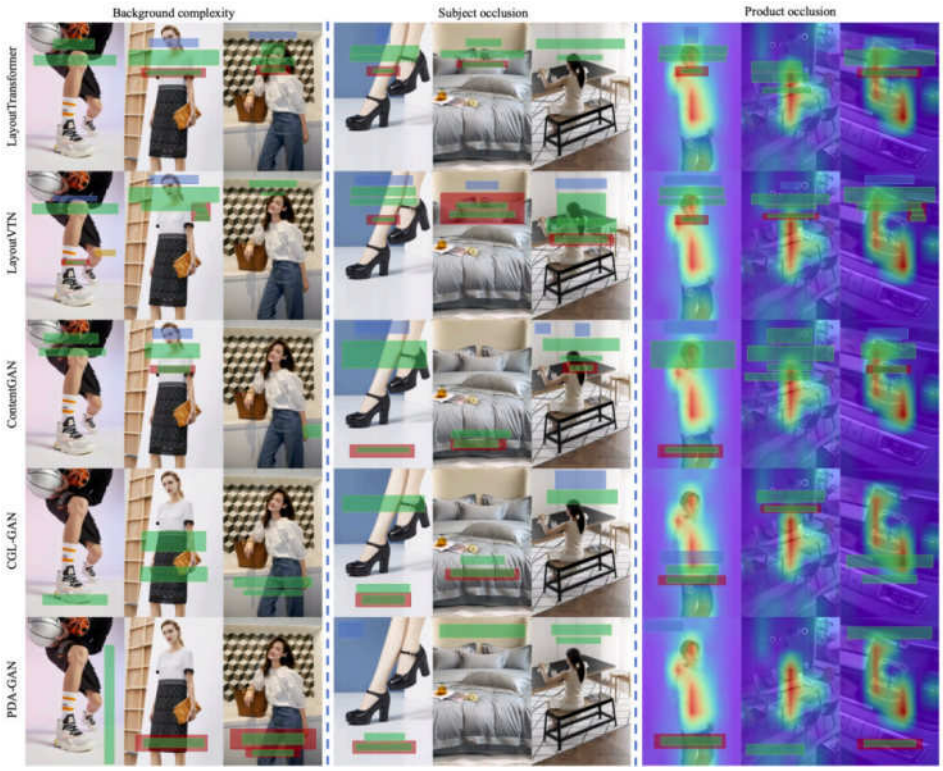
3. 实验

如下所示，我们将提出的方法与 SOTA 方法进行定量、定性的对比，其中 R_{com}、R_{shm} 和 R_{sub} 为与图像内容相关的指标，分别衡量背景复杂度、遮挡主体程度和遮挡产品程度；R_{ove}、R_{und} 和 R_{ali} 为与图像内容无关、仅与图形元素相关的指标，分别反应布局元素间重叠、衬底元素正确使用、对齐情况。R_{occ} 为模型预测的非空布局比例。可见 PDA-GAN 在大多数指标上取得了最佳结果，尤其是在与图像内容相关的指标方面。

此外，我们还通过了多个消融实验验证了模型结构、损失函数设计的有效性，具体详见论文。

Model	$R_{com}\downarrow$	$R_{shm}\downarrow$	$R_{sub}\downarrow$	$R_{ove}\downarrow$	$R_{und}\uparrow$	$R_{ali}\downarrow$	$R_{occ}\uparrow$
ContentGAN [34]	45.59	17.08	1.143	0.0397	0.8626	**0.0071**	93.4
CGL-GAN [35]	<u>35.77</u>	<u>15.47</u>	<u>0.805</u>	**0.0233**	<u>0.9359</u>	<u>0.0098</u>	<u>99.6</u>
PDA-GAN(Ours)	**33.55**	**12.77**	**0.688**	<u>0.0290</u>	**0.9481**	0.0105	**99.7**

Table 1. **Comparison with content-aware methods.** Bold and underlined numbers denote the best and second best respectively.



和 SOTA 方法的定性对比

关于我们

我们是阿里妈妈内容平台与智能创作团队，专注于图片、视频、文案等各种形式创意的智能制作与投放，以及短视频广告多渠道投放，产品覆盖阿里妈妈内外多条业务线，欢迎各业务方关注与业务合作。同时，真诚欢迎具备 CV、NLP 和推荐系统相关背景同学加入！

投递简历邮箱：alimama_chuangyi@service.alibaba.com

参考文献

[1] Diego Martin Arroyo, Janis Postels, and Federico Tombari. Variational transformer networks for layout generation. In CVPR, pages 13642 – 13652. Computer Vision Foundation / IEEE, 2021.

[2] Kamal Gupta, Justin Lazarow, Alessandro Achille, Larry Davis, Vijay Mahadevan, and Abhinav Shrivastava. Layouttransformer: Layout generation and completion with self-attention. In ICCV, pages 984 – 994. IEEE, 2021.

[3] Akash Abdu Jyothi, Thibaut Durand, Jiawei He, Leonid Sigal, and Greg Mori. Layoutvae: Stochastic scene layout generation from a label set. In ICCV, pages 9894 – 9903. IEEE, 2019.

[4] Min Zhou, Chenchen Xu, Ye Ma, Tiezheng Ge, Yuning Jiang, and Weiwei Xu.Composition-aware graphic layout GAN for visual-textual presentation designs. In IJCAI, pages 4995 – 5001. ijcai.org, 2022.

[5] Ian J. Goodfellow. NIPS 2016 tutorial: Generative adversarial networks. CoRR, abs/1701.00160, 2017.

[6] Christian Szegedy,Vincent Vanhoucke, Sergey Ioffe, Jonathon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision.In CVPR, pages 2818 – 2826. IEEE Computer Society, 2016.

基于内容融合的字体生成方法

作者：云芑

本文分享阿里妈妈内容平台与智能创作团队在字体自动生成上的探索与实践。我们提出了内容融合模块、迭代式风格向量微调、投影字符损失等方法有效提升了字体生成的效果，并助力于阿里妈妈东方大楷、刀隶体等字体的建设。基于该项工作总结的论文已被 CVPR 2023 录用，欢迎阅读交流。

论 文：CF-Font: Content Fusion for Few-shot Font Generation

下 载：https://arxiv.org/abs/2303.14017

代码链接：https://github.com/wangchi95/CF-Font

延展阅读：阿里妈妈品牌字“数智体”正式发布

1. 背景

在广告创意制作、网页设计等领域中，文字是传递信息的重要工具，而选择合适的字体可以显著影响整体视觉效果和阅读体验。然而，目前的开源字体数量有限，而商业字体需要支付昂贵的版权费用才能使用。因此，我们考虑建立自己的字体库。然而，常见的汉字就多达 6763 个，完全依靠设计师手工设计字体将会耗费大量时间和精力，因此我们开始考虑使用自动的少样本字体生成（few-shot font generation）的方法，即仅根据少量目标字体的参考图像（目标域），将字体图像从已有字体（源域）转换到目标域，来生成新字体的所有字符。通过这种方法，只需要手动设计几十个字符，便可得到对应的新字体。

在少样本字体生成中，最常用的是将内容和风格解耦的方法 [1]，用双分支网络分别学习内容特征和风格特征，其中内容特征来自手动选择的字体字符图像，风格特征来自于目标字体的字符图像。然而，由于完全分离内容和风格特征的困难，源字体的选择对生成结果产生显著影响。例如，宋体和楷体通常被选作源字体。虽然这些选择在许多情况下都是有效的，但是生成的图像有时会包含瑕疵，例如缺失和冗余的笔画等。

本文提出了一种新颖的内容特征融合方案，通过探索内容和风格特征的同步来减轻不完全分离的影响，从而显著提高少样本字体生成的质量。我们设计了一个内容融合模

块（CFM），在训练和推断过程中考虑不同字体的内容特征。该模块通过线性混合基字体的内容特征来计算目标字体字符的内容特征，混合权重由精心设计的字体级距离度量来获取的。通过这种方式，可以为语义字符的内容特征形成一个线性聚类，并探索如何利用字体级相似性在该聚类中寻找优化的内容特征来提高生成字符的质量。

此外，我们引入了一种迭代式风格向量微调（ISR）策略，以寻找更好的字体级风格特征向量。对于每种字体，我们对参考图像的风格向量进行平均，并将其视为可学习的参数。然后，我们使用重建损失微调样式向量，进一步提高生成字体的质量。

大多数字体生成算法选择 L1 损失作为字符图像重建损失。然而，L1 或 L2 损失主要监督每个像素的准确性，容易受到细节局部错位的干扰。因此，我们采用基于分布的投影字符损失（PCL）来衡量字符之间的形状差异。具体而言，通过将二维字符图像的一维投影视为一维概率分布，PCL 计算距离，更加关注字符形状的全局属性，从而显著提高骨架拓扑迁移结果。

2. 方法

我们的方法如图 1 所示，整个训练过程可以分为两个阶段。首先，我们使用 DG-Font [1] 中提出的神经网络作为基础网络进行训练，该网络被称为 DGN。该网络用于学习基础知识，即对数据集中字符图像解耦的内容和风格特征。其次，CFM 被插入到模型的内容编码器后。然后，原始的内容特征被替换为 CFM 的输出，即使用自动选择的基字体的线性内容特征插值。然后，我们固定内容编码器，继续训练风格编码器、特征变形跳跃连接 (FSDC) 和混合器。投影字符丢失 (PCL) 被用于训练以监督字符骨架。此外，为了进一步提高生成质量，我们利用迭代式风格向量微调 (ISR) 策略，仅在推理中完善学习到的字体级风格向量。ISR 的提出动机是寻求单一高质量的字体级风格向量来生成字体的所有字符的图像。具体来说，对于一个字体，我们对给定的 16 个字符（该文少样本设定）的平均风格向量进行精炼。

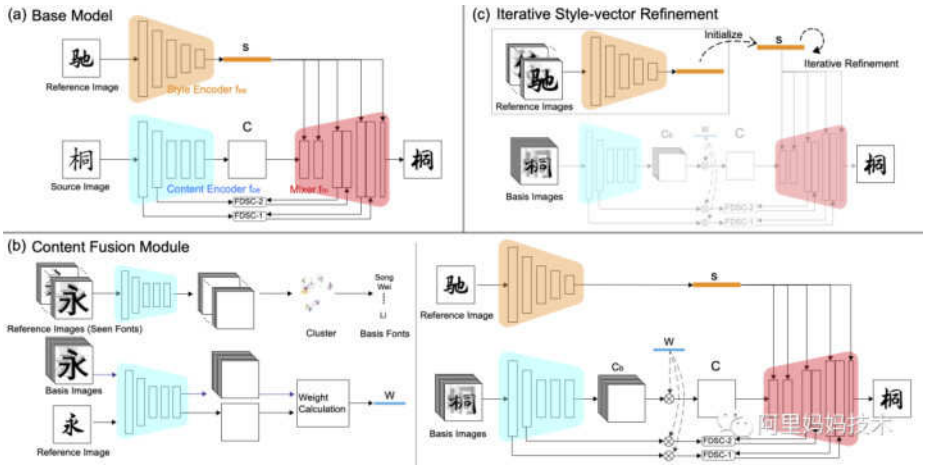


图 1　方法整体框架

2.1 内容融合模块

内容融合模块旨在通过结合基字体的内容特征来自适应地提取内容特征。这个带有 CFM 的网络结构如图 1 (b) 所示。首先，为了找到用于内容融合的具有代表性的字体，我们通过给定 16 个字符的级连内容特征对所有训练字体进行聚类，并选择最接近聚类中心的字体作为基字体。基字体一旦选定就被固定了。然后，对于每个目标字体，我们根据其与基本字体的相似性，计算 L1 范数内容融合权重。因此，内容特征（解码器的输入）被替换为基字体的加权和。此外，网络需进行微调以适应融合后的内容特征（如图 2 的蓝色圆圈所示）。

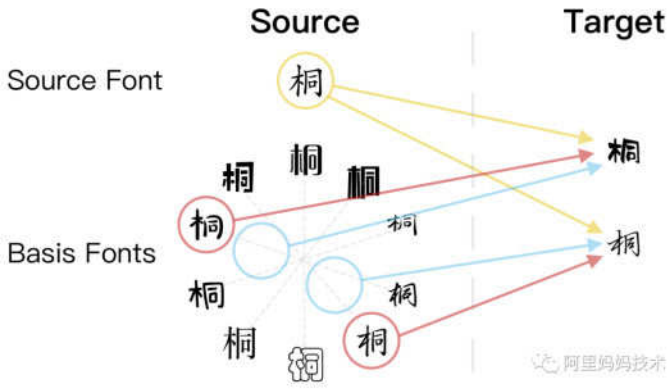


图 2　内容融合可视化

基字体选择：假设我们需要从 N 个训练字体中选择 M 个基础字体。它可以通过聚类

内容特征 $\{\boldsymbol{C}_i\}_{i=1}^{N}$ 并选择位于聚类中心的字体来实现。在具体实现中，由于 $\boldsymbol{C}_i$ 的维度太大而 N 相对较小，$\boldsymbol{C}_i$ 被映射到它与所有字体内容特征 $\boldsymbol{e}_i \in \mathbb{R}^N$ 之间的距离向量。进一步来说：

$$\begin{aligned}
&\boldsymbol{C}_i = f_{ce}(\boldsymbol{I}_i),\\
&\boldsymbol{d}_i = (d_{i1}, d_{i2}, \ldots, d_{iN}), \quad d_{ij} = \|\boldsymbol{C}_i - \boldsymbol{C}_j\|_1,\\
&\boldsymbol{e}_i = \sigma(\boldsymbol{d}_i),\\
&\mathcal{B} = \mathbf{Cluster}(M, \{\boldsymbol{e}_1, \boldsymbol{e}_2, \ldots, \boldsymbol{e}_N\}),
\end{aligned} \tag{1}$$

注：$\boldsymbol{C}_i$ 实际上是从字符中提取的级联内容特征。为简洁起见，这里省略了字符的上标。$\sigma(\cdot)$ 是 softmax 操作，d_{ij} 是两个字体之间的 L1 距离，$\mathbf{Cluster}$ 是经典的 K-Medoids 聚类算法，$\mathcal{B}$ 集合是所选字体的索引。

权重计算：对于目标字体 t 及其内容特征 $\boldsymbol{C}_t$，我们测量其与基础字体 $\{\boldsymbol{C}_m\}_{m=1}^{M}$ 的相似性，即 $\boldsymbol{d}'_t \in \mathbb{R}^M$。那么内容融合权重 $\boldsymbol{w}_t \in \mathbb{R}^M$ 的计算如下：

$$\begin{aligned}
&\boldsymbol{d}'_t = (d_{t1}, d_{t2}, \ldots, d_{tM}), \quad d_{tm} = \|\boldsymbol{C}_t - \boldsymbol{C}_m\|_1,\\
&\boldsymbol{w}_t = \sigma(-\boldsymbol{d}'_t/\tau),
\end{aligned} \tag{2}$$

内容融合：一旦获得基础字体和内容融合权重，原始的内容特征映射将被替换为融合后特征 $\boldsymbol{C}'_t$，其中 CFM 的内容融合权重与其目标字体相关。

$$\boldsymbol{C}'_t = \textstyle\sum_{m \in \mathcal{B}} w_{tm} \cdot \boldsymbol{C}_m. \tag{3}$$

2.2 投影字符损失

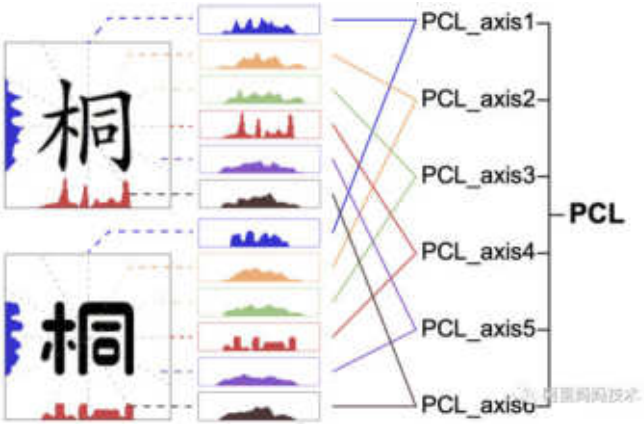


图 3　投影字符损失可视化

为了更好地监督骨架，我们增加了一个投影字符损失，它使用多个一维投影的边缘分布距离来度量图像差异，如图 3 所示。由于分布对于相对关系非常敏感，PCL 更注重字符的全局形状。

$$\mathcal{L}_p(\boldsymbol{Y}, \hat{\boldsymbol{Y}}) = \frac{1}{P}\sum_{p=1}^{P}\mathcal{L}_{1d}(\phi_p(\boldsymbol{Y}), \phi_p(\hat{\boldsymbol{Y}})), (4)$$

其中 $\boldsymbol{Y}$ 和 $\hat{\boldsymbol{Y}}$ 分别代表生成图像和真实图像，P 是投影的数量，$\phi_p(\cdot)$ 表示具有第个方向的投影函数，可以衡量一维分布之间的一致性，例如 KL 散度、最优传输距离均可。

2.3 迭代式风格向量微调

对于目标字体，通过将 f_{se} 的输出与一组字符图像进行平均，可以将鲁棒的样式信息提取为潜在风格向量 $\boldsymbol{s}'_t$。受到“迭代推理”策略 ([3]) 在推理阶段对输入优化的启发，我们提出了迭代式风格向量微调方法，以进一步优化风格特征。如图 1 (c) 所示，在推理阶段，首先将 $\boldsymbol{s}'_t$ 作为初始化；然后使用目标字体提供的少量参考字符作为监督样本，根据重建损失的反向传播对进行约十轮微调。最后，采用优化后的风格向量进行推理。值得注意的是，这个风格向量可以作为目标字体的标识进行存储，并在引用同一字体的所有字符时复用，这使得 ISR 在实际系统中具有高效性。

3. 实验

我们收集了 300 种中文字体 (包括印刷和手写字体) 来构建数据集，用于验证中文字体生成方法。字符集总共包含 6446 个字符，几乎涵盖了 GB/T2312 标准中文字符集的大部分。训练部分包含 240 种字体，每种字体 800 个字符。测试部分分为两部分：(a) 229 种训练集中出现过的字体，每种字体 5646 个训练集未出现的字符；(b) 60 种训练集中未出现的字体与 5646 个训练集未出现的字符，用于验证模型的泛化能力。

我们将提出的模型与 SOTA 方法进行比较，包括图像到图像的转换方法 (FUNIT) 和与组件相关的方法 (LF-Font，MX-Font，CG-GAN，FsFont，DG-Font)，在对比中，我们的方法在定量和定性对比中表现出更好的结果，在 L1 和 FID 指标方面分别比未见过字体的最先进方法高出 5.7% 和 5.0%。此外，我们还进行了问卷调查、消融实验等方式来验证了我们方法的有效性，具体可见论文。

Methods	Seen Fonts					Unseen Fonts					User Study %
	L1↓	RMSE↓	SSIM↑	LPIPS↓	FID↓	L1↓	RMSE↓	SSIM↑	LPIPS↓	FID↓	
FUNIT	0.08591	0.2529	0.6661	0.1169	**11.66**	0.09377	0.2686	0.6432	0.1427	28.10	11.74
LF-Font	0.08098	0.2435	0.6829	0.1226	27.73	0.09037	0.2620	0.6534	0.1448	38.46	13.01
MX-Font	0.07470	0.2319	0.7038	0.1034	18.75	0.08171	0.2468	0.6830	0.1193	27.91	10.86
Fs-Font	0.08214	0.2519	0.6657	0.1502	45.33	0.08917	0.2657	0.6467	0.1647	55.21	12.03
CG-GAN	0.07977	0.2409	0.6883	0.1117	23.93	0.08639	0.2549	0.6690	0.1303	37.22	16.67
DG-Font	0.06251	0.2105	0.7437	0.0846	17.10	0.07841	0.2442	0.6853	0.1198	27.98	14.11
CF-Font	**0.05997**	**0.2053**	**0.7538**	**0.0836**	13.13	**0.07394**	**0.2354**	**0.7007**	**0.1182**	**26.53**	**21.53**
	(4.1%)	(2.5%)	(1.4%)	(1.1%)	(-)	(5.7%)	(3.6%)	(2.3%)	(0.92%)	(5.0%)	(29.5%)

表 1　与 SOTA 方法在 seen/unseen fonts 上的定量对比

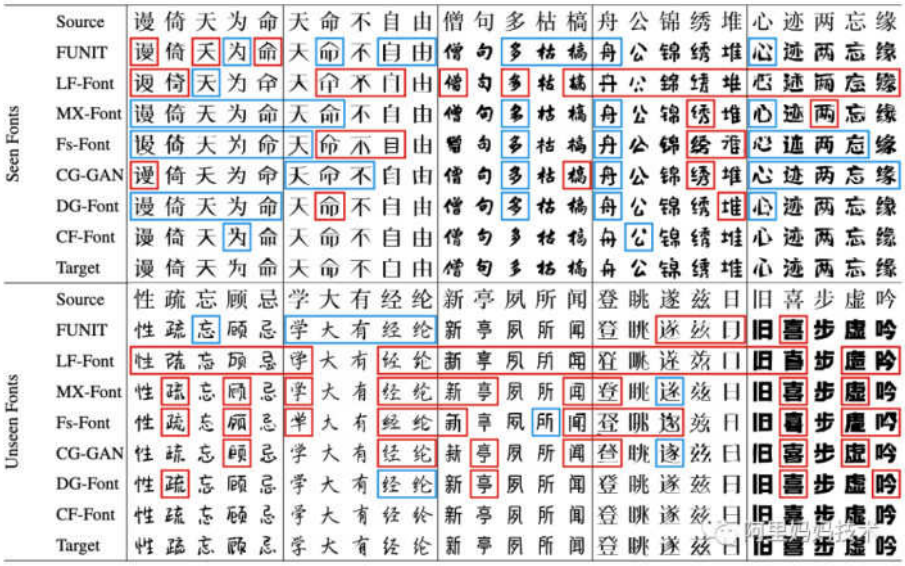


图 5　与 SOTA 方法在 seen/unseen fonts 上的定性对比

4. 业务应用

目前，基于该方法使用少量设计师拓碑字体和现有开源可商用字体进行初步生成，再结合人工精修，已完成了多款阿里妈妈自有字体的建立（如东方大楷、刀隶体等），详见《阿里妈妈智能造字，设计赋能商业再升级》。





关于我们

我们是阿里妈妈内容平台与智能创作团队，专注图片、视频、文案等各种形式创意的智能制作与投放，以及短视频广告多渠道投放，产品覆盖阿里妈妈内外多条业务线，欢迎各业务方关注与业务合作。同时，真诚欢迎具备 CV、NLP 和推荐系统相关背景同学加入！

投递简历邮箱：alimama_chuangyi@service.alibaba.com

参考文献

[1] Yangchen Xie, et al. Dg-font: Deformable generative networks for unsupervised font generation. In CVPR 2021.

[2] Ming-Yu Liu, et al. Few-shot unsupervised image-to-image translation. In ICCV 2019.

[3] Song Park, et al. Few-shot font generation with localized style representations and factorization. In AAAI 2021.

[4] Song Park, et al. Multiple heads are better than one: Fewshot font generation with multiple localized experts. In ICCV 2021.
[5] Yuxin Kong, et al. Look closer to supervise better: One-shot font generation via componentbased discriminator. In CVPR 2022.
[6] Licheng Tang, et al. Few-shot font generation by learning fine-grained local styles. In CVPR 2022.

化繁为简，精工细作——阿里妈妈直播智能剪辑技术详解

作者：柴风

1. 导读

今天越来越多的消费者在直播间下单消费，直播购物也成为了一种全新的消费体验，而直播视频作为创意载体可以为消费者提供丰富的商品讲解内容。直播通常长达数小时且包含较多空场、闲聊等与商品无关的信息，为了将直播中最精彩的内容剪辑并呈现给用户，阿里妈妈探索了一系列对直播视频加工剪辑的方案。对于正在进行直播的直播间，我们可以实时剪辑出高质量的直播片段提供给广告分发引擎，作为广告创意和落地页内容，提升广告效率。本文将详细介绍我们在直播智能剪辑技术方面的探索与实践，欢迎阅读交流。

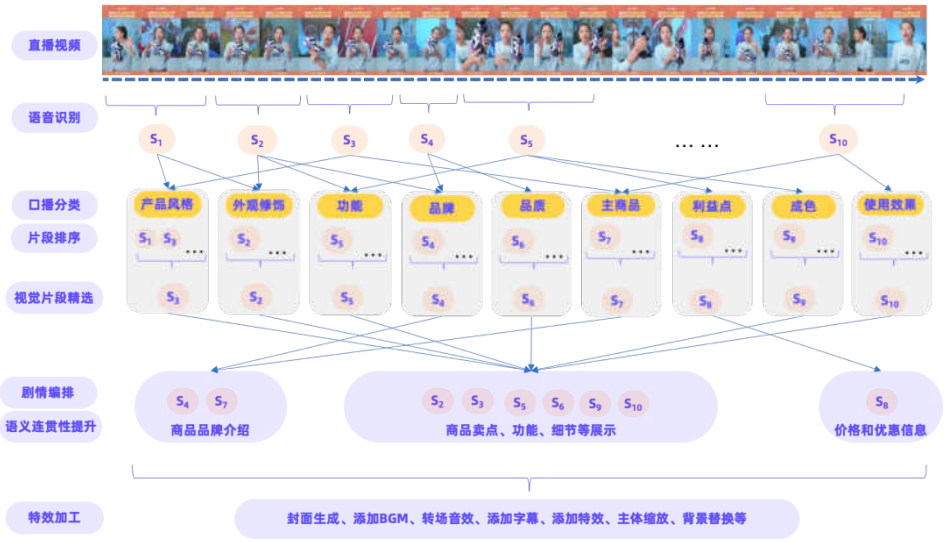
2. 智能剪辑技术方案

为了实现直播内容的自动化剪辑，我们通过分析直播视频、音频内容，完成对直播的精简、编排与加工，并搭建了直播间视频实时剪辑和投放链路。直播剪辑技术方案主要分为直播流视频拆解、口播内容理解、视频片段精选、剧情编排、特效加工等部分。具体步骤如下：

- **片段拆解**：首先，我们根据直播看点信息进行直播流文件拆解，将数小时直播流文件以商品粒度拆分为多个片段，大多数情况下每个片段约 3 ~ 10 分钟。
- **口播分类**：接下来进行片段内容分析和理解，包括口播内容和视觉内容两方面。在口播内容理解过程中，先对音频部分进行语音识别，得到口播文字内容，然后对口播文字进行分类，判断当前内容与商品哪些特性相关，比如产品风格介绍、功能介绍、品牌调性介绍、利益点介绍等。
- **视觉筛选**：得到口播内容类别之后，我们就可以将内容相似的直播片段进行聚合，然后再利用视觉片段的排序和优选，选择出多个具有较好视觉表现力的视频片段。
- **剧情编排**：我们对所有视觉片段进行剧情编排，通过各个片段的口播、视频类别进行组合，得到很多逻辑多样的组合视频，代表不同的剧情逻辑。

- **特效加工**：对每个组合视频进行特效和后期加工，包括封面生成、BGM 添加、直播背景替换、字幕、转场音效、特效贴纸、主体 / 局部变化等等，渲染在一起得到直播剪辑短视频。

具体剪辑流程图如下：



直播智能剪辑技术方案

接下来将基于上述方案步骤详细介绍直播剪辑的技术细节。

3. 方案细节介绍

3.1 口播分类 – 模型方案与语料

为了有效分析直播内容，我们根据电商口播的特点和视频剪辑的需要，确定了 21 类的口播语义标签，涵盖了品牌品类、利益点、卖点修饰类、直播信息等几个方面，同时保证标签能兼容大多数常见品类。一个句子往往包含多个语义，即对应多个标签，因此我们将该任务建模为句子的多标签分类任务。我们基于 GPT 预训练模型构建了一个多标签分类模型，然后在训练语料上进行训练。

语料来自多种品类和不同风格的多个主播，以保证语料的通用性和无偏性，语料标签经人工标注。相对传统的文本分类任务，口播语料往往断句不准且过于口语化，使得识别难度较大。比如下面一些语料示例：

例句	标签
唉，但这个我跟你说有点像吃年糕那种口感，它是那种小的Q弹的感觉	质地
是我们家一个紫色的一个款式，超级显腰身哈，超级显瘦啊，非常的收腰遮胯	外观修饰、使用效果
给大家讲解一下，喜欢的话呢就准备去入手了34号链接	下单引导
hello,亲爱的宝宝们，欢迎大家来到我们直播间，稍等一下，帮大家来设置一下优惠券	直播间信息、利益点
这款如果你是干皮的话，叠加一点乳液就可以可以使用，而且它不会卡纹，不会搓泥	适用对象、使用效果
来这个杏色给大家搭配一下，这个杏色可以搭配我们家咖色短裤	外观修饰、使用方式
今天在直播间然后还添加了一个菠萝蛋白酶，能够舒缓我的脸上，清洁效果会来的更加好	原材料、使用效果
嗯，这样的小花边的一个设计，袖子有一点点翘出来	外观修饰
我们的原料是蛇胆原汁精华和植物提取物，对止痒消包会红有很好的效果	原材料、功能
然后的话准备去给大家过一下下一个款	其他

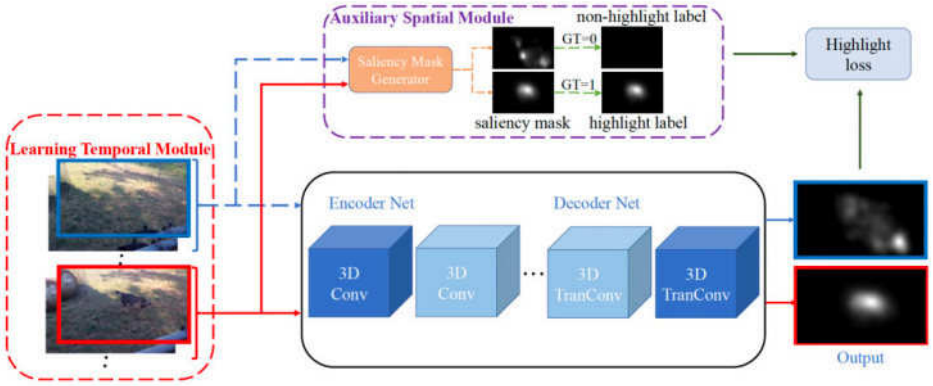
训练后的模型在直播间口播分类任务上达到 96% 的准确率。

3.2 片段排序与精选

片段的精选主要依赖画面的排序，我们利用视频高光剪辑算法，在口播标签相似的片段中选择视频片段中的高光片段，并给出对应的高光分数。算法介绍如下：

3.2.1 基于像素差异学习的视频高光检测算法

PLD-VHD（Learning Pixel-Level Distinctions for Video Highlight Detection）**方法的设计思路主要来自于我们对视频吸引力和观看者反应之间的关系的认识。** 我们认为在观看者观看视频的过程中，他的眼动信号是对内容的一个较强反馈。如果当前内容吸引他，他的注意力会集中在吸引他的区域，且这一段时间内该区域的内容往往具有一致性。如果当前片段缺乏吸引力，他的注意力会比较分散，不具备参考意义。基于以上假设，我们尝试对眼动信号和高光内容之间的关系建模。在计算机视觉任务中，主体检测任务是眼动信号的直接建模任务。于是，我们通过引入视频主体检测任务作为辅助，融合时间序列内的帧信息，学习当前帧的像素之间的差异，从更加细粒度的视角反应当前内容的吸引力。方案如下图所示：



基于像素差异学习的视频高光检测方法（PLD-VHD）

利用该模型，每一个视觉片段都会被打上一个高光分数，基于分数排序即可筛选出最优片段。

3.3 剧情编排

3.3.1 剧情编排介绍

剧情编排用于确定结果视频的各个阶段所展现内容语义，决定了视频的内容逻辑。剧情确定了生成视频的主干，以易于理解的方式将信息展现给用户。我们建立了成熟的直播剧情脚本生成方案，做到了去除冗余信息，精简直播卖点和利益点。比如下图是一种通用的剧情编排模式：



3.3.2 打造行业剧本——千行千面

在基于剧本的剪辑过程中，我们首先预设剪辑的剧本结构，根据剧本逻辑选择片段、

优化视频的讲解顺序，然后生成由各个高光片段组成的剪辑视频。剪辑后视频较原始视频有着更强的节奏感和逻辑性，但仍要解决以下几个问题：

1）依据固定剧本剪辑后的视频，剧本中需要的内容片段可能无法完全匹配，导致制作的视频整体逻辑不完整

2）剪辑后的视频逻辑单一，无法覆盖不同行业对直播内容需求的差异

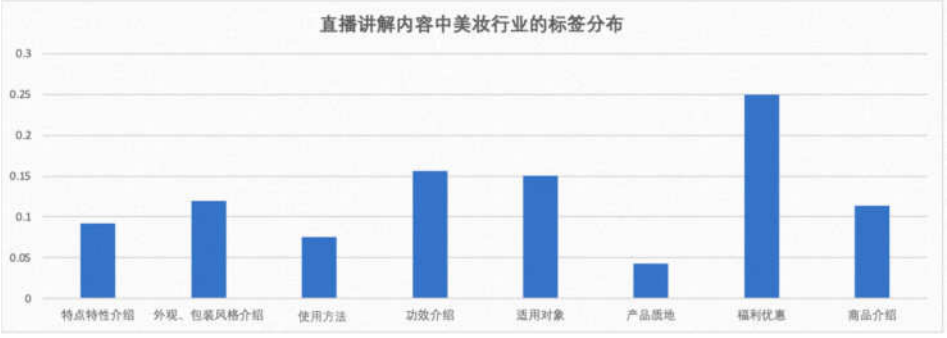
3）不同行业投放效果的影响因素不同，需要针对行业分析剧本效率

为了解决这几个问题，我们针对不同行业分析直播视频中的标签分布和投放情况，并根据行业特性定义符合行业特点的剧本方案。包括下面三个步骤：

① 行业标签分布分析

剪辑的第一个环节中，利用口播语义分类模型完成了对视频片段分类。我们根据对片段类别统计各个行业的口播内容标签分布，首先根据行业特点对标签进行归类合并处理，选择出现频率较高的标签作为当前行业剧本组合的主要参考标签。

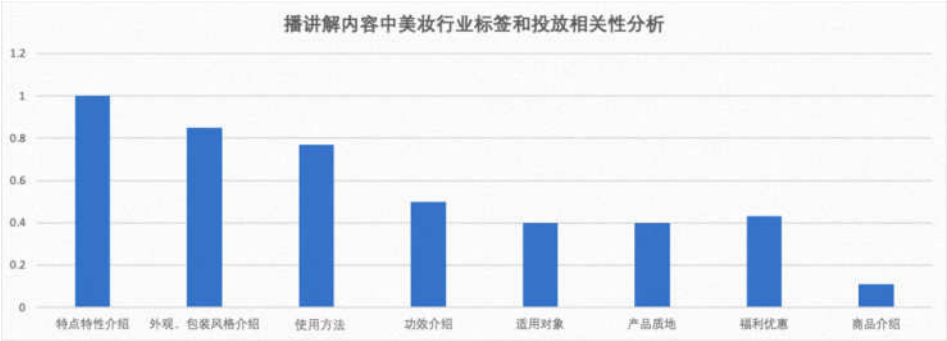
比如，直播讲解在美妆行业的标签分布情况：



② 行业标签和投放相关性分析

通过对视频投放效果和视频内容的分析我们发现，视频内容的丰富度和介绍侧重点会影响到投放效果。我们基于视频的内容标签和投放效果（广告投放 ROI），进行了投放效果和内容标签的相关性分析，分行业总结了影响行业投放的主要标签分布情况：在视频中，如果某一标签的分布（出现频率、时长）和其投放的 ROI 有较强的正相关性，则认为该标签对 ROI 有一定的正向影响。

比如，通过相关性分析后美妆行业的标签相关性（归一化后）：



③ 确定行业剧本

根据行业标签和投放相关性分析结果，我们可以确定在行业内部哪些标签内容对成交的帮助最大，据此可以根据内容的相关性定义行业剧本的结构，比如：将相关性最大的内容放在视频的开头优先透出，让用户能够第一时间获取关键信息，将最次要信息延后讲解。按照这样的剧本制定规则，我们尽量将对成交帮助大的内容提前，针对每个行业定义对应的行业剧本。

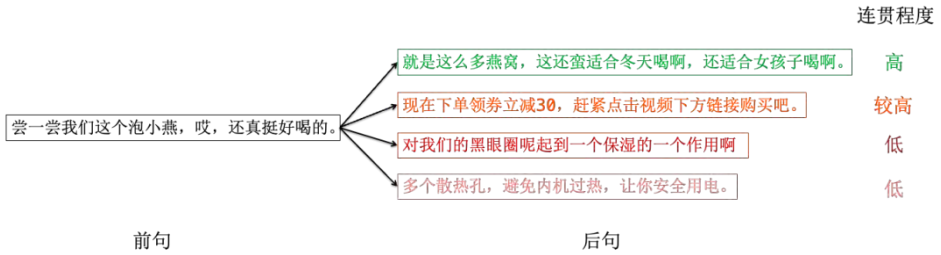
比如，根据相关性分析的结果，我们可以定义美妆的行业剧本如下：



在生成的视频中，优先展示“特点特性介绍”部分，然后介绍“外观包装风格”，以此类推，生成跟行业特点相关的高光视频。

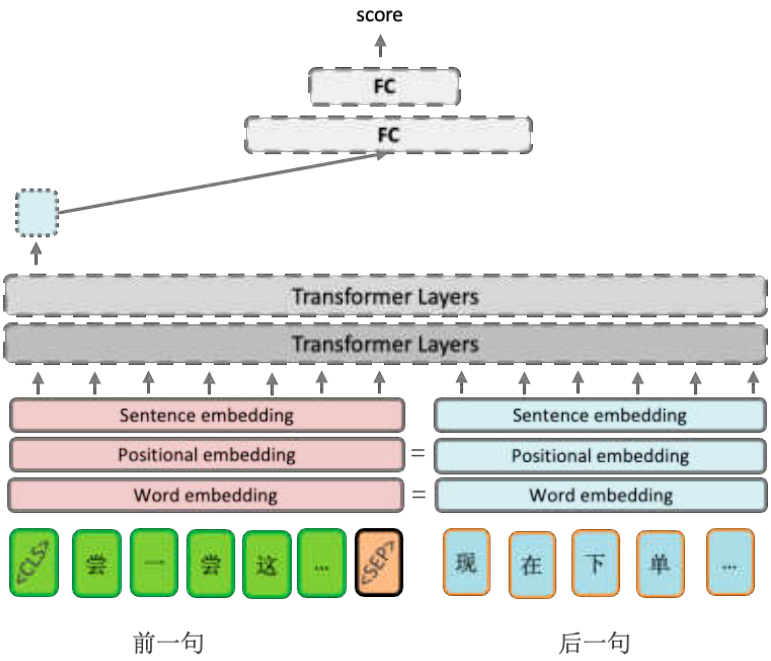
3.4 口播语义连贯性打分——让内容拼接更流畅

前面提到，在剧情编排阶段会对视频片段进行重组。由于片段重组的存在，不可避免地会出现**前后片段在口播语义不够连贯**的问题，比如前一句是“如果你皮肤也长红点的话”，下一句是“里面的这种橙色小颗粒是养肤的维 E 粒子”，会发现句子间的过渡生硬且没有逻辑，影响整体的剪辑质量。为解决这个问题，我们训练了一个打分模型对视频片段拼接时对其进行口播的语义连贯性打分，用以评估剪辑视频在连贯程度上的质量，从而可以对片段间的组合方式进行优选。值得注意的是，这里的“不连贯”既表示前后句在语法上不够通顺连贯，也表示在语义上不符合逻辑。



3.4.1 技术方案

传统使用计算困惑度（perplexity）来表示语句的“合理性”的方法对于较为口语化的口播文本并不适用。我们选择使用直播的口播语料来训练一个用于评估句子对（sentence pair）连贯性的打分模型，得分介于 0~1 之间，得分越高表示该句子对连贯性越好。模型结构如下图，句子的 embedding 使用基于 GPT 的预训练模型进行嵌入，为区分前句和后句，分别对两句增加不同的 sentence embedding，并在两句中间增加 [SEP] 标记用以表明句子边界。



3.4.2 语料构建

为了与应用场景保持一致，训练语料也来自于直播的口播，通过 ASR 将口播的音频

识别为口播文案。这里同样使用整句作为识别粒度，以便和视频片段的粒度保持一致。对于正负例的构建，很自然地，选取口播文案中真实相邻的前后句子对作为正例，而为了让模型见到足够丰富且多样的不合理的句子搭配，负例的选择则更为重要，我们采用 3 种句子对搭配方式来构建负例语料：

1）第一种是**句子对调**，即将正例的两个句子进行前后对调，以便让模型辨别前后相邻句的语序；

2）第二种是**同篇随取**，即从同一商品讲解中的所有句子中随机选一句与当前句搭配，从而让模型辨别“同一话题”下（即同一商品）非原生相邻句，这样可以避免模型塌陷为单纯的句子相关性打分，而非连贯性打分。

3）第三种是**不同篇随取**，即从不同商品讲解中句子中随机选一句与当前句搭配，以便让模型具备辨别“不同话题”的能力。

下面是一组正负样本的示例：

前一句	后一句	正负例
女神水来喽，第二代的女神水做到了平价又好用的一款水。	包括它里面还有一些后生元，就是帮助大家去调节肌肤。	正例
	第一个成分二列酵母发酵产物，溶孢物。	负例：同篇随取
	它不粘牙的糖醋牛轧糖是不粘牙的。	负例：不同篇随取
它里面还有一些后生元，就是帮助大家去调节肌肤。	女神水来喽，第二代的女神水做到了成分身体平价又好用的一款水。	负例：句子对调

模型经过训练，准确率达到 80%，AUC 为 84%。从测试集及实际打分效果来看，模型确实可以有效地衡量句子对的连贯程度。由于模型仅针对一个句子对进行打分，在衡量一组片段的整体连贯性时，分别对每个相邻句计算连贯性，然后根据平均分和最小分的排序策略进行倒排，从而在众多组合中剔除较差组合，选出较优组合。

3.5 直播高光直达标签——讲解大纲一目了然

基于已有的视频内容的理解能力，可以挖掘出一些用户关心的或者商家想强调的信息并加以呈现，从而提高用户对商品内容的感知。在这一方面我们做的尝试是高光直达标签。直达标签是对视频内容按照某种用户感兴趣的粒度进行切分，并提炼出各部分

的核心内容作为标签。



上图是一个真实的投放场景的示例，直达标签位于屏幕下方、进度条上方的位置，这里的第 0s~ 第 7s 是在讲解福利优惠，第 8s~ 第 17s 是在讲解功能介绍，等等。通过直达标签，用户不仅可以快速地感知到剪辑视频的“内容大纲”，还可以通过点击标签自由地跳转到标签对应的视频讲解位置，从更高效地让用户感知到商品讲解的关键信息。

在标签内容确定时，我们做了不同内容文案的尝试，因为清晰、直接的标签的内容可以吸引用户进行点击跳转和回看，比如一件女装的“外观样式”标签具体来说是“裙摆设计”，那么直接透出“裙摆设计”这一细粒度标签就比“外观样式”这一固定标签更具信息量，也更能吸引用户。基于此，我们在类别粒度的打标基础上，升级了细粒度标签，作为前端跳转的直达标签。类别标签和细粒度标签的对比如下：

视频片段对应的口播	类别标签	细粒度标签
拍2的话是领取我们**20元的券**。然后点击**领券购买**。	**福利优惠**	**优惠券**
他们家一般是有**两百多的指导价**，是不是页面现在是234。 首先洗面奶**能保湿**，任何肤质都可以用，就是给它保湿。	**福利优惠、功能介绍**	**心动好价，保湿功效**
这款产品的话，它里面擦在马上**水润**。可以帮你**美白**	**效果介绍，功能介绍**	**水润效果，美白功效**

3.6 视频素材投放优选 –– 选择最佳剪辑结果

视频剪辑制作完成后，需要不断地在线上广告系统中验证和迭代，剪辑后视频的内容差异化较大，投放效果也会有所波动。比如同一个商家的多个商品中，一部分商品剪辑后投放效果稳定提升，另部分商品投放效果波动或有下跌趋势。说明同类目下的商品的讲解内容仍然存在多样和差异性的需求。为了解决这个问题，我们在行业剧本剪辑的基础上丰富剪辑的多样性，将行业剧本进行扩充，每个看点剪辑出多个视频，并根据投放效果进行优选，投放效果最好的剪辑结果。

3.6.1 优选逻辑

1）根据行业剧本分析结果，设定多个不同的视频剪辑剧本，作为不同的投放版本

2）每个看点制作多个视频，并进行投放试探

- 在制作看点高光视频时，每个看点剪辑多个视频结果，并以多版本的方式绑定到对应的看点下。在试探投放时，多版本随机投放。

3）根据试探投放的效果，确定稳定的投放版本，实现商品粒度的制作优选

- 投放时按照商品粒度优选看点，选择试探投放效果最好的版本作为优选看点的投放版本。



素材优选

4. 总结

直播视频提供了丰富的商品介绍内容，我们基于对直播内容的分析和加工提升了创意和内容的制作效果，提高了广告投放效率。在本文中我们介绍了在直播视频的剪辑流程和投放优化方面做的一些探索，基于对直播内容的分析和加工提升了创意制作能力；通过对直播内容剪辑和投放，实现实时为商家提供高质量看点视频投放方案。当然，未来对直播内容的理解和利用还有很多的方向有待探索，我们也将持续在内容制作和投放领域深耕。

关于我们

我们是阿里妈妈内容平台与智能创作团队，专注于图片、视频、文案等各种形式创意的智能制作与投放，以及短视频广告多渠道投放，产品覆盖阿里妈妈内外多条业务线，欢迎各业务方关注与业务合作。同时，真诚欢迎具备 CV 和 NLP 背景的同学加入！

投递简历邮箱：alimama_chuangyi@service.alibaba.com

视频分割新范式：视频感兴趣物体实例分割 VOIS

作者：柴风

1. 背景

视频中物体分割是视频理解的基础算法，也是对淘宝商品视频分析和加工所依赖的重要能力。传统的视频分割任务一般分为两种类型：一种是 VOS（Video Object Segmentation），该任务需要在第一帧给出物体的初始分割标注，并在此基础上对视频后续帧中的标定物体进行跟踪和分割；另一种是 VIS（Video Instance Segmentation），这个任务目标是在预定的物体类别范围内，实现物体的检测、分类、跟踪和分割。

VOS 需要给定第一帧标注，在实际应用中可行性低，因为视频的第一帧可能不包含需要分割的商品，另外在批量处理场景也难以通过交互给出物体的分割区域；VIS 方案需要预先定义物体的类别范围，但对于淘宝平台而言，商品种类和样式繁多，且新品增速快，无法预先确定需要分割的物体类别集合。

为了实现从视频中分割任意物体，我们提出了一个新任务：视频感兴趣物体实例分割（VOIS，Video Object of Interest Segmentation），给定视频和目标物体图像，从视频中检测、跟踪并分割出目标物体。同时我们设计了一种基于双路 Transformer 融合图像和视频特征的方案，实现给定任意视频和感兴趣图像对，从视频中跟踪并分割出给定的物体。基于该工作论文已发表在 AAAI 2023，欢迎阅读交流。

论 文： Video Object of Interest Segmentation

下 载（点击↓阅读原文）： https://arxiv.org/abs/2212.02871

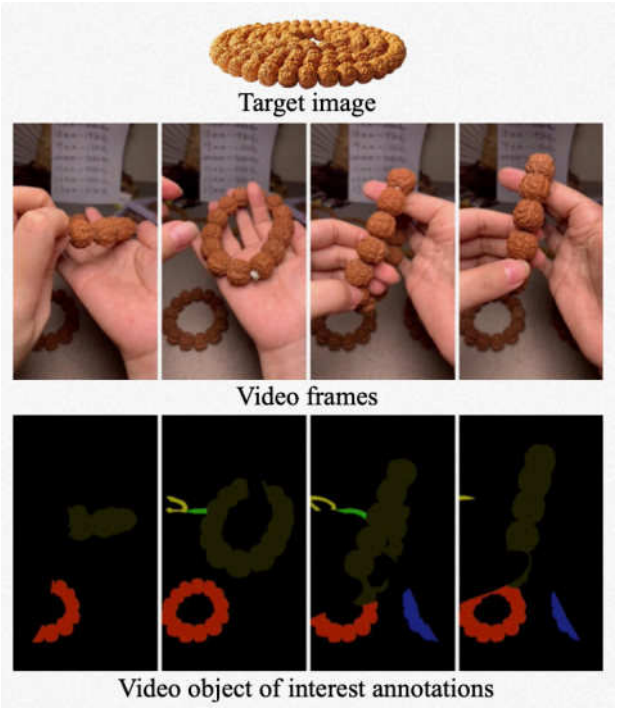
2. 任务介绍 & 数据集

2.1 VOIS 任务介绍

任务定义：给定一个视频和感兴趣物体的图像，从视频中分割出与感兴趣物体相关的实例。

相关物体（实例）：是指视频中的物体在样式、类别和颜色等方面都与图片中的物体一致。物体存在一定的形变、角度变化等情况，仍被认为是相关物体。

分割目标：需对所有的相关物体实例实现跟踪和分割，如果存在多个相关物体，需要能够单独区分实例。



从上到下依次为商品图、原视频帧、视频帧的分割结果（不同颜色代表不同实例）

2.2 数据集

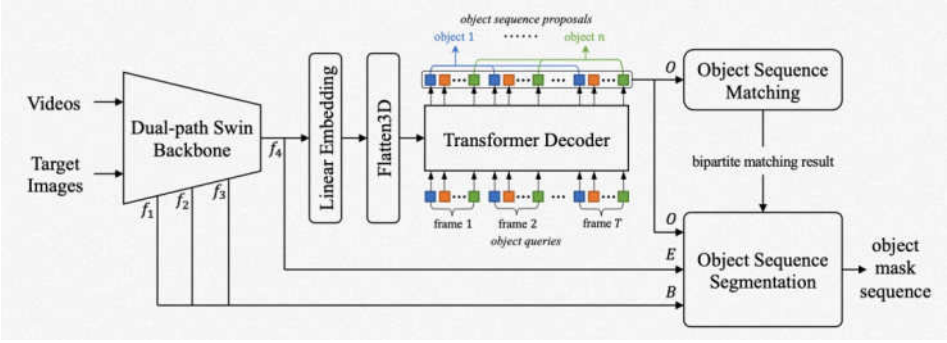
数据集构建

对于 VOIS 任务，目前没有与之匹配的数据集，因此我们重新构建了一个视频图像对组成的视频实例分割数据集。数据集中的视频来源于淘宝直播场景，图片来源于淘宝商品白底图，标注方式为人工标注。数据集中共包含 2418 个视频片段和商品图像样本对。其中，视频 2003 个，目标商品图像 2418 个，共包含 3341 个目标物体，11.4 万个掩码图（视频和图的掩码数量总和）。每个视频长度在 5 秒 ~ 7.2 秒之间，同时在数据构建时保证视频中有目标物体出现。由于视频的来源为淘宝直播，我们将数据集命名为 LiveVideos，LiveVideos 与常用视频分割数据集对比的情况见下表：

Dataset	Video clips	Categories	Objects	Masks	Exhaustive
FBMS (Ochs, Malik, and Brox 2013)	59	16	139	1.5k	✗
YouTubeObjects (Jain and Grauman 2014)	96	10	96	1.7k	✗
DAVIS2016 (Perazzi et al. 2016)	50	-	50	3.4k	✗
DAVIS2017 (Pont-Tuset et al. 2017)	90	-	205	13.5k	✗
YouTubeVOS (Xu et al. 2018)	4453	94	7755	197k	✗
YouTubeVIS (Yang, Fan, and Xu 2019)	2883	40	4883	131k	✓
LiveVideos	2418	-	3341	114k	✓

另外，这个数据集也可以作为基础数据集支持其他视频分析相关任务使用，比如视频检索（Video Retrieval），视频精彩片段判断（Video Highlight）等。**评价指标** VOIS 的目标与 VIS 任务类似，均需从视频中检测、跟踪并分割目标物体，因此，我们应用 VIS 任务中使用的平均准确率（AP，Average Precision ）和平均召回率（AR，Average Recall）指标来对 VOIS 任务的效果进行评估。

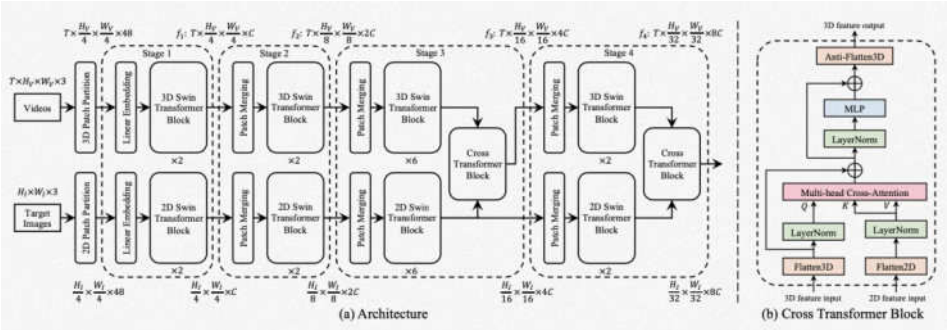
3. 方案介绍



整体方案流程图为了实现感兴趣物体实例分割，我们提出了一种 Encoder-Decoder 结构，包括对图片和视频编码、特征解码和目标物体检测、掩码分割几个部分，主要流程如下：

3.1 特征提取

由于 VOIS 任务需要同时处理一个视频和一个图像，我们使用一个双路 Transformer 结构来提取视频和图像特征。考虑到效果和时效性，我们采用了 Swin Transformer 作为特征提取结构，Swin Transformer 的特征提取包括 4 个阶段（Stage），每个阶段对空间维度进行下采样，实现特征提取。



Swin Transformer 结构和 Cross Transformer 结构图

3.2 特征融合

为了将图片和视频的特征结合在一起，我们同时将两个特征输入到 Cross Transformer 模块，生成融合特征。Swin Transformer 的特征提取模块包含 4 个阶段，我们将 Cross Transformer 模块添加到第 3 和第 4 阶段，以融合更高阶的模型特征，Cross Transformer 采用常用的 Multi-head Cross-Attention 结构。

3.3 实例生成

受到 DETR 的启发，我们使用一个 Transformer Decoder 从融合特征中生成物体的候选集合。在融合特征进入到 Transformer Decoder 之前，我们利用 Embedding 层实现对特征维度的匹配。经 Decoder 之后，我们从视频中的每一帧中解码出 n 个物体，n 为预定义的超参数。

3.4 物体匹配

在物体匹配过程，我们利用二部图匹配损失（Bipartite Matching Loss）训练匹配模块，将预测实例和标注（Ground Truth）匹配。在进行了二部图匹配之后，候选物体与目标物体之间将具有最优的匹配方案，也就是最短距离。根据匹配结果，即可找到视频中的感兴趣物体的相关实例。

3.5 视频分割

视频分割环节，使用视频序列分割模块为每个候选物体生成分割结果。我们利用匈牙利损失（Hungarian Loss）实现分割模块的训练，匈牙利损失主要包括：分类、包围框回归和分割三个模块，其中分类模块输出代表候选物体的置信度，包围框回归和分

割模块的输出分别代表物体的包围框和分割结果。

4. 实验

4.1 Baseline 搭建

由于现有的视频分割方案（如 VOS 和 VIS 任务的解决方案）跟 VOIS 设定存在差异，我们在任务定义的时候额外给定了一张输入图片，因此现有的视频分割方法无法直接与之对比。为了实现合理的方案对照，我们在现有视频分割方案的基础上，增加新的图像编码分支，复现不同方案在 VOIS 数据上的效果。我们基于 MaskTrack R-CNN 和 VisTR 两个实例分割方案实现对比 Baseline。

MaskTrack R-CNN 模型

我们额外增加一条 ResNet 分支作为图像特征提取 Backbone，然后使用 Cross Transformer 融合图像和视频两种特征，使用双路输入的特征提取和特征融合模块替代原始 Backbone。

VisTR 模型

VisTR 采用了 ResNet 作为特征提取 Backbone，我们采取跟改造 MaskTrack R-CNN 类似的方式提取图像特征，用融合特征替换原始 Backbone。由于 VisTR 包含 Transformer 特征处理模块，我们将 VisTR 模型里的 Transformer 层改为 Cross Transformer，作为视频、图像特征的融合，最后将融合特征输出给 Decoder 模块。

4.2 对比实验

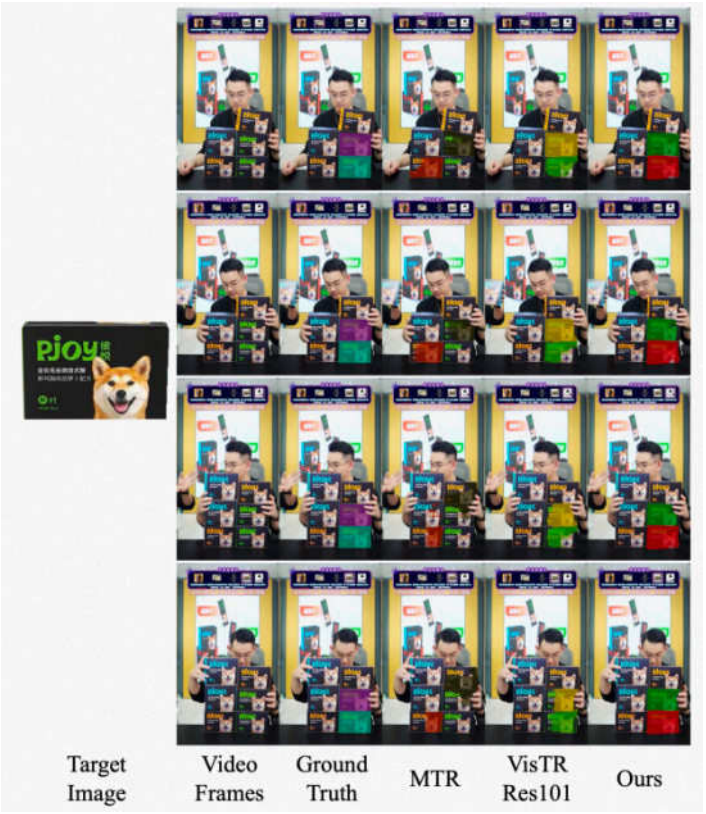
实验数据

在完成 Baseline 模型的适配后，我们实验对比了 MaskTrack R-CNN 、VisTR 和我们提出方案的视频分割效果。实验结果可以看出，我们的双路 Swin Transformer 方案在平均准确率（AP）和平均召回率（AR）指标上均优于两个 Baseline。

Method	Backbone	AP	AP_{50}	AP_{75}	AR_1	AR_{10}
MaskTrack R-CNN (Yang, Fan, and Xu 2019)	ResNet-50 (He et al. 2016)	29.0	46.4	32.1	32.5	35.5
VisTR (Wang et al. 2021b)	ResNet-50 (He et al. 2016)	34.9	54.1	37.0	38.3	41.8
VisTR (Wang et al. 2021b)	ResNet-101 (He et al. 2016)	37.0	56.4	39.2	38.1	43.7
Ours	Dual-path Swin Transformer	38.8	61.6	41.8	41.2	46.6

不同方法实验指标对比

不同方案效果对比示例



不同方法分割效果对比图

在上图中，左侧给定的是目标物体的图像，右侧的第一列为原始视频帧，右侧第二列为标注结果，右侧的后三列是不同方法的分割结果。在分割结果中，不同颜色代表不同的实例。由于给定的商品（物体）可能包含不同的包装样式，视频中包含很多相似的物品，准确地找出给定物体存在一定难度。从分割结果上可以看出，我们提出的方案在物体识别能力和目标分割准确度上均优于其他方法。

4.3 消融实验

验证目标图像的作用

在 VOIS 任务定义和模型构建过程时，我们在数据集中提供并在模型中使用了感兴趣的目标图像，我们期望提供的感兴趣图像能够引导模型从视频中找到正确的物体，在此我们验证了给定图像对分割效果的影响。具体来讲，我们在模型中删除图像特征提

取分支，只保留视频特征提取分支，与双路输入的模型对比效果。我们发现在去除目标图像分支后，模型的 AP 和 AR1 分别下跌了 12.1 和 10.2。由此验证在 VOIS 任务中，给定图像特征直接影响视频分割的效果，且图像在模型推理过程中可以正确地引导视频分割目标物体。

Target image path	AP	AP_{50}	AP_{75}	AR_1	AR_{10}
	26.7	42.9	28.1	31.0	38.0
✓	38.8	61.6	41.8	41.2	46.6

是否包含图像分支的对比结果

选择最优模型结构

在特征融合时，我们在 Swin Transformer 的第 3、4 层添加了 Cross Transformer 结构，原因为网络初始层只包含低层模型特征，同时前两层的特征图较大，使得 Cross Transformer 的计算占用过多显存空间难以计算。因此我们主要关注在第 3、4 阶段（Stage）验证添加 Cross Transformer 的效果。只在第 3 阶段或第 4 阶段添加 Cross Transformer，与两层均添加相比 AP 分别降低 1.1 和 1.3，AR1 分别降低 0.7 和 0.3。由此可见，两次特征融合可以更好地匹配视频和图像特征。

Stage 3	Stage 4	AP	AP_{50}	AP_{75}	AR_1	AR_{10}
✓		37.5	60.3	41.0	39.9	44.6
	✓	37.7	59.8	41.4	39.5	45.5
✓	✓	38.8	61.6	41.8	41.2	46.6

模型结构对比结果

5. 总结和展望

为了突破传统视频分割算法的局限性，我们提出了一种应用场景更加广泛的视频分割范式 VOIS：根据提供的视频和目标物体图像，对视频中的目标物体进行实例分割。同时，我们提出了一种有效解决 VOIS 任务的模型，该模型可以学习到一种通用的、对视频和图像特征进行提取和匹配的能力，从而能够有效处理任意给定的视频和图像对，满足我们面临的商品样式多、新品增速快的海量视频分析场景。然而，目前方案仍有一定的扩展空间，比如：给定的图像中包含多个物体时，如何在视频中对不同的物体实现多类别的实例分割等，未来我们也将持续在相关方向上探索。

6. 关于我们

我们是阿里妈妈创意 & 内容算法团队，致力于推动广告创意和内容投放产业的 AI 升级，努力推动创意制作、理解、模型预估和广告投放的全栈智能化。得益于阿里巴巴庞大而真实的营销场景，团队在图像技术、视频技术、文案生成、广告投放等领域持续发力和创新，现已构建出图片与短视频创意自动生成，创意个性化投放，智能文案写作，全自动与交互式抠图等特色产品，论文发表于 CVPR、ICCV、AAAI、ACMMM、WWW、EMNLP、CIKM、ICASSP 等领域知名会议。用 AI 赋能现代营销，驱动产业升级。真诚欢迎 CV、NLP 和推荐系统相关领域的同学加入！

投递简历邮箱：

alimama_chuangyi@service.alibaba.com

7. 参考文献

[1] Liu, Z.; Lin, Y.; Cao, Y.; Hu, H.; Wei, Y.; Zhang, Z.; Lin, S.; and Guo, B. 2021. Swin transformer: Hierarchical vision transformer using shifted windows. In ICCV.

[2] Liu, Z.; Ning, J.; Cao, Y.; Wei, Y.; Zhang, Z.; Lin, S.; and Hu, H. 2022. Video swin transformer. In CVPR.

[3] Wang, Y.; Xu, Z.; Wang, X.; Shen, C.; Cheng, B.; Shen, H.; and Xia, H. 2021. End-to-end video instance segmentation with transformers. In CVPR.

[4] Ge, W.; Lu, X.; and Shen, J. 2021. Video object segmentation using global and instance embedding learning. In CVPR.

[5] Voigtlaender, P.; Chai, Y.; Schroff, F.; Adam, H.; Leibe, B.; and Chen, L.C. 2019. Feelvos: Fast end-to-end embedding learning for video object segmentation. In CVPR.

[6] Yang, L.; Fan, Y.; and Xu, N. 2019. Video instance segmentation. In ICCV.

风控技术

阿里妈妈内容风控模型预估引擎的探索和建设

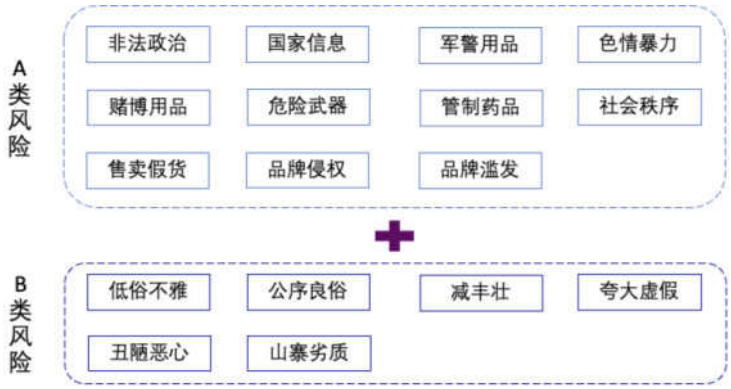
本文作者：徐雄飞、金禄旸、滑庆波、李治

内容作为营销的重要载体，能够促进信息的交流和传播。在营销场景中，广告高曝光的特性放大了风险外漏带来的一系列问题，因此对内容的风控审核就显得至关重要。本文将为大家分享阿里妈妈内容风控模型预估引擎的探索和建设。

1. 业务背景及问题

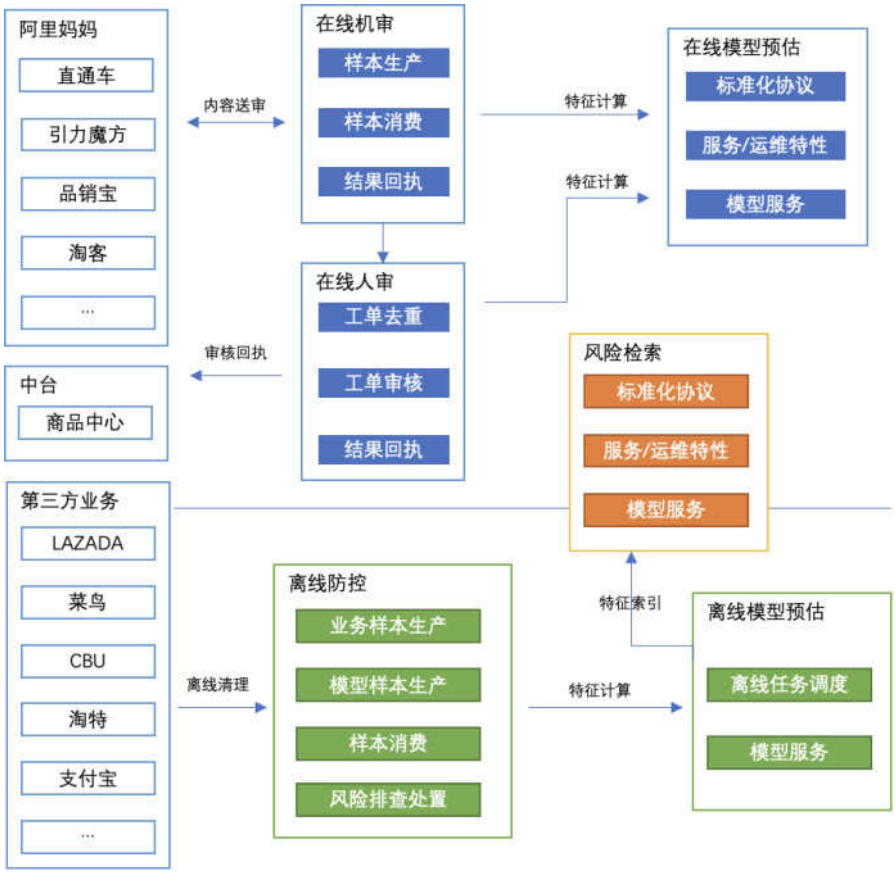
1.1 业务背景

内容作为营销的重要载体，能够促进信息的交流和传播。在营销场景，广告内容需要满足广告法要求，常见的风险类型有两类：一类是面底线的 A 类风险，如政治等，一类是面向平台调性，广告法要求的 B 类风险，如低俗不雅、公序良俗等。一旦风险外漏，由于广告高曝光的特性，会导致舆论压力，甚至有业务关停的风险。



阿里妈妈内容风控核心由机审、人审两部分组成，其中机审防控包括增量和全量防控（链路如下图所示），机审与人审环节都依赖模型服务，由于广告数据和主站存在差异，且风控的强对抗性、模型都自研实现。随着模型不断增多，模型的线上服务部署和管

理变的越来越臃肿，出现诸多问题。



1.2 现存问题

现有的模型预估引擎（Inference-kgb）由于历史业务演进、临时性支持妥协，随着模型数量不断增加问题逐渐暴露出来，下面从能力、效率、质量、成本四个方面详细分析现有的问题，为我们重构新的引擎打下基础。

1.2.1 能力问题

缺乏统一的加速方案：仅部分模型通过 GPU 进行加速，大部分模型性能较差；异构混部模型缺乏通用的接入、导流、蓄洪和限流等服务特性和运维特性。

一致性问题：离线和在线模型结果一致性难以保障，没有形成通用方案；跨平台模型 Case by Case 上线，没有统一的服务 API；运维干预纯靠人工经验，无可靠的管控 API。

1.2.2 效率问题

研发周期长：Inference-kgb 框架底层采用 C++ 实现，算法同学上线前需要将训练的 Python 代码翻译成 C++ 代码，开发周期长且部分库无法直接翻译需要手写，会导致实验结果和线上不一致。

部署上线流程复杂：每个模型上线都需要代码开发，像分类模型代码及配置都相同只更换模型文件，由于框架限制需要走完开发、测试、部署、上线全流程。过去模型部署链路在 / 近 / 离线是分别管理的，模型上线过程中需要跨多个平台进行配置，操作繁琐。每次模型上线需要通过邮件申请 Metaq（阿里内部消息中间件）消息，等 Metaq 同学回复邮件后才能够继续发布流程；研发流程无法保障多人协作安全、高效。

1.2.3 质量问题

缺乏统一的接入标准：早期整个框架是完全自研实现，可参考的方案较少，不断演进之后与业界方案存在差异。如缺乏统一的 Tensor in Tensor out 模型层面无业务语义的接口，用户接口设计不合理，接口核心字段为 imageUrl 和 textString 导致很多字段都通过拼接的方式传入 textString 中；输入非结构化存在诸多问题，如离线冷启动执行模型时，需要手动按照线上逻辑进行字段拼接。

子模型问题：将多个通用分类、检测模型部署在一个服务中，然而服务及运维属性都是按照模型粒度设置的，无法根据子模型定制：如 Cache、计算资源等是针对整个服务维度生效的，无法根据子模型进行精细化控制。

同质代码缺乏复用：如检索类模型提取向量特征之后需要进行检索库匹配操作，过去的服务框架将特征提取和检索融合在一起，由于检索库的差异会导致部署模型数量膨胀。

1.2.4 成本问题

GPU 利用率低下：部分大流量的模型之前没有 GPU 版本，GPU 模型普遍使用率比较低。离线无法充分利用资源提取特征，由于模型缺乏 GPU 镜像，导致离线提取特征是无法充分利用智能引擎 (MDL) 资源池、主搜 (Hippo 资源调度) 资源池的离线 GPU 资源。

2. 业界对比选型

基于以上问题，我们考虑对现有模型预估系统（Inference-kgb）进行重构，优先考

虑复用集团或开源现有的能力，因此我们对集团及开源方案进行了调研。

2.1 业界方案对比

分别调研阿里云 EAS、达摩院 Aquila、CRO 灵境、阿里妈妈 HighService、开源 TritonServer。

系统	EAS	Aquila	灵镜	HighService	TritonServer
开发语言	C++ Python Java	C++ Python	C++ Python	Python	C++ Python
支持量化优化	是	是	是	否	是
支持Batching优化	是	是	是	是	是
支持剪枝优化	是	是	是	否	是
支持模型类型	PyTorch Tensorflow Caffe PMML XGBoost TensorRT等	PyTorch Tensorflow OnnxRuntime HIE TensorRT 等	PyTorch Tensorflow OnnxRuntime Mxnet Paddle 等	PyTorch Tensorflow TensorRT 等	PyTorch Tensorflow OnnxRuntime TensorRT 等
支持Tensor接口	是	是	是	是	是
支持Graph编排	是	是	是	否	是
平台化支持	是	是	是	否	是
支持SDK本地对接	是	是	否	是	否
目前使用范围	集团	集团	CRO	阿里妈妈	NVIDIA开源
核心优势	在集团广泛推广，和PAI、Blade加速生态打通，支持HTTP及SDK多种对接方式；跨平台支持Aquail、Triton部署，是目前集团使用最广泛的平台	解决Python>视频流处理、进程内/进程间推理	从调度、部署、编排、加速、服务化、上云整体生态完善	解Python性能问题，BU内部紧密合作，提供多种加速方案	定义业界标准解决方案

系统	EAS	Aquila	灵镜	HighService	TritonServer
支持弹性伸缩	是	是	是	是	是
模型存储方案	OSS	OSS	OSS	OSS	无
对接的训练平台	PAI	PAI	Quake	MDL	–
资源调度	K8S	K8S	K8S	K8S	K8S
支持私有化部署	是	是	否	否	是
支持云上部署	是	是	是	否	是
支持音视频解码能力	是	是	是	是	是

2.2 业务诉求

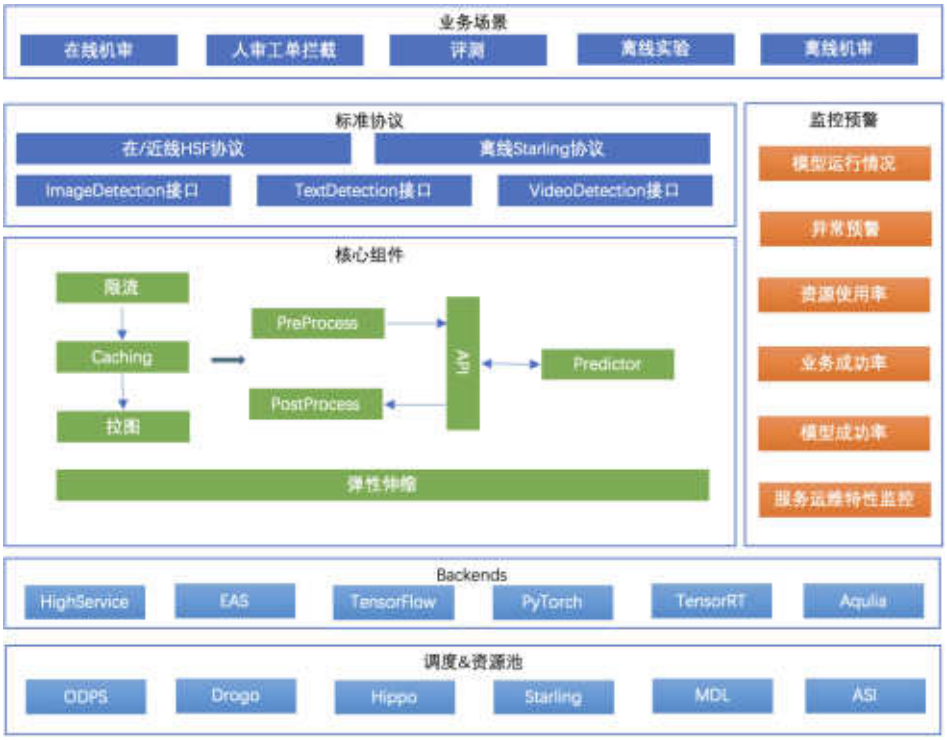
以上平台 / 框架，有些包含完整的模型部署、运维功能，有些只包含模型加速功能。我们尝试推演将模型整体部署到平台上，遇到以下几个问题：

- 资源互通问题：以上平台（如 EAS），需要在 ASI（Alibaba Serverless infrastructure）上统一申请资源，现有的 Hippo 资源无法直接迁移过去使用。
- 离线计算问题：目前已有平台解决的基本都是在线服务部署的问题，由于风控业务特性，我们遇到风险事件时需要在离线环境进行大批量的特征提取及全量数据的风险防控，在指定时效内完成风险清理，这部分没有现有能力支持。
- 缺乏偏业务语义的功能：针对偏业务语义的功能，平台没有很好的支持，如模型对应的风险管理，版本迭代后的业务效果，模型结果复用等。

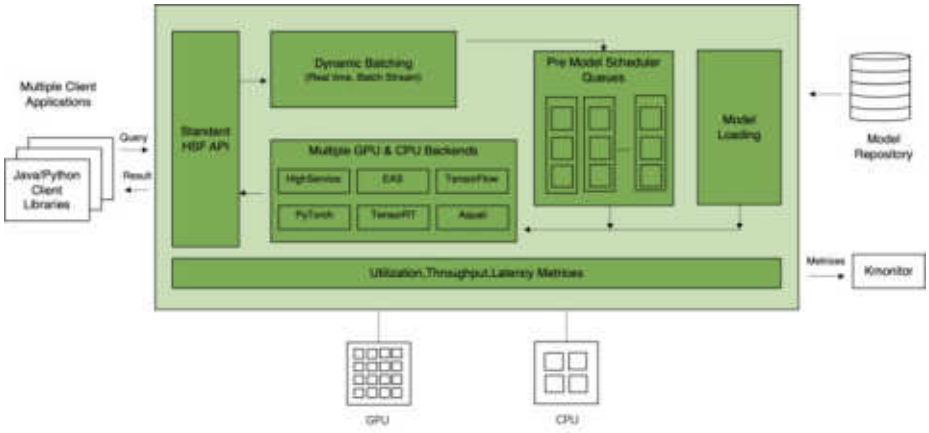
考虑到业务诉求及已有能力最大程度复用，核心构建上层偏业务的服务能力，底层能够接入 EAS、HighService、Aquila 等多种 Backends，进行模型服务和加速。

3. 模型服务框架

风控业务场景，模型服务核心解决机审风险识别及人审工单拦截等相关问题。模型分别服务于在 / 近 / 离线业务，引擎内核复用同一套逻辑，确保在 / 近 / 离线数据一致，底层接入多种 Backends，对模型进行部署、加速。可以在各个调度器及资源池中进行计算。重构后新的风险预估引擎 RD（RiskDetection）层次结构如下图：



RD 内核部分，参考 NIVIDIA Triton Server 实现（NVIDIA 开源推理框架），我们定义了一套标准业务接口，支持多 Backends 对接的在线服务。其中 Standard API 支持 Tensor in Tensor out 标准模型协议，由 Model、Version、Tensor 标准三元组构成，模型内部支持动态 Batching 特性。通过对接集团及开源社区已有的 Backends 低成本的实现模型预估能力。



3.1 标准化接口

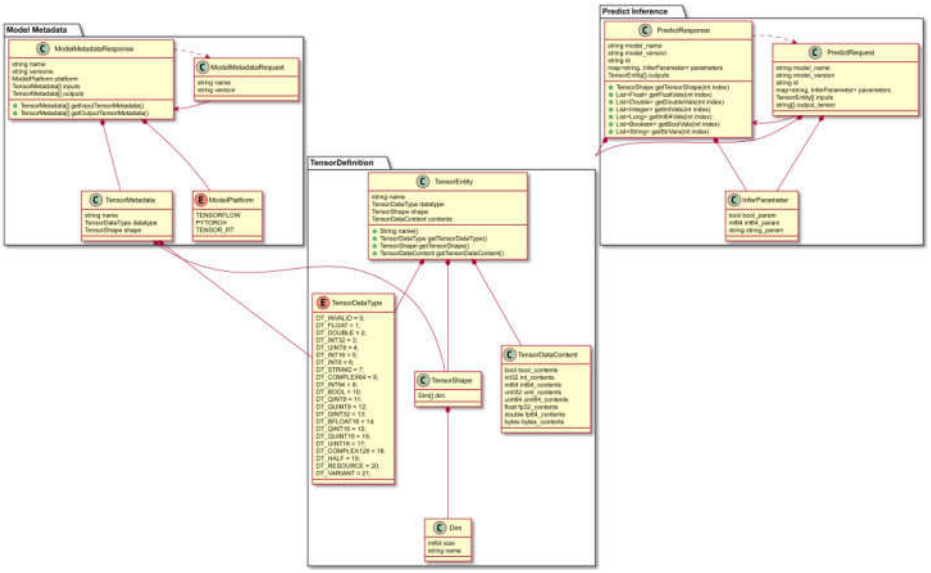
接口定义分为面向业务及面向数据两个维度，面向业务包含图片、文本、视频等内容识别输入输出接口；面向数据面，我们与主流的 KServe（开源云原生模型推理框架）对齐，采用 Tensor in Tensor out 进行模型预估逻辑。

3.1.1 业务接口

业务侧支持图片、文本、视频等素材类型，包含检测、识别、分类、检索等模型服务。过去的系统（Inference-kgb）所有的模型都共用一个接口，输入是图片、文字，输出是 Map 对应的 K,V。系统运行一段时间之后，发现输入、输出变的越来越复杂，于是在字段上进行各种拼凑，导致输入、输出变的越来越复杂。吸取 Inference-kgb 的经验教训，也参考阿里云等开源接口设计，我们针对不同类别的模型进行独立接口设计。

3.1.2 数据面接口

在模型内部，都是以 Tensor 数据格式进行传输，我们支持调用方通过 Tensor 接口调用模型服务，通过 TensorFlow、KServe 以及内部 EAS 对 TensorFlow/PyTorch 模型的服务接口抽象，定义我们自己需要的 TensorFlow/PyTorch 模型服务标准 API 如下图：



3.1.2.1 Predict 接口

1）TensorDataType：TensorDataType 用于标准化指定 Tensor 的数据类型。

2）TensorShape：用于标准化指定 Tensor 的维度。

3）TensorDataContent：内部的 EAS 和外部开源的 TensorFlow 其实都没有 TensorDataContent 这个抽象，都使用的完全平铺的方式定义 Tensor 内容，KServe 最新版本对 TensorData 做了抽象，是能够包含 EAS 和 TensorFlow 的定义的，因而采用。

4）InferParameter：InferParameter 这个数据结构，内部的 EAS 和开源的 TensorFlow 都没有定义，但是 KServe 最新版本的 API 定义里预留出来了，目的是未来能够基于标准化的 Tensor 定义支持跨多个不同的非标准化推理服务平台后端，通过拓展参数增强跨平台的可移植性。考虑到内容风控未来线上模型也会对接多个非标准化 Backend，所以考虑在前面统一的 API Protocol 标准化 Tensor 结构定义的基础上，也引入 InferParameter 以实现跨平台的对接能力。

5）TensorEntity：把 TensorDataType、TensorShape、TensorDataContent 和 InferParameter 组合，得到一个标准化定义的 TensorEntity 数据结构。

6）PredictRequest：如前面所述，我们把原子的 Tensor in Tensor out 的推理服务命名为 Predictor，PredictRequest 则是每个 Predictor 角色的标准化输入接口入参，这里提取 EAS、TensorFlow 和 KServe 各自有用的部分组合而成。

7）PredictResponse：PredictResponse 核心返回数据结构是 TensorEntity，同时把 PredictRequest 原始请求的模型信息、拓展参数信息也带回来（参考 KServe 设计，原生 TensorFlow 和 EAS 没有）。

3.1.2.2 Metadata 接口

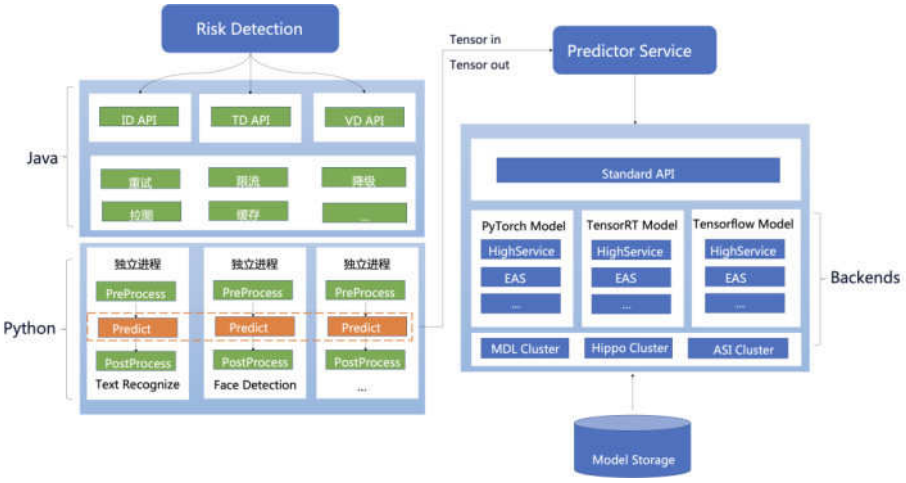
1）TensorMetadata：Tensor 的元数据由 Tensor 名、Tensor 数据类型和 Tensor 维度组成的三元组表示。

2）ModelPlatform：后端支持的模型框架平台类型，目前支持原生的 Tensorflow、PyTorch、TensorRT 三类，以及集团内部 HighService、EAS、Aquail 等。

3）ModelMetadataRequest：ModelMetadataRequest 是用于请求模型推理服务元信息的标准化抽象接口，由模型名 (name) + 模型版本 (version) 构成二元组。

4）ModelMetadataResponse：根据 ModelMetadataRequest 指定的模型名 + 模型版本二元组获取详细的模型推理服务信息，主要由传入的模型名、模型版本、输入 Tensor 元数据和输入 Tensor 元数据这四部分组成。

3.2 RD 内核技术方案



RiskDetection：提供标准的模型用户接口，提供模型服务特性和运维特性，串联模型前处理、模型预估、模型后处理阶段。

Predictor：提供模型标准数据面 Tensor in Tensor out 接口，兼容多种 RTP 协议的 Backends。Predictor 逻辑上是拆分成独立的服务，物理实现为了减少 I/O 请求，可以考虑和前后处理部署在同一台机器上。

Transformer：Transformer 用于模型服务的前后置处理，它和 Predictor 共同完成一个完整的模型推理请求。Transformer 一方面可以通过拓展前处理逻辑把外部输入转换成 Predictor 接收的标准输入格式，另一方面可通过拓展后处理逻辑把 Predictor 返回的结果做进一步转换供给业务方应用。

Backends：底层具体采用的模型服务框架，包含集团及开源解决方案。

3.3 数据一致性保障

风控业务包括了在 / 近 / 离线场景，不同场景对资源、时效、吞吐等要求不一样，各场景提取出来的特征需要保障最终一致。对一致性保障可以划分成两个层级：特征完全一致、业务效果一致。

1）特征完全一致：高保障业务场景，对风险判断要求更严格，我们需要确保模型输出 Tesnor 保持一致，因此在 / 近 / 离线需要保证镜像、模型文件版本、计算资源完全一致。

2）业务效果一致：在一些保证级别要求不高，且计算量大的场景，我们允许在 / 近 / 离线计算资源不一致，如在线采用 GPU，离线采用 CPU（GPU 通过 TensorRT 半精度加速后结果有损），只需要在业务效果上保障一致即可。

4. 模型推理加速

过去的 Inference-kgb 基于原生 TensorRT 进行模型加速，随着模型迭代更新越来越频繁，每次算法同学训练模型到上线都需要将 Python 代码用 C++ 重新实现一遍。通过对集团及开源方案调研，现有的框架很多都支持 Python 直接进行模型部署、加速，能够解决 Python 语言带来的 GIL 锁冲突问题，新的框架我们采用 Python 进行模型预估开发，并且能够支持多种模型加速 Backends(如 HighService,EAS,Aquila 等)，可以直接复用已有能力，无需从 0 开始重新搭建模型推理加速功能。由于整体框架初见雏形，我们先接入了 HighService 和 EAS 两个 Backends，支持我们核心模型的落地。

4.1 HighService Backend 接入

HighService 是阿里妈妈异构计算团队内部开发提供的，基于 Python 多并发异构计算服务框架。HighService 框架通过解耦 GPU 及 CPU 计算，使用多进程进行 CPU 计算，充分利用多核 CPU 资源，在多个 CPU 进程与单个 GPU 进程之间通信，避免 GPU 进程饥饿，同时 CPU 进程数不受 GPU 大小内存限制。内部集成了 TensorRT 加速功能，成倍地增加了 PyTorch 模型的计算能力。使用 HighService 框架服务的动机，除了它的高性能以外，还有一个重要原因，就是利用 Python 模型易于迭代、维护、定位问题的优势，可以更好地服务业务，解决了业务方研发周期长的痛点。相比于过去的 Inference-kgb 框架，利用 HighService 的 RD 框架上线一个 GPU 模型的研发周期得到大幅度提升。

4.2 EAS Backend 接入

EAS 广泛在集团使用，它与 PAI 平台（模型训练平台）无缝衔接，并且底层拥有完善的 Blade 加速方案，服务和运维特性非常全面。在与 EAS 同学深入沟通后，发

现由于计算资源池及离线业务特殊性等原因，无法直接将服务完全托管。但 EAS 提供的 SDK 能够给支持模型的部署、加速能力，这部分能力可以直接接入 RD 作为 Backend，接入后基于镜像部署可以同时支持在 / 近 / 离线的业务。

EAS 主要的接入方式有两种：Processor 方式接入和 Mediaflow 方式接入，Processor 方式需要将模型部署在 EAS 平台上，能够方便的部署模型，并且提供标准化的 Tensor 接口；由于我们无法直接将模型部署到 EAS 平台，因此我们采用 Mediaflow 的方式接入。具体实现 demo 如下：

```
# pipeline 的方式进行构建
with graph.as_default():
mediaflow.MediaData() \
        .map(tensorflow_op.tensorflow, args=cfg) \
        .output("output")
# 调用方式
results = engine.run(data_frame, ctx, graph)
```

Mediaflow 方式是 EAS 团队今年新研发的方式，可以将整个模型流程采用 DAG 进行串联（也可以是单个节点），能够轻量级接入 SDK，并且享受到底层 Blade 加速效果。

5. 业务支持及效果

完成 RD 系统重构，我们进行压测及效果验证达标后，开始对接业务。目前 RD 主要服务于阿里妈妈内容风控在 / 近 / 离线模型计算，为内容风控机审风险识别和人审提效服务。在 / 近线都是常驻型服务，拉起后提供 HSF 服务接口，离线是通过批量任务的方式启动。

5.1 在 / 近线业务支持

对比过去的 Inference-kgb 服务拉起配置较为繁琐，需考虑了多个子模型 DAG 之间的串联，采用 OP 算子进行关联。每个模型上线都需要开发 OP，哪怕仅更换模型文件拉起不同的服务。我们将子模型组装 DAG 前置到业务编排组件，RD 就是纯粹的进行特征提取、模型计算，因此在服务拉起及服务运维方面有较大提升。

5.1.1 快速服务拉起

拉起服务主要依赖 镜像 + 模型文件 + 模型配置 三元组，三元组唯一确定一个模型服务，因此快速拉起服务的首要任务就是管理服务依赖的三元组。

- 镜像：RD 所有支持的模型都共用相同的镜像，同时支持 CPU、GPU 模型运行，这样方便整体的管理和运维。统一镜像也会带来一些负面作用，如修改 A 模型可能会影响 B 模型，这个需要我们在开发过程中，修改到公共部分时特别小。需要借助单元测试、集成测试等标准化流程规避此类问题。
- 模型文件：统一管理在 OSS 上，由模型名称 + 版本号唯一确定一个模型文件。可以根据模型文件恢复历史任意的模型服务。该功能为后续做模型效果对比等功能提供底层能力支持。
- 模型配置：吸取过去的 Inference-kgb 复杂模型配置教训，RD 极大的简化配置内容，将配置分为静态和动态两部分。静态模型配置：如模型代码路径，模型 Backend 配置，模型是否需要 Cache 等，这些配置与部署的模型实例无关。如部署 GPU、CPU 版 Face，以上配置都是相同的。动态模型配置：如服务接口、模型版本、资源类型等，每个部署实例有不同的配置。动态配置能够方便的启动相同模型的不同服务版本，能够支持业务进行效果对比或者 CPU、GPU 混部等能力。



如上图，通过以上设计我们能够更方便的针对同一个模型启动各种版本，使用不同类型和保障级别的资源，支持各种隔离级别的业务。

5.1.2 模型服务 / 运维特性

参考集团及开源方案，核心服务 / 运维特性主要包括 Caching、Batching、Scaling，支持以配置化地方式拓展服务 / 运维特性从而拓展系统处理能力，主要特性描述如下：

服务特性名	服务特征增强维度	服务特性作用	对Backend Engine的能力要求
Caching(缓存特性)	纵向-单机	模型推理服务开启结果缓存	1.支持是否开启缓存特性 2.支持配置缓存存储地址 a.OFF_HEAP b.IN_MEMORY c.LOCAL_DISK d.REMOTE_STORAGE 3.支持配置多种缓存淘汰策略 a.基于TTL的淘汰 b.基于Capacity的淘汰 c.基于引用计数的淘汰
Batching(批量特性)	纵向-单机	模型推理服务开启服务端批量化推理能力	1. 后端推理服务支持Batch推理优化 2. 可由客户端指定Batch> 3. 单个请求，多个返回的处理
Scaling(弹性伸缩)	横向-集群	削峰填谷、不同模型之间进行资源相互腾挪充分利用已有资源	1. 支持自动、弹性的对各个模型进行缩扩容，提升资源利用率 2. 尽量复用集团及开源社区已有的能力

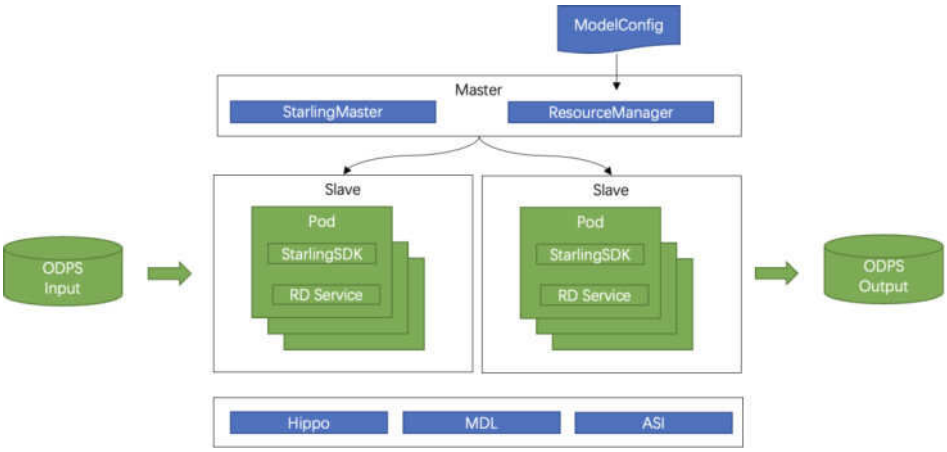
5.2 离线业务支持

通过在线防控，我们能够解决增量风险问题，然而如果有风险外漏，或者监管调整，我们都需要对全量数据进行风险防控。同时算法同学需要在离线进行大量实验、评测，这些都依赖于离线具备大规模模型计算能力，因此 RD 需要具备离线计算的能力。

5.2.1 离线任务对接

Starling 是阿里妈妈风控自研的基于主搜 Drogo（资源调度管理平台）之上进行业务调度平台，底层使用主搜 Hippo 资源池。风控场景在离线需要提取百亿级图片、文本特征进行实验或者风险防控，模型计算需要大量的计算资源，Hippo 资源池拥有海量非保障（离线）资源，Starling 的初衷是可以削峰填谷地使用这些资源支持业务特性。

RD 对接上 Starling 之后，便拥有了离线调度计算的能力，能够灵活的拉起 Hippo 资源，并与 ODPS（对外产品为 MaxCompute）打通，对 ODPS 数据进行读写。同时能够通过 ODPS 进行任务调度，进行定时增量数据特征提取。



如上图为 RD 与 Starling 对接的架构图，Starling 的 Master 节点管理和调度 RD 进行模型计算，和模型配置文件分发，Starling SDK 实现 ODPS 对接及 Slave 节点状态上报。

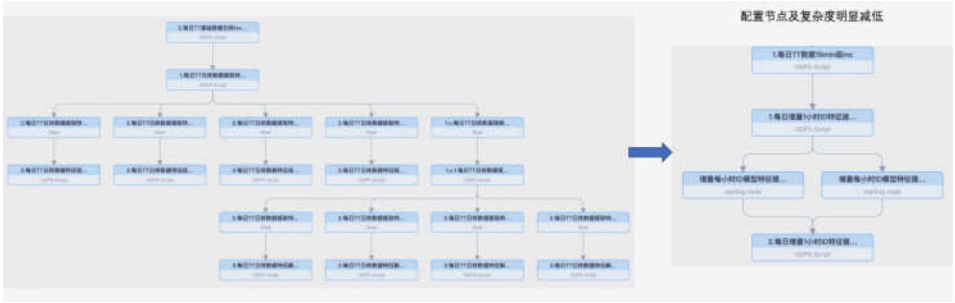


上图为通过 ODPS 配置 Starling 任务节点进行调度，实现定时特征提取功能。

5.2.2 单容器多实例部署

对于模型计算而言，离线与在线存在的核心差异是数据源及消费方式，在线是流式源源不断输入，离线则是固定数据集运行特定的模型。一个任务计算量是亿甚至百亿级别，内容场景有很多图片模型。考虑到拉图成本及多模型任务依赖的管理成本，我们通过单容器支持多个模型运行的方式，节约拉图 60% 左右，以及简化任务配置实例。

支持单模型多实例部署的前提是模型之间没有相互冲突，所以 CPU 资源上更容易实现，GPU 资源由于 16G 显存的限制，很容易 OOM。CPU 也需要根据模型大小、内存申请等控制实例部署个数，例如 30G 内存部署 3 个模型。以下是合并前和合并后任务节点的配置，可以发现合并之后节点明显简洁得多。



5.3 业务效果

RD 上线在 / 近线迎来首个双十一考验，离线需解决大规模特征提取问题，无论是性能、效率都需要有显著提升，能够给业务带来可感知变化，接下来分别介绍在 / 近 / 离线因为 RD 上线带来的相关改变。

5.3.1 离线清理支持

离线部署 6 个核心模型支持全量及增量数据提取任务，通过 Starling 使用非保障资源批量提取增量百万级图片特征可在 20 分钟内完成。

5.3.2 双十一业务支持

大促结束后平台要求广告主在规定时间内下线双十一相关内容，因此相当于整体的广告内容这个时刻开始全量更新一次。

RD 目前对外提供的接口是同步请求方式，QPS 超过模型服务单机承载能力（CPU/GPU Load 打满）时，会造成模型服务 RT 成倍增长影响业务稳定，因此需要对每个模型进行精确限流。内容风控模型服务众多，从模型部署角度上讲：同一个模型可能跑在 GPU 或者 CPU 上，也有可能部署在不同机房（机房拉图带宽有上限，需要多机房部署），不同集群之间有 Backup 的关系（比如 GPU 集群被限流则自动将流量导入到 CPU 集群），从业务角度上讲：对于高优先级场景中的底线类风险需要百毫秒级以内得到模型结果，而一些低优先级的体验类风险则可以使用一些便宜的机器进行计算，能够承受一定 RT 延迟。仅依赖 HSF（阿里 RPC 框架）的限流配置或 Sentinal 限流（阿里分布式限流框架）组件远不能满足这种异构部署和业务上多样化的导流需求，因此我们自建了 InferenceProxy 服务来支持基于业务标签的 QoS 和导流能力，在保护下游 RD 服务能力的同时，对物理层面的调用关系进行配置化的编排。

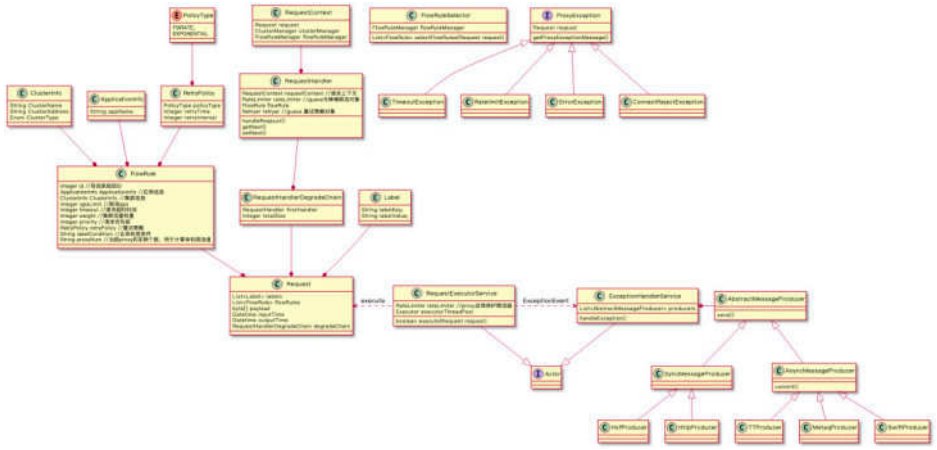
导流表其中几项配置如下：

application	timeout	weight	priority	clusterType	label
1	100	10	0	http	1,44,1125
2	100	20	1	http	1,44,1125
3	100	40	0	http	7
4	100	50	0	http	1125
5	100	30	0	hsf	1,44
6	200	40	0	hsf	1

由用户自定义的一段逻辑从请求原始 Payload 提取信息并打上标签，例如上面这个请求会打上 label=1125 这个标签。执行时会首先查出符合条件的所有导流项：1、2。按照同优先级 + 权重的方式分配流量，详细来说：其中 1 和 4 的 priority 都是 0，1 的 weight 是 10, 4 的 weight 是 50，则这个请求以 1/6 的概率打在服务 1 上，以 5/6 的概率打在服务 4 上。

如果请求 priority=0 的服务失败或超时，则会打到 priority 等于 1 的第二项服务 2 上，仍旧超时或失败，则该请求会进入离线兜底队列。参考导流表设计，一个请求到来时，会筛选出符合条件的几个导流策略，按照 priority 进行排序分层，各 priority 之间使用责任链模式串联，如果请求失败或超时，则进入责任链的下一个策略 priority 进行执行。

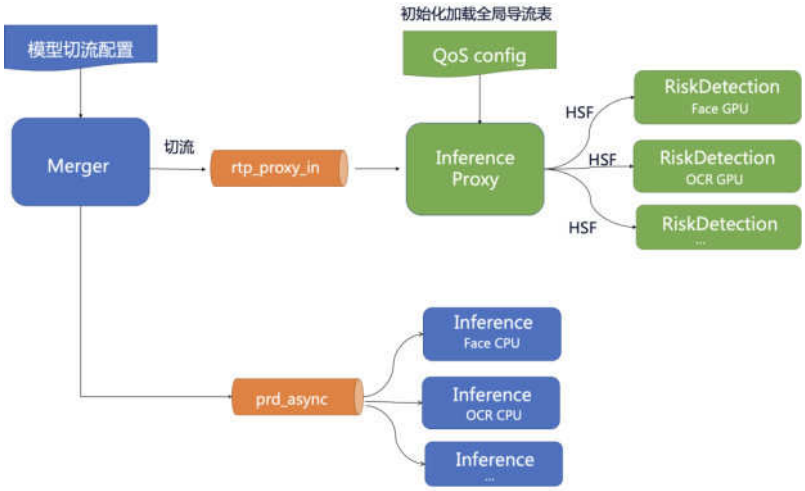
核心类图：



过去的 Inference-kgb 由于性能问题，TOP3 请求的模型存在严重的性能瓶颈。RD 上线后，我们可以同时支持 GPU 版和 CPU 版在线上运行，性能相比线上 CPU 版有了几十倍的提升，通过 InferenceProxy 进行模型服务接入，能够精准的控制机器

和流量配比。

通过 InferenceProxy 切流方案，我们能够将新的 RD 安全稳定的接入到在线服务中，承载 TOP3 模型流量，优雅解决了双十一大促过程中模型积压的问题。切流链路如下所示：



6. 未来展望

6.1 能力方面

模型推理加速：完成 Inference-kgb 迁移 RiskDetection GPU 加速，并且持续打磨 HighService 及接入更多 Backends，让模型接入更加简单方便，形成标准化接入流程；新增 CPU 通用加速流程，形成 CPU 标准化加速方案，并在 ID 模型上落地；调研 PPU 加速方案，尝试采用 PPU 解决在 / 近 / 离线资源瓶颈问题。

服务运维特性：完善核心三大服务及运维特性：Caching，Batching、Scaling；其中 Batching、Scaling 依赖 Backends 提供相应的能力，ID 实现通用配置化管理。

6.2 效率方面

模型源数据管理：模型文件、镜像、配置进行标准化管理，在 / 近 / 离线复用同一套管理标准。

简化模型发布流程：借助已有平台标准化模型效果验证、配置、部署、回滚、灰度发布流程。

6.3 质量方面

代码质量：由于 Drogo 多 Role 部署不能直接按照 Aone 发布流程部署，需要手动在 Drogo 上部署，导致 Aone 上设置的 CodeReview、单元测试等卡点无法直接生效。因此我们需要制定发布标准，并严格按照标准执行。

服务质量：标准化开发、测试流程，对模型服务质量、代码质量、服务质量提出明确要求，并遵守标准规范，不合规范不允许上线。完善系统监控，报警信息。

6.4 成本方面

提升资源利用率：通过 CPU、GPU 加速、Batching 等功能提升系统性能，从而达到减低计算成本的作用。通过弹性缩扩容落地，更加合理使用资源提升整体资源使用效率。

对模型类型约束：相同类别的模型尽量复用同一套代码（如分类模型）限制模型类别，不过度限制模型数量。类别固定有利于模型更方便的复用，减低开发、运维成本。

降本增效：持续度量模型 ROI，替换低 ROI 模型及策略。

大模型时代的阿里妈妈内容风控基础服务体系建设

本文作者：御医、陌奈、列宁、陌瑶、加木、吉多

一、内容风控业务背景及挑战

1.1 业务背景

内容作为营销的重要载体，能够促进信息的交流和传播。在营销场景中，广告高曝光的特性放大了风险外漏带来的一系列问题，少数用户为了引流获利，可能会发布一些涉嫌违规内容，也存在部分用户对广告法的理解存在偏差，误发布涉嫌违规内容。对于发布平台而言，如果这些内容确实违反法律法规，将会影响用户对平台的正面评价。

因此，阿里妈妈的努力方向不光是使内容风控系统对变更的广告进行实时管控，还需要具备对百亿级别存量广告数据快速定位和清理的能力，迅速发现风险，防止涉嫌违规的内容扩大传播。

随着新一轮 AI 技术浪潮的兴起，阿里妈妈也推出了 AI 创意生产工具——万相实验室，助力商家实现 0 成本上新，推动电商进入“AI 上新“新时代。但相比原有的业务场景，新业务在响应时效和风险特性上都存在很大差别，这对内容基础服务能力建设工作带来了更大的挑战。

1.2 面临的挑战

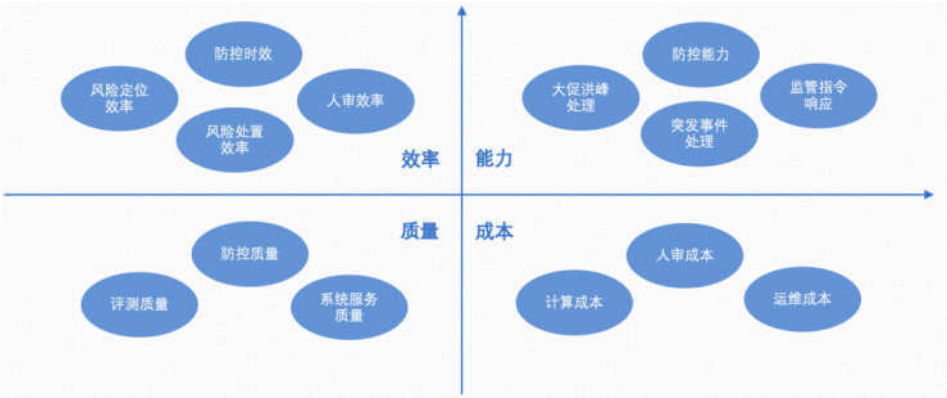
内容风险类型的多样性、变异性和迅速传播性等特征使内容风控面临巨大挑战，而新一轮 AI 技术浪潮的兴起对内容风险的防控能力、响应速度、计算资源以及成本管理都提出了更高要求。

1.2.1 业务的挑战

下面列举多种风险类型样例，上述特性导致防控过程中存在较大难度，且一旦外漏，影响可能难以控制。



1.2.2 技术的挑战



- **能力挑战**：比如每次大促结束后，相关内容的素材会集中在凌晨修改去标，整个系统面临巨大压力；为降低涉嫌违规内容外漏的风险，需要具备对全量数据快速识别、清理的能力，及时响应下线。
- **效率挑战**：AI 交互式生成内容形式，需要在内容展示前进行风险拦截，对内容识别时效提出了更高的要求。
- **成本挑战**：增量及存量内容数量急剧增加，图片、视频、直播内容风控计算成本较高，如何保障识别效率的同时，提升资源利用率、降低计算成本；部分风险机器识别准确率不高，需要人工进行审核，如何提高机器识别准确率，降低人审成本。
- **质量挑战**：由于数据量大，涉及的业务众多，如何快速评价风控效果及判断漏出情况，如何评估整体风险防控质量。

二、AI 驱动下的风控引擎演进

2013 年开始，阿里妈妈在广告内容风险治理方面开始加大投入。持续迭代至今，内容风控建设主要经历了三个阶段，逐步从纯人工规则到算法辅助规则，再到数据和算法驱动的阶段演进。期间无论是在业务层面还是在技术层面，都经历了巨大的变化。接下来，我们将介绍风控技术体系演进过程中遇到的问题，并分享基于 AI 驱动下的风控引擎设计和思考。



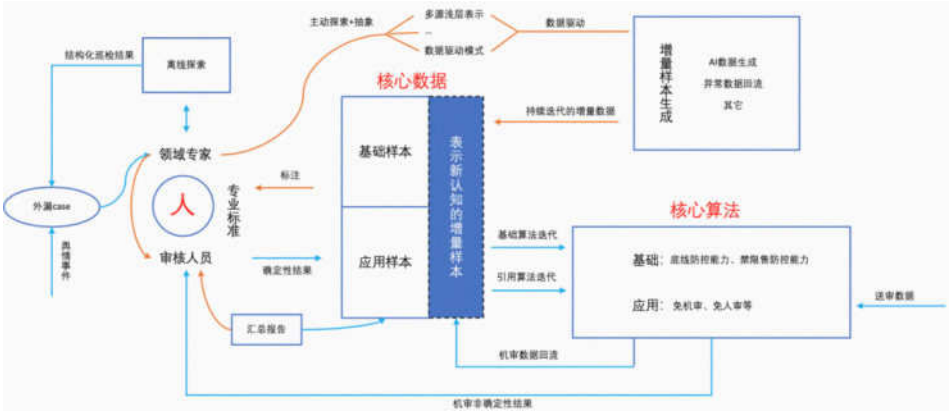
2.1 业务抽象层面

第一阶段：快速从 0-1 搭建防控链路，采用规则作为主要的防控手段，其中包括配置管控词、黑名单 / 白名单以及根据商品信用、用户和店铺等信息进行风险防控。然而，这种防控手段相对单一，只能有效应对一些简单的风险。然而，风险是不断变异的，通过简单的变换就能轻松绕过规则的限制。例如，微信这个词可以通过各种方式进行变化（如 VX、威信、徽信等），而图片上的风险更是难以在这个阶段进行有效的防控。

第二阶段：为了提升防控能力，我们逐步引入算法模型，对各类风险进行防控。如下图所示，我们提供了运营可以配置算法模型的功能，对应左上图中的规则配置中的操作符。这些表达式涉及到不同的模型防控能力。尽管我们引入了算法模型来进行风险防控，但整体流程仍然以规则驱动为主。然而，随着模型数量不断增加，通过手动调节阈值的方式很难使效果和计算成本达到最优。

第三阶段：搭建通过数据 + 算法的方式驱动风险防控，人善于定义和发现问题，因此

可以指导模型去解决问题及提供 Base case 辅助模型不断迭代及完善。针对业务定制化诉求及应急事件处理还需要人进行主动干预。下图是整个 AI 驱动流程，描述了样本、模型、领域专家在风控链路中的扮演的角色。

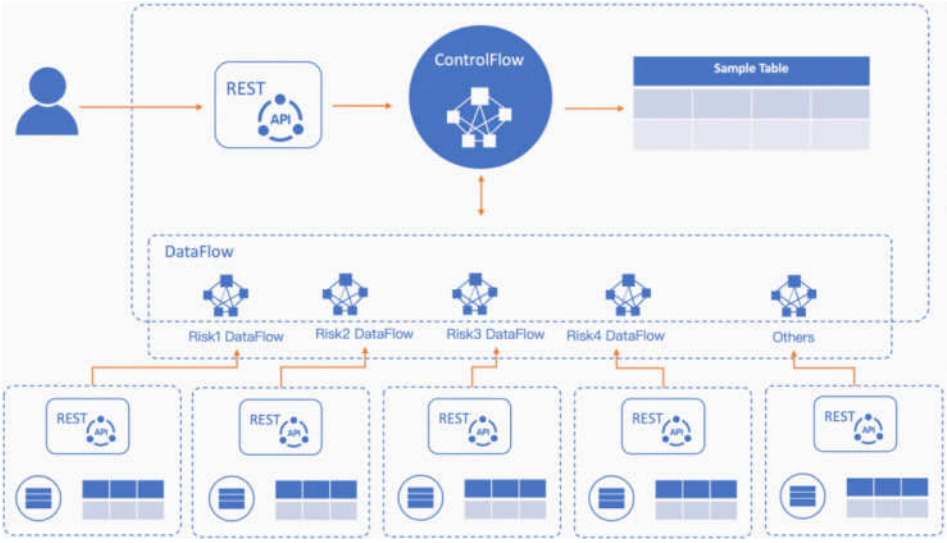


2.2 系统建设层面

第一阶段：主要涉及到词匹配、规则、及黑白名单系统，直接通过同步调用的方式，整体链路比较简单。

第二阶段：增加了模型及检索服务，模型服务由于 RT 较长采用异步（Metaq）调用方式，整个执行过程是以 DAG 形式表达，执行过程中需要读写状态。由于 DAG 节点较多（1000+）状态的管理已经成为性能瓶颈。而且异步链路无法保障返回时效，在强交互式的 AI 业务风控场景下，没法满足业务诉求。同时由于设计之初离线方面考虑较少，导致在离线表达和执行存在较大差异。为了解决以上问题且满足新的业务诉求，我们进行第三阶段的重构。

第三阶段：引入 Serverless 思想，将服务能力进行明确拆分，区分 DataFlow 和 ControlFlow 不同的职责，所有服务通过网关进行输出。DataFlow 负责纯粹的样本构建工作，ControlFlow 控制样本消费及最终结果流程。DataFlow 中以 DAG 的方式并发的调度下游基础服务（模型、词匹配、检索等），在离线在逻辑及结果层面统一。



上面简要介绍了整体框架流程，没有做太详细的展开，后续可以专门介绍整体框架设计细节。本文主要重心在基础服务能力建设上，接下来的内容将重点介绍内容风控涉及的模型、检索、词匹配引擎建设过程，及整体叶子节点运维管控相关内容。

三、大模型时代的风控基础服务能力建设

前面介绍了业务背景、挑战以及阿里妈妈内容风控演进的思路和整体方案，接下来重点介绍整体方案中涉及到的基础服务能力演进过程。内容风控核心建设的能力包括模型、检索、词匹配及 Ops 管控等。下面着重介绍这几个能力的建设过程。



3.1 极致性能的模型服务

去年我们重构了初版的模型服务能力，详见：阿里妈妈内容风控模型预估引擎的探索和建设。随着新一轮 AI 技术浪潮兴起，风控业务场景也在尝试“用魔法打败魔法”（如通过大模型识别图片中的内容转成文本，能够更好的理解图片中的内容），引入 BLIP、CLIP 等大模型进行风险防控。大模型对资源消耗较大，我们希望过滤掉一些确定没风险的内容以减少计算成本，因此启动了 RiskFree 项目，通过提取各个维度的特征采用 XGB 模型过滤掉没有风险的内容，从而减低模型计算量。针对以上业务诉求，我们在加速方面新增了大语言模型及传统机器学习两个方向的能力建设。



3.1.1 大语言模型加速

内容风控场景也在不断探索大模型在业务场景上的落地，应用大语言模型的认知解决深度模型难以解决的问题。在落地过程中遇到 RT 高，计算成本高等问题，我们调研了业界主流加速方案并进行对比，综合考虑，我们最终选择了基于 CUDA Graph 技术的自研加速方案。

3.1.1.1 主流加速方案简介

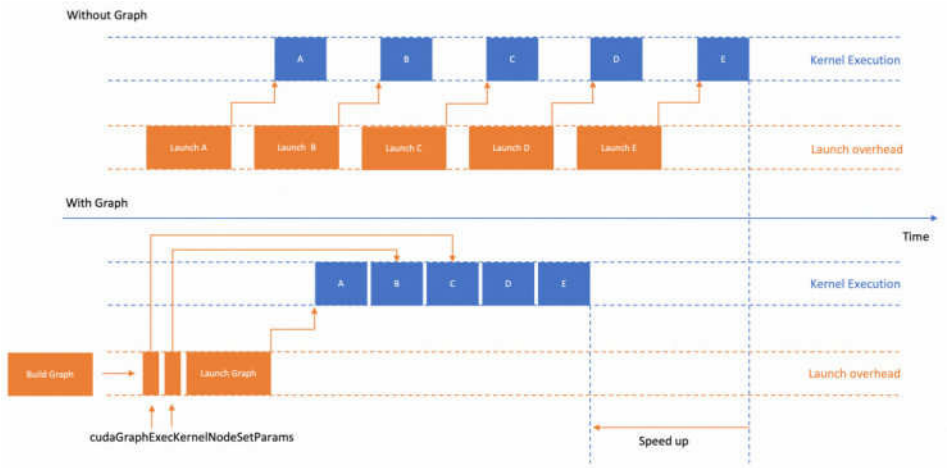
大语言模型加速方案也如雨后春笋般不断涌现，我们调研了市面上主流的大语言模型加速方案并进行对比如下表所示：

对比项	vLLM	Tensor-RT-LLM	PLL.LLM	DeepSpeed-MII	AIACC-LLM
Github	https://github.com/vllm-project/vllm	https://github.com/NVIDIA/TensorRT-LLM	https://github.com/openppl-public/ppl.nn.llm	https://github.com/microsoft/DeepSpeed-MII	暂无
公司	UC伯克利	NVIDIA	商汤	MicroSoft	阿里云
Batch	是	是	是	是	-
KVCache	是	是	是	否	是
INT8	是	是	是	是	是
FP8	否	是	否	否	否
多卡并发	是	是	-	是	-

3.1.1.2 基于 CUDA Graph 的自研加速方案 Kangaroo-Engine

TensorRT-LLM 有望成为行业标准，但一方面正式发布较晚，不能满足业务的急切需求，另一方面从 TensorRT-LLM 对运行环境、镜像、驱动等要求较严格，像我们主流在用的 T4、P100 卡不在其支持范围之内。综合考虑我们自研了一套加速引擎 Kangaroo-Engine。

Kangaroo-Engine 是一个基于 CUDAGraph 对大语言模型进行优化的推理引擎，它的设计优势在于保留原生 PyTorch 易用性的同时，支持高性能的推理加速方案。下图将一组 CUDA 算子调用和其他 CUDA 操作组合在一起，并使用指定的依赖树执行。通过结合与 CUDA 内核启动和 CUDA API 调用相关的驱动程序活动来加速工作流程。



CUDAGraph 没有广泛应用的核心因素是其苛刻的运行条件。模型必须是静态 Shape，且输入输出的地址以及中间使用的地址都必须固定。如果上述变量有变化，要么需要重新 Capture 一个新的 Graph，要么需要调用一个开销比较大的接口来修改 CUDAGraph。而 CUDAGraph 会占用比较大量的 GPU 内存，导致不可用。

为了更加方便高效的使用 CUDAGraph 技术，Kangaroo-Engine 实现了以下功能：

1. 对动态 Shape 情况，进行多次 CUDAGraph capture。
2. GPU 内存复用。多个不同的 Graph，复用 GPU 内存池，复用输入输出地址空间。
3. 对不同 Shape 做分桶 Padding，避免 Graph 过多，占用过多的 GPU 内存。

在对内容风控业务 BLIP 模型的优化过程中，Kangaroo-Engine 可以将图生文 BLIP 模型在 A10 上的典型数据单并发单次推理 RT 降低约 1 倍，可以满足内容风控业务的需求。

3.1.1.3 在不同设备上的定制优化

针对内容风控 P100 板卡，无法使用 MPS（Multi Process Service）来增加 SM（Stream Processor）利用率的特点，我们利用不同 Stream 可以并行计算的特点，取得了约 1/4 的 QPS 提升。

针对 Tesla T4 计算卡相对来说带宽更低，FP16 算力更高的特点，我们利用 TensorRT 8.6 Engine 充分利用 TensorCore 上的计算资源对矩阵乘进行优化，将 ViT（Visual Transformer）的每个 Block 所需时间降低 5 倍。

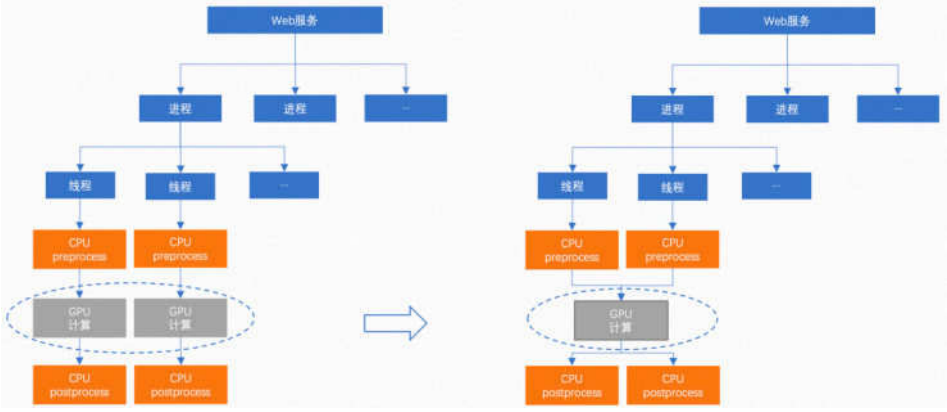
3.1.1.4 优化效果

模型	机器	当前QPS提升比例
BLIP图生文模型	P100	不使用
	T4	不使用
	A10	273%
BLIPITEM图片打分模型	P100	124%
	T4	390%
	A10	356%

以上针对内容风控业务大模型，构造了服务优化 + 多进程多线程 + Kangaroo-Engine/TensorRT 的优化方案。未来我们还将进一步尝试服务优化和低精度量化，为内容风控业务的大模型探索提供算力支持。

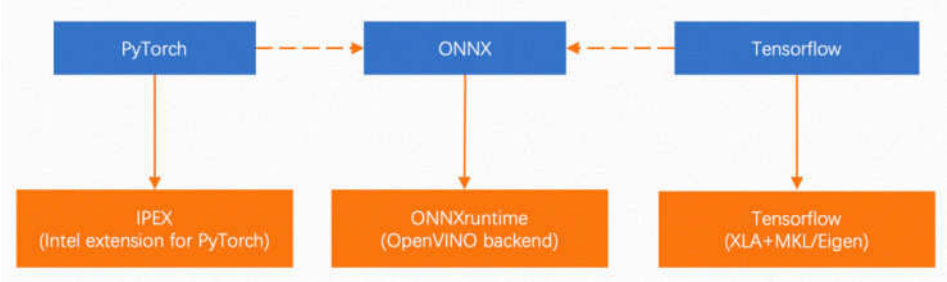
3.1.2 神经网络模型加速

针对神经网络模型的优化，我们在《阿里妈妈内容风控模型预估引擎的探索和建设》的工作基础上，增加了 CPU、GPU 分离能力，具体方案是将模型计算过程中依赖的 CPU 计算和 GPU 计算进行进程分离。可以避免因 CPU 计算瓶颈影响 GPU 性能，同时避免 GPU 内存大小限制了 CPU 进程数量导致性能瓶颈。缺点在于，CPU、GPU 的通信需要通过共享内存，CPU 和 GPU 进程通信需要序列化反序列化。对于较小的模型，这部分开销占比较大效果不佳，需采样原方案。详细设计如下图所示：



此外，由于内容风控业务经常会利用 CPU 资源来弥补 GPU 资源的资源缺口，我

们针对 CPU 做了定制化优化。对于 PyTorch 模型，我们使用 Intel 提供的 IPEX（Intel Pytorch Extension）进行自动转换和加速。对于 Tensorflow 模型，我们使用 RTP 自研 Tensorflow 版本，利用其中的 Eigen 以及 MKL 库进行加速。对于 ONNX 格式的模型，我们使用 ONNXruntime 进行加速。对人脸特征提取模型、涉政 Logo 模型、OCR 模型、Easyvision 图片分类模型、Moco 图片特征提取模型，本地 20 Core CPU 压测 QPS 分别提升 3~13 倍。离线压测有约 18 倍的提升，方案细节如下图所示。



该功能用于以下两类模型，取得非常显著的优化效果：

优化方式	QPS提升幅度	机型	优化推理引擎	优化整体服务
CPU/GPU分离	EasyVision 图片分类模型	T4 GPU	38%	830%
	Moco图片特征提取模型	T4 GPU	12%	68%
CPU定制优化	EasyVision 图片分类模型	20C CPU	610%	-
	Moco图片特征提取模型	20C CPU	260%	-

3.1.3 传统机器学习模型加速

内容风控有很多特征数据，这些数据无法通过人工配置阈值的方式使用，需要借助传统的机器学习模型来辅助进行决策。这些决策场景对延迟较敏感，直接通过官方协议部署无法满足业务诉求，我们针对多种业界决策树推理方案进行了深入实验，最终我们选择了对于业务场景来说性能最好且最合适的决策树推理工具 Treelite。

对比项	JPMML-Evaluator	TreeLite	H2O-XGBoost	XGBoost4J
支持语言	Java/Python/R	C++/Python/Java/R	Java/Python/R	Java/Scala/Python/R
开发厂商	OpenScoring	Microsoft Research	H2O.ai	DMLC
接入成本	低	中	中	中
说明	标准主流通用方案，支持大多数ML模型	支持几乎所有的决策树格式，采用分支预测技术加速CPU推理	一体化方案，不只支持树模型	官方版本但非官方开发，按框架划分各管一摊
支持模型	XGBoost、LightGBM、TensorFlow	XGBoost、LightGBM	XGBoost、LightGBM	XGBoost
目标方法	回归、排序、二分类/多分类、生存分析	回归、二分类/多分类、排序	回归、二分类/多分类、排序	回归、二分类/多分类、排序

Treelite 是一个决策树推理库，可以支持 XGBoost、LightGBM、Scikit learn 格式的决策树推理。可以把 Treelite 类比为决策树领域的 ONNX。它也包括 Treelite 格式和 Treelite runtime 两部分。此外 Treelite 还集成了 NVIDIA 的 FIL 加速库，虽然当前业务用不到，未来还可以支持 GPU 加速。

Treelite 在加速 CPU 推理的时候，会将决策树结构编译成 .so 文件，在指令层面对决策树的分支进行分支预测等优化，加快推理速度，在我们实测过程中，它的推理速度可以比 XGBoost、PMML 快一个数量级，我们采用 Treelite 作为通用的机器学习模型部署方案。

3.2 百亿规模的检索服务

3.2.1 业务背景

内容风控经过多年的发展，逐步形成了基于音、视、图、文模型的风险识别能力。但是由于风险变异快、对抗强，某些突发风险无法在短时间内积累足够多的样本来训练和升级模型，因此我们探索了类似拍立淘“以图搜图”能力，通过相似图文内容召回风险，满足快速响应、泛化识别的要求。下图展示了线上部分使用的内容风险相似检索用例：业务说明 业务要求 Doc 规模 Query 规模 读写比 RT 要求 索引构建效率 更新时效 应用成本 人脸检索 相似人脸检索（明星）“底线风险 在线高保障 要求低延迟” 十万级别 万级别 01:10 毫秒级 中 T+1 高

实时风险召回 通用历史相似黑样本过目不忘 “增量更新 运营手动添加样本 响应时效”

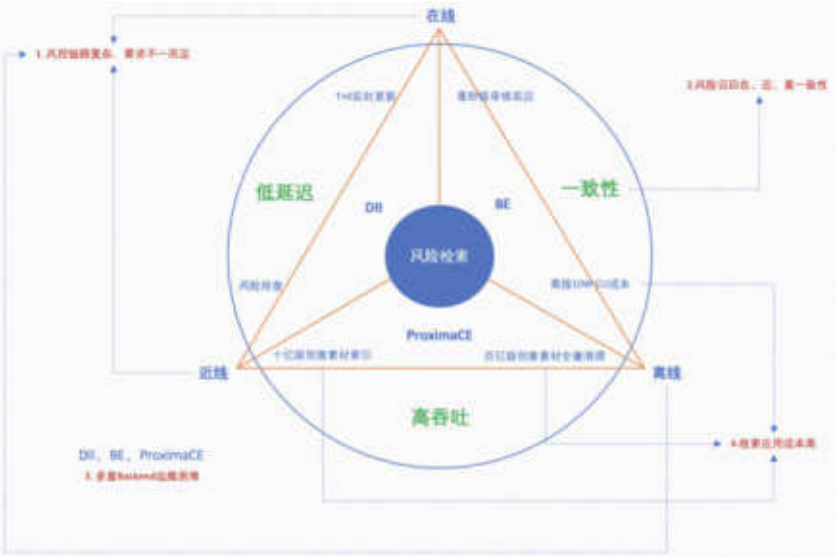
十万级别 千级别 0.111111111 毫秒级 中 T+0 高 免审通过 人审前、机审前低风险内容免审“全站低风险图文索引 吞吐和延迟性双重压力”十亿级别 万级别 1:10w 毫秒级 低 T+1 极高 风险排查 运营手动排查“百亿规模 Doc Ad-Hoc 查询、多字段过滤 类目检索、增量更新”百亿级别 百级别 1:1 亿 秒级 低 T+0 极高 全量风险召回 百亿级别离线全量内容风险清理“离线 UDF 类目检索、多字段过滤 高吞吐（百亿）”百万级别 百亿级别 10w:1 – 低 T+1 极高

	业务说明	业务要求	Doc规模	Query规模	读写比	RT要求	索引构建效率	更新时效	应用成本
人脸检索	相似人脸检索（明星）	底线风险 在线高保障 要求低延迟	十万级别	万级别	01:10	毫秒级	中	T+1	高
实时风险召回	通用历史相似黑样本过目不忘	增量更新 运营手动添加样本 响应时效	十万级别	千级别	0.11111111	毫秒级	中	T+0	高
免审通过	人审前、机审前低风险内容免审	全站低风险图文索引 吞吐和延迟性双重压力	十亿级别	万级别	1:10w	毫秒级	低	T+1	极高
风险排查	运营手动排查	百亿规模Doc Ad-Hoc查询、多字段过滤 类目检索、增量更新	百亿级别	百级别	1:1亿	秒级	低	T+0	极高
全量风险召回	百亿级别离线全量内容风险清理	离线UDF 类目检索、多字段过滤 高吞吐（百亿）	百万级别	百亿级别	10w:1	-	低	T+1	极高

为了更好解决业务问题，我们对主流的检索方案进行调研，希望通过一套方案解决以上不同的业务问题

3.2.3 方案选型

早期在建设检索引擎的过程中，我们面向解决线上业务问题进行方案选型，采用了淘宝搜索的 DII 引擎。随着业务发展，遇到外漏 Case 需要紧急处理时，很难排查到具体问题，因为我们建设了北斗风险排查功能，将全量的图、文提取向量进行索引，能够通过外部截图等快速定位到问题。由于 DII 不支持实时更新，我们采用了淘宝搜索的 BE 引擎。遇到紧急管控事件时，需要对全量数据进行清理，与北斗功能的差异在于全量清理需要进行相当复杂的计算，部分任务依赖检索功能，我们采用 ProximaCE 进行离线检索计算。业务发展至今我们在解决检索相关问题用了三套方案，这也给我们服务质量、运维、开发等带来很多问题，主要有：

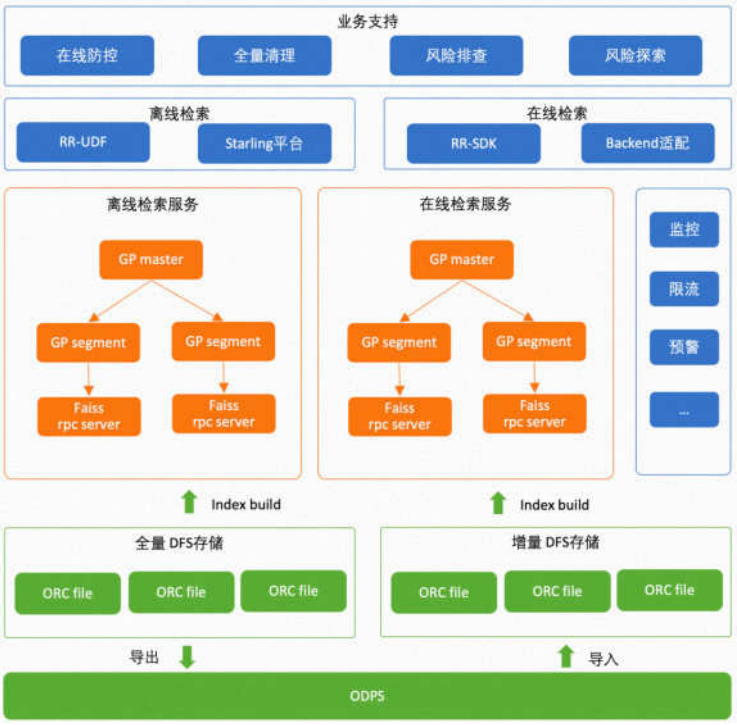


- **能力问题**：多套引擎在离线一致性无法保障。在线基于 BE 或 DII 实现，离线基于 ProximaCE 实现，引擎在线和离线完全割裂，没有统一的 Backend 及索引，导致在离线检索结果无法完全对齐。
- **效率问题**：多套系统操作和使用方式不一样，导致解决问题的效率降低
- **质量问题**：检索服务稳定性问题。主要体现在 DII 版本较老，过去曾出现因为请求参数的向量维度与 DOC 不一致导致 DII CoreDump 的问题，不同的业务场景对检索系统的要求有较大差别。
- **成本问题**：引擎接入使用方式不一致导致开发、运维成本上升

我们希望通过一套引擎能够解决检索在离线所有问题，通过调研发现内部的阿里妈妈自研超融合多模智能引擎 Dolphin 具备在离线一体的基础能力，因此我们与 Dolphin 团队同学合作共建，解决风控引擎遇到的问题。

3.2.4 检索核心方案

在线及离线的业务通过统一的引擎进行支持，在线、离线管控及运行环境存在一些差异，但底层的 Dolphin VectorDB 方案是统一的。在离线能够复用相同的索引文件，使用同一套 Core 执行逻辑，最终在原理上保障在离线统一，详见下图。



上图是业务整体框架，其中具体 Dolphin VectorDB 具体内核实现，详见《Dolphin：面向营销场景超融合智能引擎》。

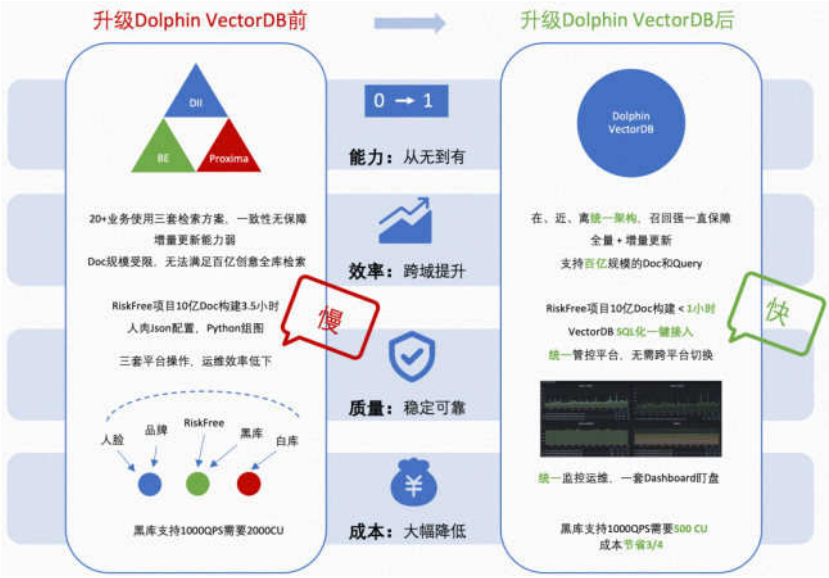
3.2.5 业务效果

升级 Dolphin VectorDB 后我们在多个维度对系统进行效果评测及验证，如下：

- **在线检索能力**：在小规模索引场景在线检索性能与 BE 相当，但在 RiskFree 超大规模 10 亿级索引业务场景存在显著成本和性能优势。
- **在离线一体**：实现了真正的在离线一体，使用的索引一体、引擎一体，评测结果表明在离线检索结果精确到小数点后 5 位的数据一致率为 100% ，业务一致率 100%。
- **统一检索协议**：统一基于 RR 协议接入业务，打通 Dolphin 作为 Backend 引擎，用户接入不同平台仅需掌握一种协议，大幅降低接入门槛。
- **统一管控平台**：打通 Dolphin 管控和索引构建接口，业务接入效率由小时级提升至分钟级。
- **统一监控告警**：构建统一的实时 Dashboard，一个大盘即可对所有业务的核心指标 (如：错误量、吞吐、RT、机器水位) 和配置 (如：是否黑库、表示算

法、索引算法、距离算法）一目了然。

- **实时业务报表**：一个页面展示所有业务数量、引擎分布、索引数量、详细配置等信息。



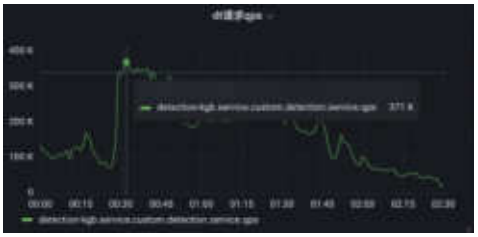
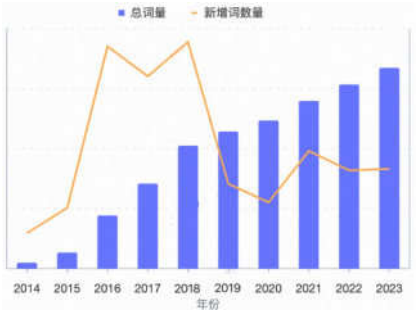
此外在开发成本方面，Dolphin VectorDB 提供 SQL 数据库查询语法和产品化索引导入平台，新业务接入时效从小时到天级别提升到分钟级，开发效率提升非常大。

3.3 全弹性的敏感词匹配服务

词匹配服务是风控最古老也是最直接有效的系统之一，它的功能主要检测输入文本中是否存在风险词，匹配方式包括模糊和精确两种，类似字符串的 Contains 和 Equals，当然真实使用的时候还包括加减词、通配等逻辑。具体管控的词由运营配置，按照词包的方式管理，如红线词包，敏感词词包等。

3.3.1 业务背景及遇到的问题

由于词匹配服务承载的 QPS 较高（峰值 40W QPS），且加载的词表逐渐增大（超过 100 万词），我们对其做了多次优化，包括解决输入文本归一化瓶颈问题，升级底层匹配算法等。还存在以下几类问题。

峰值QPS	敏感词增长趋势
	

- **能力问题**：文本预处理能力较弱，只是简单列举，没有引入文本还原，字音字形相近字匹配能力；
- **效率问题**：词数量超过多大，人工运维效率低。词表大加载出现 FullGC 导致时效问题；
- **质量问题**：词管理混乱，出现大量重复配置的词内容，在离线代码存在不一致问题，导致计算存在差异；词表过大，系统加载时出现 FullGC，DB 经常出现慢 SQL 影响服务质量；
- **成本问题**：历史原因运营管控词和算法产出词，由于匹配逻辑存在差异实现两套系统，导致开发、运维成本增加。

之前优化思路是面向解决现有问题出发，没有从更长远的角度考虑未来可能出现的问题，如词超过 1000 万单机无法承载；词匹配除了功能之外的服务及运维特性等。今年我们从现有问题出发的同时，跳出线性优化的思路对词匹配服务进行整体重构。

3.3.2 词匹配算法对比

我们调研了主流的词匹配相关算法，从多个维度对比算法性能，如下：

算法	多模支持	增量支持	时间复杂（平均/最坏）	空间复杂度	优点	缺点
KMP	否	否	$O(n)/O(n)$	$O(n)$		只有单模匹配
AC	是	否	$O(n)/O(n)$	$O(n)$		不能增量
BM	否	否	$O(\frac{n}{m})/O(n)$	$O(n)$	平均时间复杂度小	只有单模匹配
WuManber	是	是	$O(\frac{n}{m})/O(n)$	$O(n)$	平均时间复杂度小、支持增量更新	• 复杂度依赖模式串长度 • 复杂度不稳定，有可能退化 • 需要根据长度建立多个数据结构才能覆盖模式串长度范围

由于 KMP 和 BM 只支持单模匹配，我们选取 AC 和 WuManber 算法进行实验对比，相关指标如下图所示：

算法	匹配耗时数据（1000w数据平均）	构建时间	内存使用量（MB）	CPU使用率
Wumanber	100us	2.0s	396	100%
AC自动机	70us	2.4s	1354	100%

Wumanber 内存占用更小，且支持词表增量更新，我们采用 Wumanber 算法作为默认算法。

3.3.3 词匹配与向量检索方案融合

对比发现检索和词匹配在功能上有很多相似之处，例如都依赖一份检索文件，都需要 Build 索引、分发加载索引、通过匹配算法获取结果。主要的差异在于索引的内容及匹配算法。因此我们希望能够复用检索引擎索引管理、分发、分布式分行分列执行的能力。

3.3.4 优化效果

系统升级后，我们从匹配 RT、QPS、系统负载、内存占用等多个维度进行评估，可以看到新系统在多个指标上都有成倍提升，详见：

序号	指标	对比项	新系统	原系统
1	单次匹配	RT	0.18ms/次	0.6ms/次
2	系统负载	QPS	25 QPS/CU	13 QPS/CU
		CPU	10%	4%
		load	1.3	0.8
3	批量匹配	QPS	25 QPS/CU	13 QPS/CU
		匹配RT	1ms	4.5ms
4	索引内存加载	内存大小	1.22G	1.71G

3.4 云原生的 DevOps 管控

随着底层基础服务能力建设逐渐完善，我们遇到了新的问题。目前，有上百个模型服务、检索、词服务等都需要对应开发、测试和运维。尤其大促期间需要扩容，而 GPU 资源紧俏，缩扩容时可能会面临资源会被抢占或没有资源。即使有资源支持，手动对 100 多个服务进行资源调节也是非常繁琐的事情，更别提调节资源后还需要手动修改对应服务的限流值。

因此我们考虑建设 Ops 相关能力，解决我们服务及运维遇到的问题。我们通过与阿里妈妈 DevOps 团队合作共建，将服务托管并且带来以下增益：

- 同时为模型、检索、词服务提供统一的运维管控方案；
- 面向研发插件的发布链路；
- 区别于传统的面向应用的整体发布方案，改一行代码也需要重新构建和发布镜像。我们提供了类似插件的版本管理和发布的能力。从而，可以实现：面向业务 / 算法代码做发布；发布更高效；高效的共享资源池的管理；
- 规划建设逻辑资源池来缓存释放走的资源，方便资源在逻辑上共享的几组服务之间腾挪。以避免大池子资源紧张时，释放的资源无法快速申请。

四、业务支持

基于基础服务能力，我们支撑了多个新业务场景的落地，及大幅提升老业务场景的性能及效果。下面分别从在线和离线两个维度介绍典型支持的业务场景及相关效果。

4.1 在线业务支持

4.1.1 AI 创新业务支持

今年在线最大的变化就是新增了 AI 创新业务风控场景，AI 创新业务风控与现有业务的差异点在于与用户是强交互特性。其它业务提交之后承诺在小时或者天级别返回最终结果（可能需要人审），然而 AI 无论是生文、还是生图，都需要在秒级返回。并且 AI 生成的内容与我们传统的内容也存在差异。

目前我们支持了万相实验室在内的多个全新 AI 业务及应用，底层依托于全新的基础服务建设，无论是上线、迭代速度，还是服务性能及稳定性都有非常大的提升，实现文本审核 RT 在 100ms 以内，图像 RT 在 1s 以内。

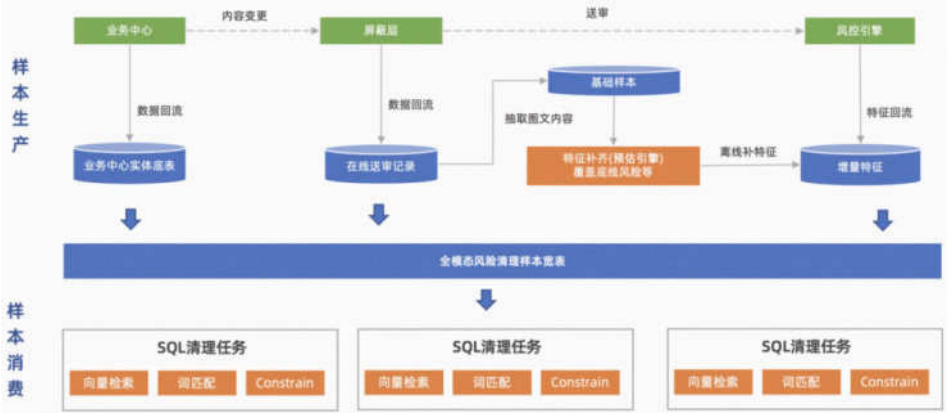
4.1.2 双十一大促表现

风控大促峰值流量与大部分业务是错峰的，每次活动结束后，带活动相关标识的内容都需要集中变更，这时候风控承载的流量是之前的几倍。

往年大促基础服务压力都很大，今年整体服务升级后，大促峰值期间服务整体水位稳定，基本没有明显的瓶颈。

4.2 离线业务支持

离线最核心的场景包括全量风险的清理及算法、运营同学基于历史数据进行效果回测。基础服务能力升级之后，已经支持部分场景的落地。针对离线核心解决在、离线一致性问题，确保每个服务在线、离线运行结果一样。这里以全量清理为例，下图是具体流程（黄色部分是基础能力在流程中应用的位置）：



基础能力上线后，我们解决了在离线一致性问题，提升了系统性能及资源使用率，更好的为整体业务保驾护航。

五、未来展望

阿里妈妈内容风控基础服务能力已进行全新升级，并在风控多个业务场景落地，取得了显著的业务效果。但随着未来技术的不断演进，我们还需要持续迭代，现阶段我们已从功能或者性能层面对各个基础服务能力进行完善，未来面向在线、离线计算一致性、基础能力的服务及运维特性等还需要进一步建设，具体包括：

5.1 能力方面

- **基础服务性能优化**：持续优化模型、检索、词匹配、通用预处理等基础服务性能，后续大模型在风控场景上的应用会越来越多，对算力的要求也越来越高，需要持续进行性能优化。词匹配适配检索链路，分行分列能力还在持续建设中。检索支持十亿级别的实时更新和查询能力还在持续优化中。
- **在、近、离线统一**：基于基础服务能力，构建在、近、离线统一的服务能力。每个基础能力做的足够简单可复用之后，可以基于这些能力之上，经过简单的封装调用就可以在不同资源、环境下进行快速拉起运行，解决在、近、离线业务问题。
- **弹性伸缩**：解决现有资源池 GPU 不足导致缩扩容困难，完善基础服务可观测性、可运维性，实现整体资源池合理隔离、按需分配，手动或自动进行资源伸缩。

5.2 效率方面

- **管控效率**：管理所有基础服务，支持所有基础服务任意版本在线回滚加离线回放，提供完善的模型上线压测流程
- **运行时效**：通过优化资源及系统，达到对业务透出时效减低的诉求，更好的服务低延迟应用场景

5.3 质量方面

- **服务质量**：对基础服务上线提出标准 SOP，必须经过完整的压测、评测、灰度流程才能发布到线上。发布后持续对服务进行检测，如果有问题支持快速恢复、回滚等应急措施。
- **代码质量**：随着基础服务个数越来越多，对在、近、离线一致性要求也越来越

高，需要将在、近、离线复用的代码逻辑进行统一管理，更新后在、近、离线能够确保都更新到最新代码。同时也支持指定任意版本进行发布和回放。

5.4 成本方面

- **计算成本：**未来模型会越来越重，优化模型本身性能及资源弹性调度和隔离，是减低成本的重要手段。后续也会在 PPU 上做更多尝试，相对而言 PPU 的成本更低。
- **运维成本：**目前有上百个模型，需要更好保障模型服务，管理模型完整生命周期，减低人工运维成本。需要结合前面提到的能力、效率、质量等综合落地。

六、写在最后

新一轮的技术浪潮兴起伴随着新的机遇与挑战，阿里妈妈内容风控团队无论在算法还是工程方向，都在积极探索前沿技术，以构建更全面的 AI 驱动的风控解决方案。期待与大家分享交流，同时也欢迎感兴趣的朋友加入我们 ~

简历投递邮箱：alimama_tech@service.alibaba.com

七、附录

- 阿里妈妈内容风控模型预估引擎的探索和建设
- 阿里妈妈是如何做品牌风险管理的
- Dolphin：面向营销场景的超融合多模智能引擎
- 广告深度学习计算：多媒体 AI 推理服务加速利器 high_service
- 新时期的阿里妈妈广告引擎

隐私计算

广告营销场景下的隐私计算实践：阿里妈妈营销隐私计算平台 SDH

作者：翾逸

一、概览

随着全球主要市场陆续出台个人信息保护政策，互联网生态中的数据安全和用户隐私保护问题变得越发重要且日趋严格。

如何在营销场景下安全合规的使用数据，维护在线广告商业模型的核心运作，成为当下广告生态中各企业亟需解决的问题。阿里妈妈一直注重对于隐私数据的安全合规使用，最大限度地保护用户隐私和数据安全。本篇分享阿里妈妈在保护数据安全和用户隐私方向的 Data Clean Room 实践产品**营销隐私计算平台 Secure Data Hub**（以下简称“**SDH**”），欢迎阅读交流。

1.1 产品介绍

营销隐私计算平台 SDH（Secure Data Hub）是由阿里妈妈提供的一套面向广告引擎、广告主、第三方检测公司在隐私安全环境下进行数据融合、隐私计算、联合建模的 Data Clean Room 解决方案。

SDH 基于多方安全计算（Secure Multi-Party Computation, MPC）、隐私保护机器学习（Privacy-Preserving Machine Learning, PPML）等隐私增强计算技术，立足于广告营销场景，贯穿广告投放的跟踪、采集、激活和衡量的全流程，实现对隐私数据的安全合规使用。在营销场景下的数据处理、人群洞察、投放优化、归因衡量、增效度量、触达监测等流程中严格保障多方数据的隐私安全和数据合规，为品牌提供跨域安全一致的数据决策能力。

SDH 已于 2022 年 12 月份通过了中国信通院第七批“可信隐私计算”评测，并获得

多方安全计算（MPC）和联邦学习（Federated Learning, FL）的基础能力专项评测的两项评测证书。



1.2 核心能力

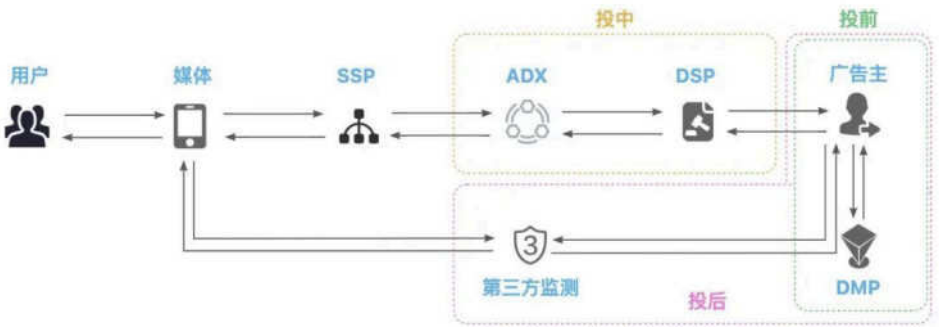


- **数据可用不可见**：业务方持有数据不出业务私域，通过对 MPC 元数据管理实现对数据“表级”和“列级”的隐私保护，网络通信中不泄露各方任何隐私保护字段（网络传输数据全部为可见数据，明文数据传输采用 RSA 加密等多种加密算法），基于 SDH 的 MPC 底层计算框架实现了数据可用不可见下的多方数据联合分析计算。
- **简单易用 API 接口**：SDH 提供了 SQL 的用户接口，向业务用户屏蔽了分布式执行、密码学技术等底层细节，并且 SQL 学习门槛低、具有完备的问题表达能力，极大地降低了 SDH 对接的开发和运维成本。

- **通用营销分析组件**：集成多种面向营销场景下的通用型数据转化、联合计算和归因模型分析组件，通过组件化使用支持业务方快速完成人群洞察洞察、归因衡量、增效度量等计算分析，提高开发和分析效率。
- **轻量化云部署方案**：面向不同云环境（阿里云、第三方云、私有云）提供多种轻量化部署方案，部署流程方便简洁，网络打通后即可享有 SDH 提供的隐私增强计算能力。

二、背景

在广告营销业务场景下，数据隐私问题贯穿整个广告投放的投前、投中和投后的全环节，覆盖广告投放链路的跟踪、收集、激活和衡量的全流程，同时涉及广告生态下的多方角色。如下图所示：



在广告生态中，需要在保障多方数据隐私安全和数据合规的基础上，合理使用数据构建广告系统，维护在线广告商业模型的核心运作过程，解决广告在不同投放阶段的多方数据联合分析和算法联合建模问题，同时合规适配来自个人隐私和数据安全合规性的约束。广告营销场景下的隐私计算问题既是挑战更是机遇，也同样是广告业内一直探索的技术方向。因此在广告营销业务场景中要实现数据的可用不可见，严格保障多方隐私安全和数据合规，并提供完整的数据融合、隐私计算、联合建模的隐私增强分析能力，SDH 项目应运而生。

三、技术架构

3.1 系统架构

SDH 系统架构分为 Console 管理、Agent 代理和计算引擎三层结构。



- **Console 管理**：负责基础数据管理和任务调度分发，不涉及业务方数据的存储和计算，由业务方管理、元数据管理、实例管理、权限管理等模块构成。
- **Agent 代理**：实现身份认证，并提供实例生命周期管理的 API，负责运行实例的启动、查询、停止等。
- **计算引擎**：对应各业务方在私域环境中部署的异构执行引擎，负责私域环境中逻辑执行计划的生成和物理执行计划的调度执行。计算引擎可细分为驱动层、调度层、引擎层和存储层，分别承担不同的执行计算能力。

SDH 同时提供了 SQL 的用户接口，利用 SQL 完备问题表达能力的优势，向业务用户屏蔽了分布式执行、密码学技术等底层细节，极大地降低了 SDH 对接伙伴的开发运维成本和技术门槛。

3.2 核心原理

3.2.1 元数据设计

为描述“数据可用不可见”能力，SDH 对数据的可用性和可见性按照数据列粒度进行了详细的分层定义，包括：

- **可用性**：关联键列属性、分组键列属性
- **可见性**：可见属性、哈希可见属性、分组可见属性、聚合可见属性



3.2.2 执行计划生成

SDH 计算引擎基于 Flink 计算框架实现，在执行计划生成阶段自底向上遍历执行计划，主要包含合法性校验和拆分改写两阶段。

合法性校验

SDH 定义了完整的数据可用性和可见性的推导规则，覆盖 Flink 内置的 Operator 算子、系统函数和自定义 UDF 函数。包括但不局限于继承输入列属性、继承可用性、调整列属性等。

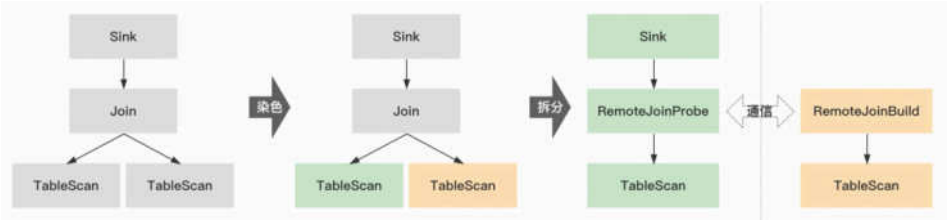
在 SQL 执行计划生产阶段，会优先级完成数据合法性校验。在此阶段，系统会结合输入数据的元数据信息进行数据可用性和可见性的推导及校验，验证满足合法性要求（即满足数据“表级”和“列级”的隐私保护要求）后，再进行 SQL 的拆分改写，否则任务返回权限不足的报错。

拆分改写

拆分改写阶段自底向上遍历执行计划，从输入数据开始根据数据持有方对执行计划染色，同时对 Operator 进行改写，最终根据染色结果将执行计划拆分成若干子图（每个参与方对应一个或多个子图）。以下面的 SQL 任务为例，其中 a 表和 b 表分别来自两个业务方，两表的 id 为不可见字段。

```
INSERT INTO result
SELECT a.id,
FROM a JOIN b
ON a.id = b.id;
```

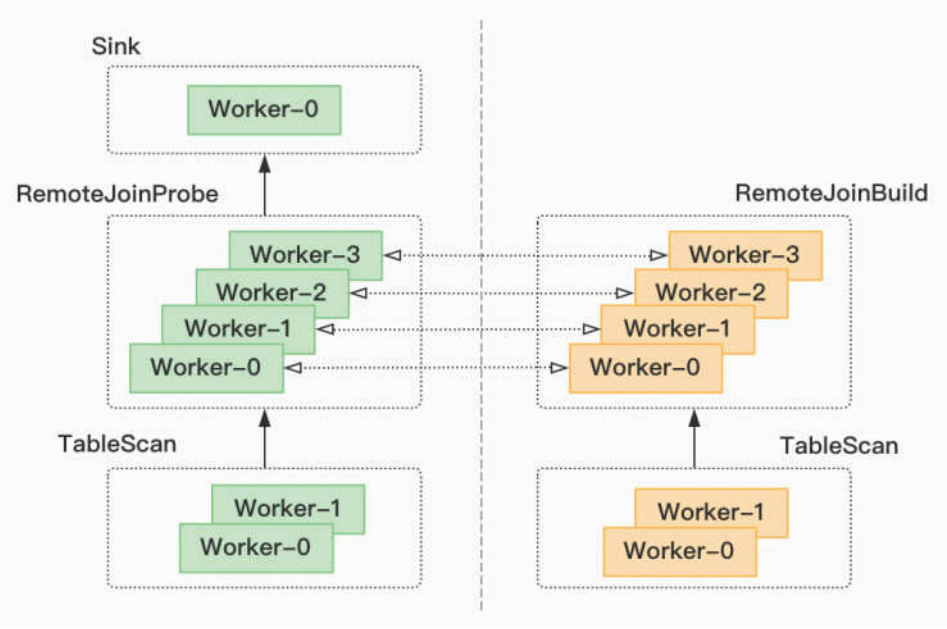
执行计划的拆分改写过程如下图所示，其中 Join 节点被改写为 RemoteJoinProbe、RemoteJoinBuild 节点，两节点基于网络通信实现了 id 字段的密文计算。



3.2.3 密态算子实现

Join 算子

分布式 Join 的常见实现包含 Sorted-Merge Join、Hash Join，目前 SDH 中已支持（Shuffle）Hash Join，即两方的数据根据等值条件中的 Join Key 按相同的规则进行分片，且分片数一致，这样双方相同 Join Key 的数据 Shuffle 后必然会分布在相同分片 ID 的 Worker 上，双方的 Worker 基于 Hash Join 进行连接即可。



Hash Join 划分为 Building 和 Probing 两个阶段。Building 阶段由 Build 侧遍历数据，对 Join Key 使用 ECDH 加密，同时发送给 Probe 侧请求二次加密，最终生成以加密 Join Key 为键的哈希表。Probing 阶段由 Probe 遍历数据，同样对 Join Key 使用 ECDH 加密，再发送给 Build 侧请求 PSI（Private Set Intersection）求交，从而完成 Join 条件中等式真值判断。同时为了提升 Hash Join 计算性能，SDH 在 Join 算子里引入了 Bloom Filter，在 Probing 阶段实现 Join Key 的预过滤，Join 性能有显著提升。

不等式运算算子

不等式真值的判断由表达式执行引擎执行计算，表达式执行引擎是多方安全计算能力的核心。以下面的 SQL 任务为例，其中 a 表和 b 表分别来自两个业务方，两表的

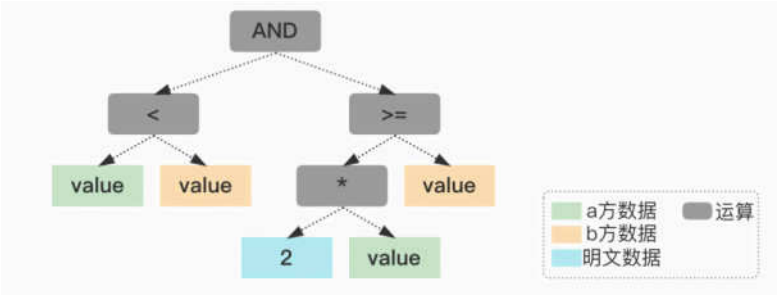
id、time、value 字段均为不可见字段。

```
INSERT INTO result
SELECT a.id, a.time, a.value
FROM a JOIN b
ON a.id = b.id
AND a.value < b.value
AND 2 * a.value >= b.value;
```

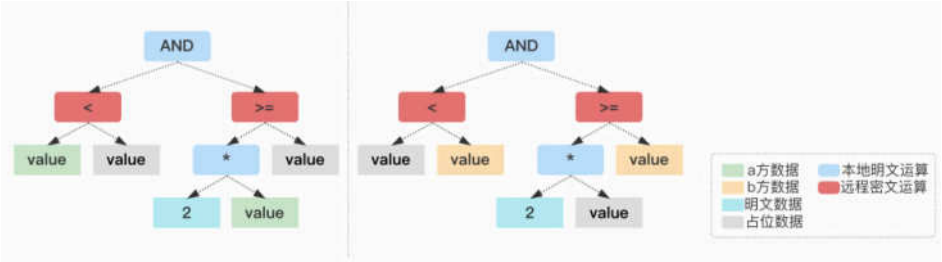
Join 条件如下：

```
a.id = b.id AND a.value < b.value AND 2 * a.value >= b.value
```

当 Join 实现采用 Hash Join 时，Join 条件中等式的真值会在 Hash Join 的 Probing 阶段进行判断，因此表达式执行引擎首先简化表达式，裁剪掉 Probe 阶段已执行的的等式，裁剪后生成的表达式树如下图所示：



表达式树中的运算节点分为本地明文运算（单侧参与运算）和远程密文运算（两侧参与运算）两类。表达式树执行阶段，两侧表达式执行引擎会按完全一致的后序遍历的顺序同步执行运算。



明密文运算单元

通过使用密码学的相关技术，包括 ECDH（Elliptic Curve Diffie - Hellman key

Exchange），秘密分享（Secret Sharing），同态加密（Homomorphic Encryption）等，SDH 里集成了多种类型的密态算子。SDH 中明密文运算单元已支持常见的逻辑运算（AND、OR）、关系运算（<、<=、==、!=、>=、>）、算术运算（+、−、*、/）。并且通过对密态算子的优化，持续提升密文运算单元的计算效率。

3.3 隐私安全保护

3.3.1 隐私保护能力

- **元数据保护**：提供“表级”别的权限控制；
- **字段级别保护**：提供“列级”别的字段可用性和可见性控制，支持针对不同的 operator 的字段隐私保护属性推导和合法性校验；
- **数据保护**：业务方原始数据不离开本地；同时保障网络传输的数据全部为可见数据（明文数据或加密数据），明文数据传输采用 RSA 加密。

3.3.2 密态算子能力

- **PSI 算子**：SDH 实现了基于 ECDH 的 PSI 密态算子，在 Hash Join 的 Building、Probing 通过 ECDH 的加密完成 Join 条件中等式真值的判断。支持百亿数据规模的隐私求交，并通过多种优化手段保证计算准确性和时效性；
- **密态比较 & 算术运算算子**：SDH 基于 Secret Sharing 封装了密态比较和算术运算算子，在保证计算精度（2 的 −32 次方）的前提下完成亿级别数据量级的密态比较和算术运算。

四、业务应用

UniDesk（https://unidesk.taobao.com/）是阿里妈妈推出的一款品牌数字营销的 Working Desk，立足于站外媒体矩阵，服务阿里经济体内部各业务、电商行业和非电商广告主进行站外广告投放和全域营销分析。目前 SDH 已经和 UniDesk 完成系统打通，服务集团内部和部分品牌广告主，主要用于对站外广告投放进行人群洞察、联合建模、效果衡量等营销分析。

借助 SDH 平台的隐私增强计算能力，在双方数据不出私域的前提下通过 MPC 和 FL 计算，实现多方的数据联合分析和建模，产出市场洞察和结案分析报告，帮助广告主衡量广告的投放效果，优化广告投放策略。

五、总结及规划

SDH 营销隐私计算平台通过 MPC 元数据管理实现对数据“表级”和“列级”的隐私保护，集成多类密态算子，兼顾明密文计算，基于 Flink 和密态执行引擎支持明密文计算任务的分布式执行，同时提供以 SQL 的用户接口和通用型营销分析组件，在保证数据可用不可见前提下可快速实现营销场景下的数据处理、联合建模、效果衡量等的计算分析。此外联邦学习解决方案 EFLS（Elastic Federated Learning Solution）已完成项目开源，对营销场景中的大规模稀疏的联邦学习应用有很大的参考价值。

SDH 未来将持续推进营销隐私计算平台的建设，基于隐私增强的大数据处理与机器学习建模能力，完善异构环境下的多模式、弹性化部署方案，优化百亿级数据规模的计算性能，支持更高计算复杂度的联合统计能力。以提供营销客户标准 Saas 产品化隐私解决方案，帮助广告主高效地进行广告营销场景下数据处理、投放优化、效果衡量的隐私计算分析或联合建模计算。

关于我们

阿里妈妈 SDS（Strategic Data Solutions）团队致力于用数据让商家和平台的增长战略更加科学有效。我们为阿里妈妈全线广告客户提供营销洞察、营销策略、价值量化、效果归因、隐私计算的技术服务。我们将持续在营销场景下的数据隐私安全和解决方案方向进行探索和落地，欢迎各业务方关注与合作。

阿里巴巴智能引擎算法平台团队负责为阿里巴巴的广告搜索推荐等核心商业提供 AI 工程平台和隐私增强计算服务。我们长期追踪各类广告营销以及搜索推荐场景所需的超大规模计算存储、稀疏及多模态深度学习、联邦学习及隐私增强计算等领域前沿，欢迎进行技术交流。

阿里妈妈营销隐私计算平台 SDH 在公用云的落地实践

作者：翾逸

一、概览

如何在营销场景下安全合规的使用数据，维护在线广告商业模型的核心运作，成为当下广告生态中各企业亟需解决的问题。阿里妈妈一直注重对于隐私数据的安全合规使用，最大限度地保护用户隐私和数据安全。继上篇分享阿里妈妈营销隐私计算平台 Secure Data Hub（以下简称“SDH”）在集团生产环境的技术方案后（延展阅读：广告营销场景下的隐私计算实践：阿里妈妈营销隐私计算平台 SDH），本篇分享阿里妈妈营销隐私计算平台 SDH 在公有云的技术实现和应用实践，欢迎阅读交流。

二、背景

随着全球主要市场陆续出台个人信息保护政策，互联网生态中的数据安全和用户隐私保护问题变得越发重要且日趋严格。2019 年国家将数据纳为“生产要素”，提倡驱动数据流通体现数据价值。2022 年“数据二十条”对外发布，加速了数据要素市场发展和数据要素高效流通，形成了数据要素开放共享的新形势。数据要素流通是数据价值释放的本质要求，安全合规是数据有序流通的基本前提，隐私计算技术为解决数据流通问题和数据价值挖掘提供了关键的技术基础和重要的技术支撑。

广告作为互联网最大的商业模式，2022 年中国网络广告市场规模突破万亿关卡，2023 年预计仍保持 12.9% 的高速增长，逐步形成一个体量巨大、生态完整的广告营销行业。隐私和数据安全问题对全球广告营销行业产生了巨大冲击和影响，产生了诸如禁止第三方 cookie、设备 id 合规采集使用、数据确权、数据安全合规流通等一系列的问题。在数字广告行业，数据是营销开展的基础，数据流通也会使得数据价值不断放大及提升。考虑到广告和用户数据会分散在广告生态的多个角色内，包括：用户、媒体、广告主、SSP、ADX、DSP、DMP、CDP 等，如何解决广告营销场景中数据孤岛和跨域数据流通问题，在保障多方角色数据隐私安全和法务合规的基础上为媒体、广告主和营销参与方提供及时、准确和安全的营销服务，已成为全球广告营销行业敏捷探索的前沿方向和共识。

阿里妈妈营销隐私计算平台 SDH 是一个面向广告引擎、广告主、三方 DSP/DMP 等合作方，在隐私安全环境下进行数据融合、隐私计算、联合建模的 Data Clean Room 产品。基于多方安全计算 MPC（Secure Multi-Party Computation）、联邦学习 FL（Federated Learning）、差分隐私 DP（Differential Privacy）等隐私增强计算技术，SDH 为品牌提供跨域安全一致的数据决策能力。

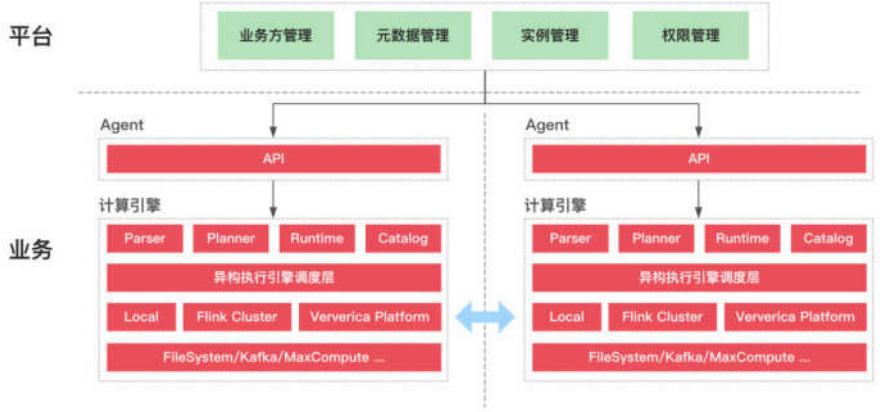
三、技术架构

3.1 核心能力



3.2 系统架构

SDH 公有云系统架构如下：



参与角色：

- **平台方：**部署 SDH 服务，负责基础数据的管理和任务的调度分发，不涉及业

务方数据的存储和计算。

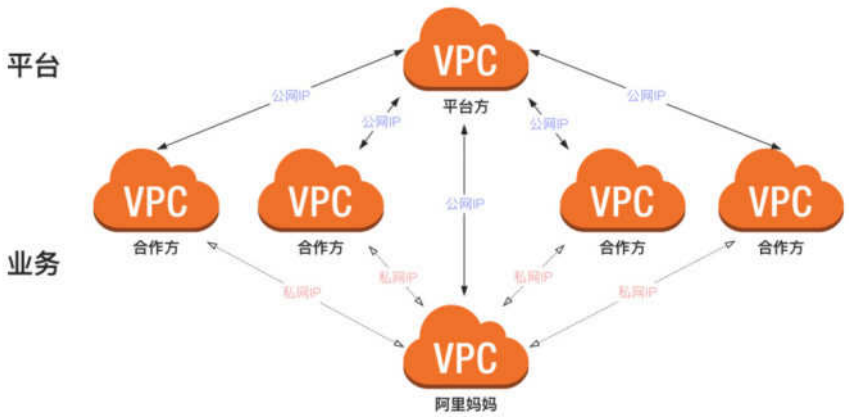
- **业务方：**在私域环境中部署 SDH 计算引擎，负责业务方私域环境中的存储和计算。

功能模块：

- **Console：**负责基础数据管理和任务调度分发，不涉及业务方数据的存储和计算。
- **Agent：**负责身份认证，并提供实例生命周期管理的 API，包括实例的启动、查询、停止等。
- **计算引擎：**负责私域环境中逻辑执行计划的生成和物理执行计划的调度执行。

网络通信：

- **平台方与业务方：**使用公网 IP 通信，传输元数据访问和任务分发，为单机通信，通信量较小。
- **业务方与业务方：**使用私网 IP 通信（VPC 对等连接），传输业务方间明密文计算数据，为分布式通信，通行量较大。



3.3 核心原理

3.3.1 元数据设计

SDH 对数据的可用性和可见性按照数据列粒度进行了详细的分层定义，以实现数据“可用不可见”能力：

- **可用性：**关联键列属性、分组键列属性。
- **可见性：**可见属性、哈希可见属性、分组可见属性、聚合可见属性。

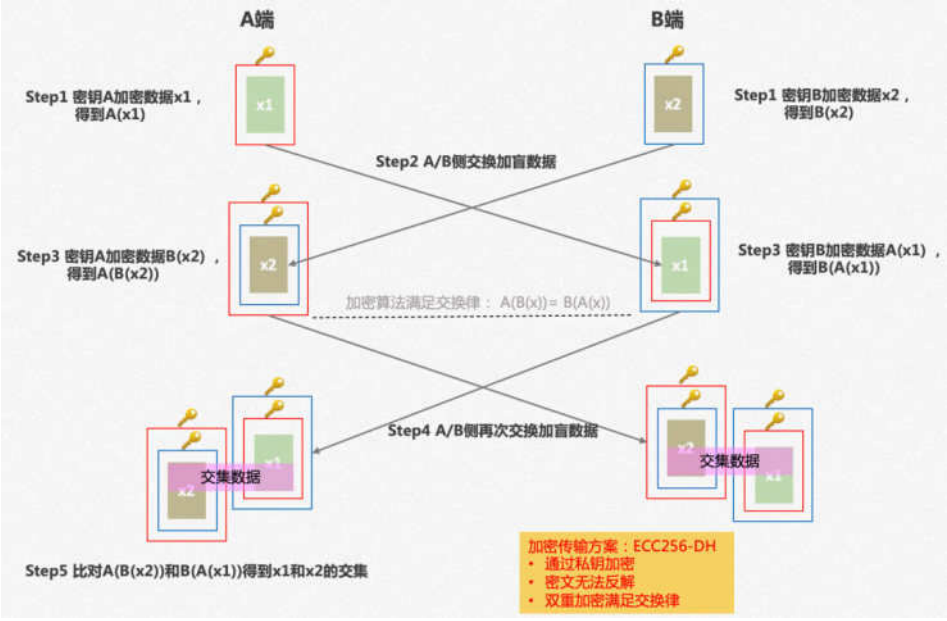
3.3.2 执行计划生成

SDH 计算引擎基于 Flink 计算框架实现，在执行计划生成阶段自底向上遍历执行计划，主要包含合法性校验和拆分改写两阶段：

- **合法性校验：**定义完整的数据可用性和可见性推导规则，覆盖 Flink 内置的 Operator 算子、系统函数和自定义 UDF 函数，以验证数据是否满足“表级”和“列级”的隐私保护要求。
- **拆分改写：**自底向上遍历执行计划，根据数据持有方对执行计划染色，对 Operator 进行拆分改写，将执行计划拆分成若干子图。

3.3.3 密态算子实现

- **Join 算子：**SDH 实现了基于 ECDH（Elliptic Curve Diffie–Hellman key Exchange）匿名密钥合意协议的 PSI Join 密态算子，加密流程如下图所示。在 Hash Join 的 Building、Probing 通过 ECDH 加密完成 Join 条件中等式真值的判断，同时引入 Bloom Filter 在 Probing 阶段实现 Join Key 的预过滤，以优化 Join 性能，支持百亿数据规模的隐私求交。



- **不等式运算算子：**基于 Secret Sharing 封装密态比较算子，其中不等式真值的判断由表达式执行引擎执行计算，可在保证计算精度（2 的 −32 次方）的前提下支持亿级数据量的密态比较。

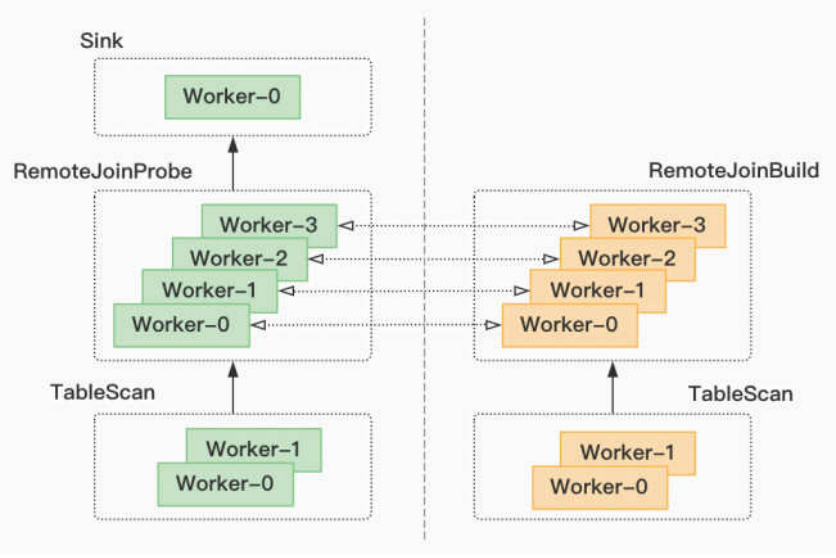
- **明密文运算单元：**基于 ECDH、Secret Sharing、HE 等密码学技术封装多种类型的密态算子，支持常见的逻辑运算（AND、OR）、关系运算（<、<=、==、!=、>=、>）、算术运算（+、−、*、/），并通过密态算子优化持续提升密文运算单元的计算效率。

3.3.4 隐私安全保护

- **元数据保护：**提供“表级”别的权限控制；
- **字段级别保护：**提供“列级”别的字段可用性和可见性控制，支持针对不同的 operator 的字段隐私保护属性推导和合法性校验；
- **数据保护：**业务方原始数据不离开本地，平台提供提供完备的数据授权机制，云上服务设置最小化访问控制策略，并支持多层访问鉴权保证数据隔离；
- **通信保护：**基于非对称加密 + 对称加密完成通信加密，即初始阶段双方使用非对称加密传输随机生成的对称加密密钥，后续采用对称加密方法进行加解密。保障网络传输的数据全部为可见数据。

3.3.5 分布式计算优化

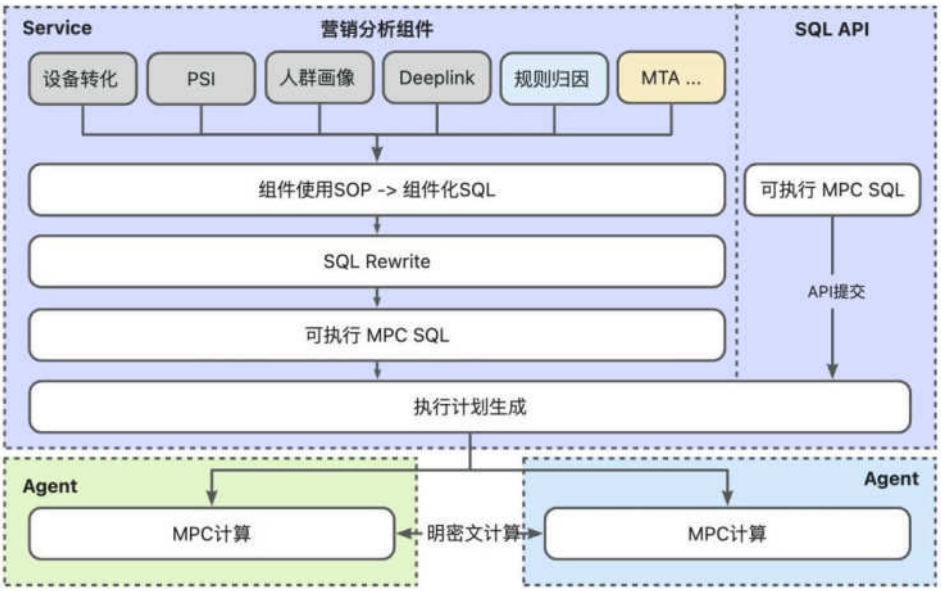
- **分布式 hash join：**SDH 支持（Shuffle）Hash Join，即两方的数据根据等值条件中的 Join Key 按相同的规则进行分片且分片数一致，即双方相同 Join Key 的数据 Shuffle 后会分布在相同分片 ID 的 Worker 上，双方的 Worker 直接点对点基于 Hash Join 进行关联。



- **分布式通信优化：**双方通信过程均为加密传输，为提升加密性能，采用非对称加密 + 对称加密的方案，即初始阶段双方使用非对称加密传输随机生成的对称加密密钥，后续通信采用对称加密方法进行加解密。为降低网络传输的开销，通信过程中的数据会组 batch 传输，并压缩数据以降低网络通信的数据规模。对于逻辑相对复杂的多方安全计算任务，借助谓词下推等优化规则将计算逻辑尽可能的前置，在本地对本方数据提前进行预过滤，从而进一步降低网络通信的数据量

3.3.6 营销分析组件

- **对外统一的查询 API：**SDH 对外提供统一的轻量化查询 API 接口，用户可通过提交 MPC SQL 或调用营销分析组件两种方式进行逻辑查询，其中分析组件可支持自动化 MPC SQL rewrite 再提交至计算引擎
- **Service 内组件集成：**营销分析组件集成在 SDH 的 Service 内，减少额外的部署和网络打通成本

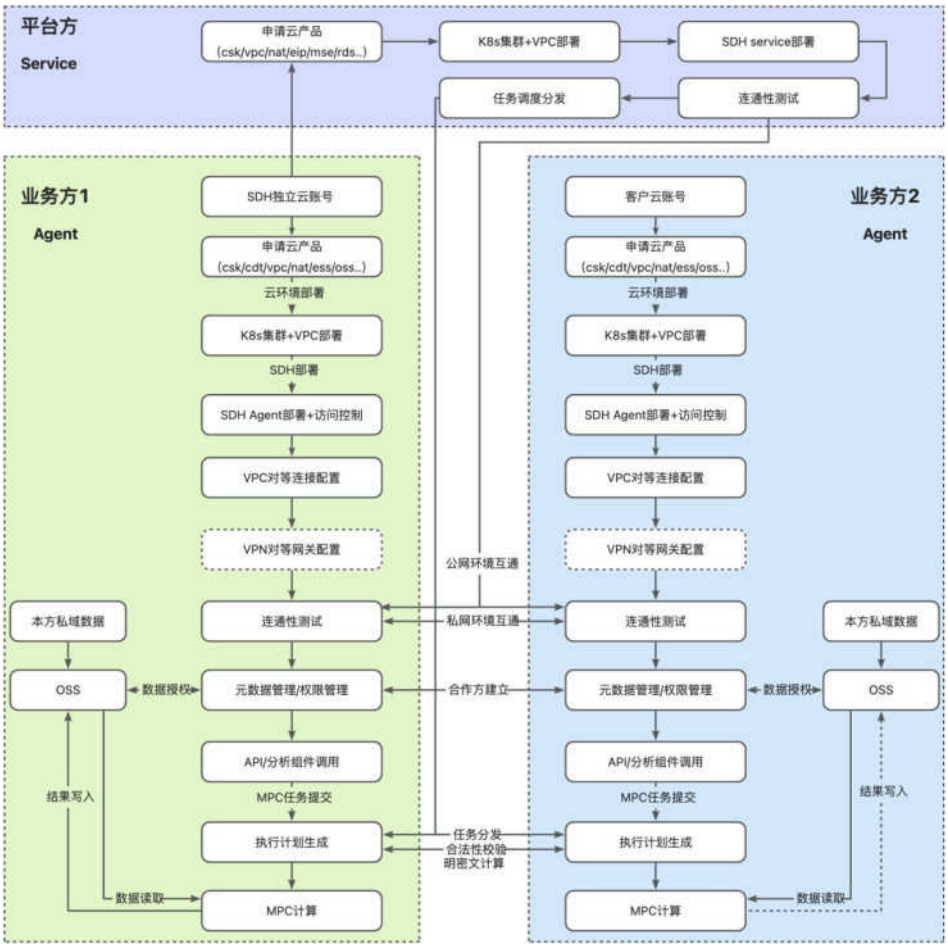


四、部署架构

SDH 提供面向不同云环境（阿里云、第三方云、私有云）下的云化部署方案。基于 Serverless K8s 集群可支持一键式 SDH 引擎部署，部署轻量，流程简洁，技术对接成本低。同时支持云资源的弹性扩缩容、按量计费。SDH 公有云部署方案如下图

所示，整体部署流程可概括为：

1. 云账号准备
2. 云产品范围申请，配置访问控制
3. Servicess K8s 集群部署
4. SDH 引擎部署
5. VPC 对等连接
6. VPN 连通测试（适用于第三方云、私有云部署）
7. API/ 分析组件调用测试



五、应用案例

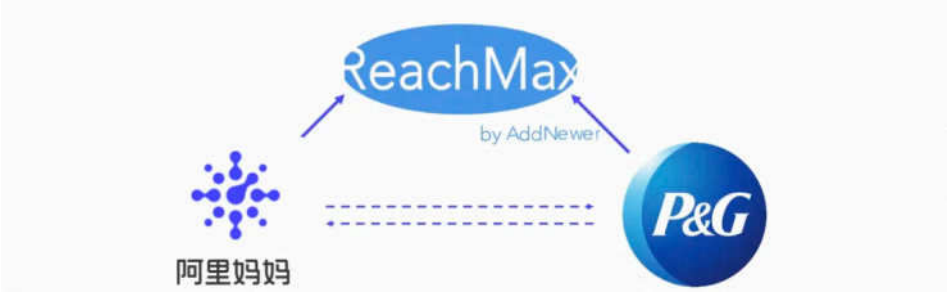
5.1 全域消费者资产分析

阿里妈妈联合伊利基于 SDH 营销隐私平台合力打造了全域消费者资产分析和数字化运营的应用实践案例。通过 SDH 的隐私增强分析能力连通伊利品牌域外投放人群（综艺回流人群、媒体直投人群等），结合“达摩盘”营销策略中心丰富的“人 – 货 – 场”用户标签数据和营销策略分析进行全域资产投产，形成投放策略，最后同步万相台人群超市进行场景投放，保障高价值人群的触达和投放效果。

基于 SDH 平台提供的 PSI 和 MPC 的隐私增强分析计算能力，伊利实现了在数据不出域的前提下，一方人群资产和达摩盘上品牌用户资产进行 MPC 的联合计算分析，完成一方人群的上翻和全域消费者资产分析，帮助客户完成全域资产沉淀释放营销价值，带来了 30%+ 的全域资产渗透率、购买转化率和 ROI 的全面提升。



5.2 广告跨域营销效果追踪



阿里妈妈联合加和科技在隐私计算技术上进行深入合作。利用 SDH 营销隐私计算平

台提供的隐私增强分析计算能力，在保障多方数据隐私安全和数据合规使用的基础上，针对广告投前的跨域用户识别、投中算法联合建模、投后的跨渠道广告效果衡量和全渠道用户资产分析场景进行深入的技术探索，完成了“基于隐私计算的广告跨域营销追踪和全域资产分析项目”的落地实践。解决了广告主跨域用户无法追踪识别、公域广告投放效果无法准确衡量、用户资产分散且数据割裂的实际营销痛难点问题。

基于加和科技持有广告公域投放数据和品牌私域数据，和阿里妈妈持有平台广告投放数据、用户标签数据和电商转化数据。利用 SDH 的 PSI、人群画像、规则型归因等营销分析组件，高效完成双边数据的 MPC 计算，实现跨域用户的识别和用户旅程追踪。沉淀安全、高效的跨域广告投放效果衡量和全域人群资产分析的解决方案，从而进一步完成跨域广告营销的触达人群特征分析、广告在淘宝和天猫店铺的转化效果追踪衡量和全渠道广告主人群资产分析，提供科学、真实的广告后链路转化和用户特征分析报告，并在数据安全性、分析多样性、计算准确性和数据时效性上较传统的数据授权方案上有显著提升。

该套基于 SDH 的隐私数据解决方案服务了加和科技 ReachMax 产品下 10+ 的头部品牌广告主，覆盖美妆、食品、日化等多个行业。帮助广告主提升广告投放的效果、充分挖掘广告数据价值，为商家预算的合理分配提供有力参考，形成“投放→引流→增长→投放”的良性循环。本解决方案已入选 2023 大数据“星河”优秀案例（延伸阅读：阿里妈妈 x 加和科技隐私计算合作成果入选 2023 大数据“星河”优秀案例）。



六、总结展望

阿里妈妈营销隐私计算平台 SDH 支持明密文混合复杂任务的分布式数据处理，能够实现包含隐私集合求交、密态关系及算术运算、窗口聚合等 20 亿 /h 的计算任务，计算准确率高达 2-32。基于 EFLS 框架支持十到百亿级别样本的 FL 训练。同时 SDH 提供 SQL 的 API 接口，集成多类通用化营销分析组件，支持多种轻量化云部署方案，进一步降低接入门槛并提供高效的营销隐私计算分析。跨域广告投放效果衡量和全域人群资产分析的解决方案和应用案例，打破了广告营销场景中数据孤岛和跨域数据流通问题，探索建立“可用不可见”的数据要素流通新范式，是隐私计算技术在整个广告行业中数据要素流通的创新性应用。

未来 SDH 会持续推动广告生态中数据要素安全合规地流通，致力于为品牌提供跨域安全一致的数据决策能力。不断完善 Saas 产品化能力，持续建设更高计算复杂度的联合统计和建模的隐私增强分析能力，帮助广告主安全、高效地进行广告营销场景下数据处理、投放优化、效果衡量的分析计算和数据建模。

关于我们

阿里妈妈 SDS（Strategic Data Solutions）团队

致力于用数据让商家和平台的增长战略更加科学有效。我们为阿里妈妈全线广告客户提供营销洞察、营销策略、价值量化、效果归因、隐私计算的技术服务。我们将持续在营销场景下的数据隐私安全和解决方案方向进行探索和落地，欢迎各业务方关注与合作。联系邮箱：alimama_tech@service.alibaba.com

阿里巴巴智能引擎算法平台团队

负责为淘天、AIDC、本地生活、菜鸟等业务集团提供 AI 工程平台和隐私增强计算服务。我们长期追踪各类广告营销以及搜索推荐场景所需的超大规模计算存储、稀疏及多模态深度学习、联邦学习及隐私增强计算等领域前沿，探索大模型预训练技术和大模型的商业化落地的工程底座支持，欢迎技术交流。

算法工程 / 引擎 / 系统建设

积沙成塔——阿里妈妈动态算力技术的新演进与展望

作者：木行、相遇、冰瞳、堇华、青礞、子瑞、叶卿等

在绿色计算和高质量发展的大背景下，算力的使用将朝着更加高效和智能的方向持续演进。**本文将介绍阿里妈妈广告引擎在优化算力使用方向上的最新实践**，欢迎留言一起讨论。

写在前面

在追求绿色技术的大趋势下，充分利用好算力资源实现系统的高效运行，尤其是在阿里妈妈广告引擎这种使用近百万 core 的系统中，就显得尤为重要。动态算力正是基于这个背景开始探索和建设起来用于优化算力使用效率的技术。它始终围绕让系统更加高效智能，充分合理的使用算力这个目标进行。今年通过跟业务方的紧密合作，也落地了多个有代表性的功能，主要包括潮汐算力，同城互备以及大促快恢等。经过几年的探索和积累（可参考文章 [1][2][3]），动态算力技术已经逐步形成自己的体系，具体如下：



图 1　动态算力技术体系

图中标黑色字体的部分是前两年建设的，红色字体是今年新增的部分，灰色字体是持续推进建设的部分。

本文将分别从应用层，通用层和基础层展开介绍动态算力技术的新变化，以及动态算力在阿里妈妈统一广告系统开发框架 EADS 中的接入方式，最后会对现阶段的工作进行总结并概述未来发展方向。

一、应用层

1.1 日常应用

1.1.1 潮汐算力

1）离在线潮汐算力

展示广告引擎召回阶段日常存在两条链路，分别是在线和离线，离线负责全量标签召回，在线因 RT 限制负责实时增量标签召回。离在线潮汐算力的目标是通过实时调配离在线的机器比例，来提升资源使用率，从而节省机器资源。示意图如下：

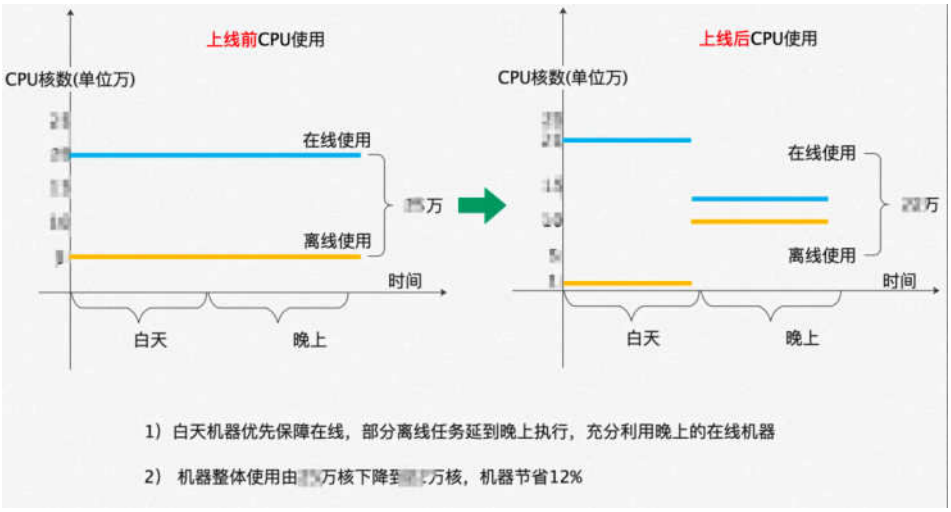


图 2　在离线潮汐算力原理示意图

在不影响离线任务实效性的前提下，节省整体机器资源。

基于动态算力的实时调控能力，技术架构图如下：

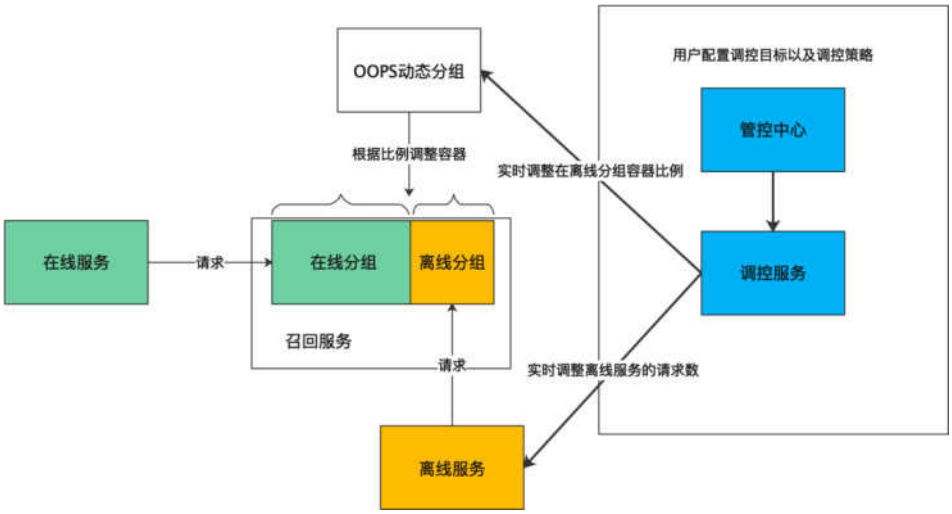


图 3　在离线潮汐算力方案

图中 oops 动态分组可以实现 10 秒内完成分组机器比例的调整。

整个功能经过与业务同学的多轮调优最终得以顺利上线，整个召回系统机器节省数万 core，比例接近 20%，全链路 RT 均值下降 2ms。日常调控图如下：

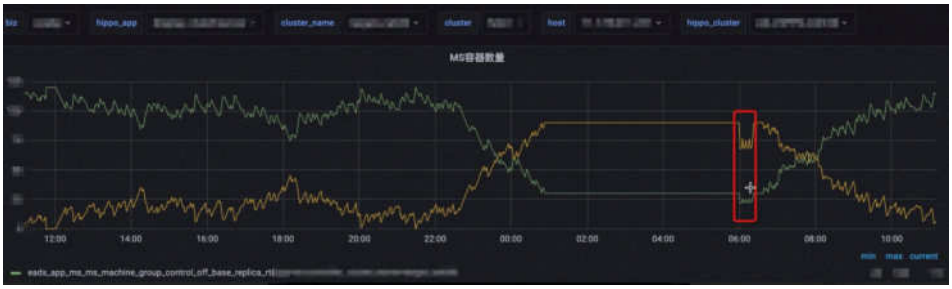


图 4　真实在离线容器调整图

说明：红框处波动原因是早上 6 点定时切数据引起

2）在线潮汐算力

广告引擎的四大核心链路包括召回，排序，策略和创意均已接入动态算力，原有功能里动态算力只能降档，在系统容量不足的情况下调整档位以增加系统容量。虽然有离线混部提升晚间集群 cpu 水位，但是考虑到在线服务稳定性，混部拉升水位有严格的限制，并未充分使用空闲算力。在线系统水位仍然存在着比较明显的日间高，夜间低的潮汐现象，如下图某服务的夜间 CPU 明显低于白天。



图 5　某服务一天 CPU 使用图

某场景夜间平均 RT，比日常低 20ms。

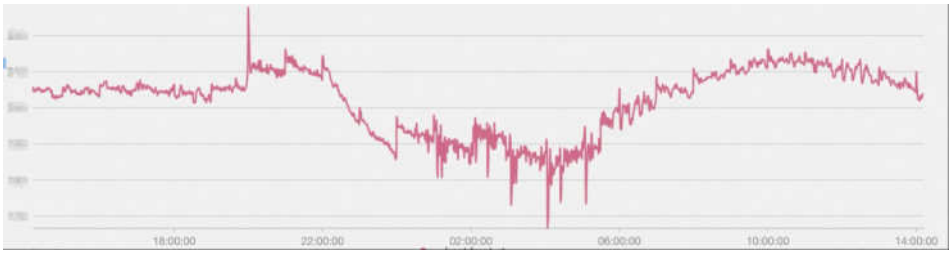


图 6　某核心场景一天响应时间图

这些都意味着在凌晨还有空闲的算力并未得到充分利用，而且混部占用的资源带来的收益跟业务收益无法直接度量。基于上述情况结合动态算力实时调整的能力，完全可以实现日常低峰时对某些功能点进行升档，这样可以直接利用空闲 CPU 升档带来业务上的收益，也是一种在混部之上贴近业务层的充分精细化利用空闲算力的能力。

技术方案：

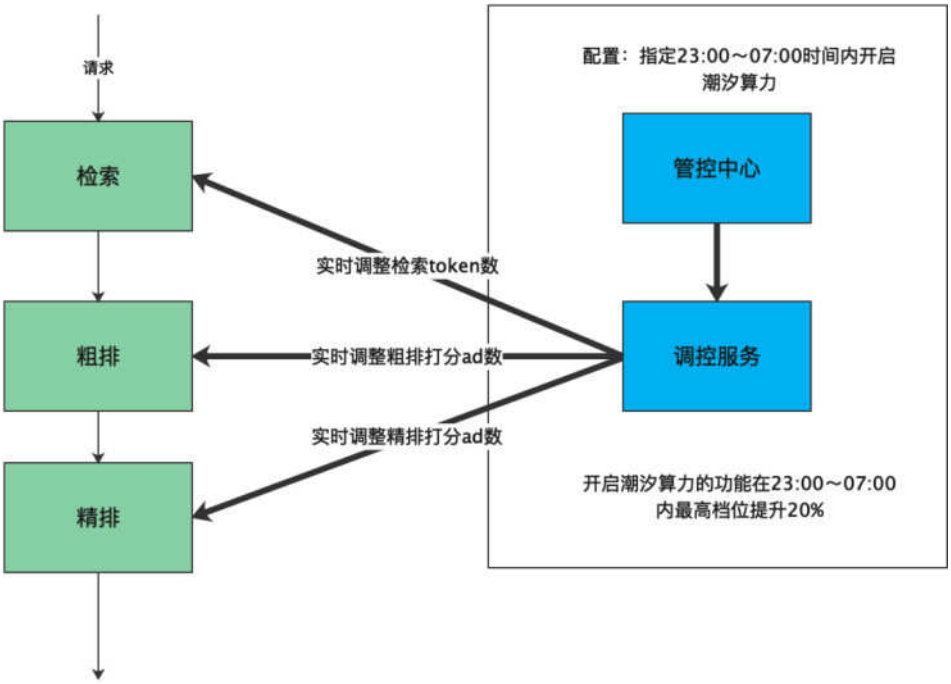


图 7　在线潮汐算力方案

实验效果：小流量上实验情况，检索 + 粗排 + 精排 相比日常满档升档 20%，多天累计实验效果 cost + 0.8%, pv+0.6%, rpm + 0.2%。

1.1.2 同城互备

展示引擎经过多年发展，系统已经非常庞大。在线机器资源使用了近百万 core CPU，随着集团对机器资源的控制，**增量机器资源有限**；另外核心场景的**可用 RT 也到达上限**。因此不管是机器还是可用 RT，展示引擎都没有空闲。这种情况下在线引擎机房间无法实现互备，线上一旦某个机房异常，无法即时切流止损，因为切流到另外一个机房，该机房因算力不足会出现大量超时甚至服务雪崩。展示引擎机房分布图如下：

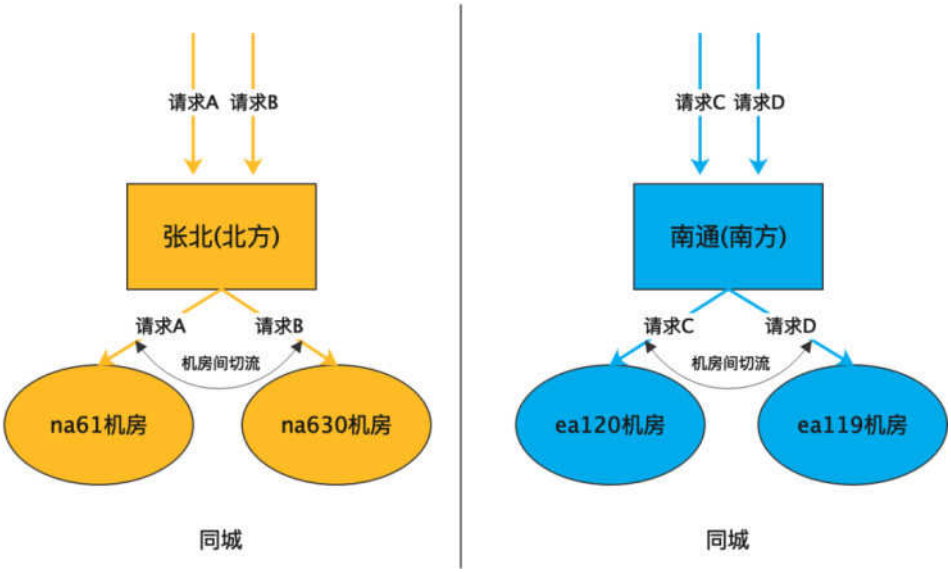


图 8　展示引擎机房分布图

但是日常高频的业务迭代，又无法避免线上偶尔会出现单机房异常 (coredump 或者数据异常导致客诉)，需要将流量切到正常的机房，此时需要接流的机房能够快速自动的调整扛住一倍的流量，并要求效果损失低于目标值 10%，从而为定位并修复问题争取更多的时间。

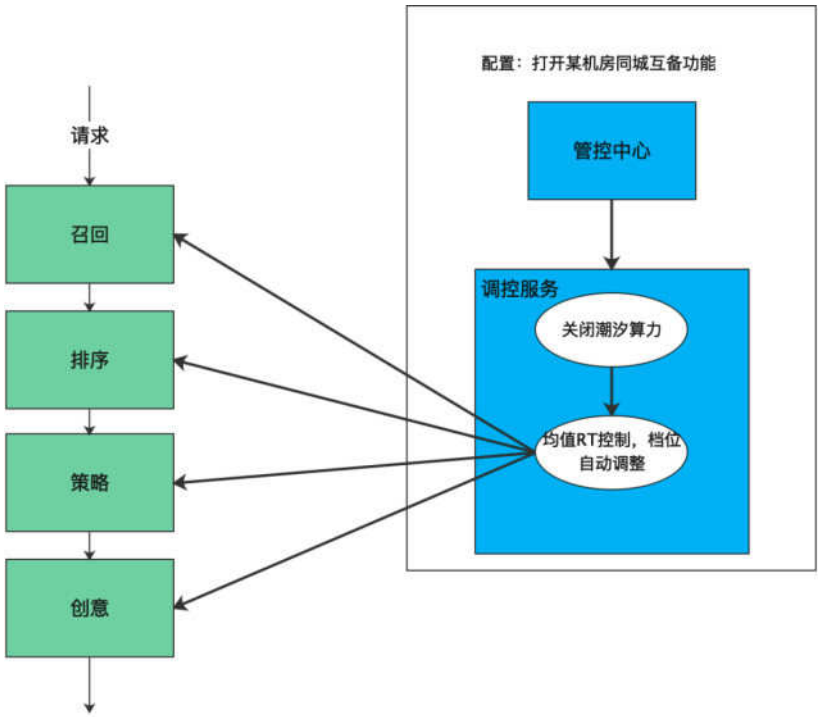
落地过程

第一步：梳理所有流量场景，保障各流量场景核心服务均能被调控和切流

第二步：单机房切流演练，通过优化调控策略以加快调控速度，另外重新分配各阶段均值 RT 以避免前端超时且降低效果损失。对于算力有明显瓶颈的模块 (0 档仍然无法满足 RT 要求)，进行适当的机器腾挪以满足稳定性和效果目标。

第三步：同城机房互切演练，拿到每个机房的效果损失数据形成报表。

技术方案



当需要机房切流时，对于接流的机房，人工打开该机房的同城互备功能：先关潮汐算力，将离线机器给在线用；然后通过全链路均值 RT 控制（自动化降各模块档位，优先降边际收益低的档位），避免前端超时增加。

最终结果

切流后可以在 3 分钟内完成所有核心档位收敛，并在前端超时率增加低于 1% 的前提下，最差机房 cost 下跌低于 7%，其他机房下跌均低于 5%。

未来同城互备能力还需要进一步优化，包括自动化的去适应业务引擎的变更，日常定期产出切流后效果折损的预估报告，保障系统随时可以以预期的效果损失实现快速切流，成为保障线上系统稳定性的一个基础可靠的系统工具，先卡大故障，再卡小故障，最终实现无故障。

1.1.3 RT 控制

展示引擎接入了数十个业务场景的流量，不同的场景前端给的可用 RT 不一样，但是又共用了一套引擎，那么当流量增加导致系统水位上升时，可用 RT 少的业务场景流

量会先出现超时 (现实中，虽然不同场景在不同阶段使用的策略有差异，可用 RT 少的场景策略会相对简单些，但存在有的模块在 RT 体现上差异不大，比如某场景 A 和某场景 B 的模型召回和精排)。今年新接入的场景 B 前端只给 200ms，而其他核心场景均在 300ms 甚至更多。为了保证该场景的可用性，在高峰的时候就要自动的降档实现 RT 的控制以避免超时。

结合动态算力的 RT 控制功能，具体的技术方案如下：

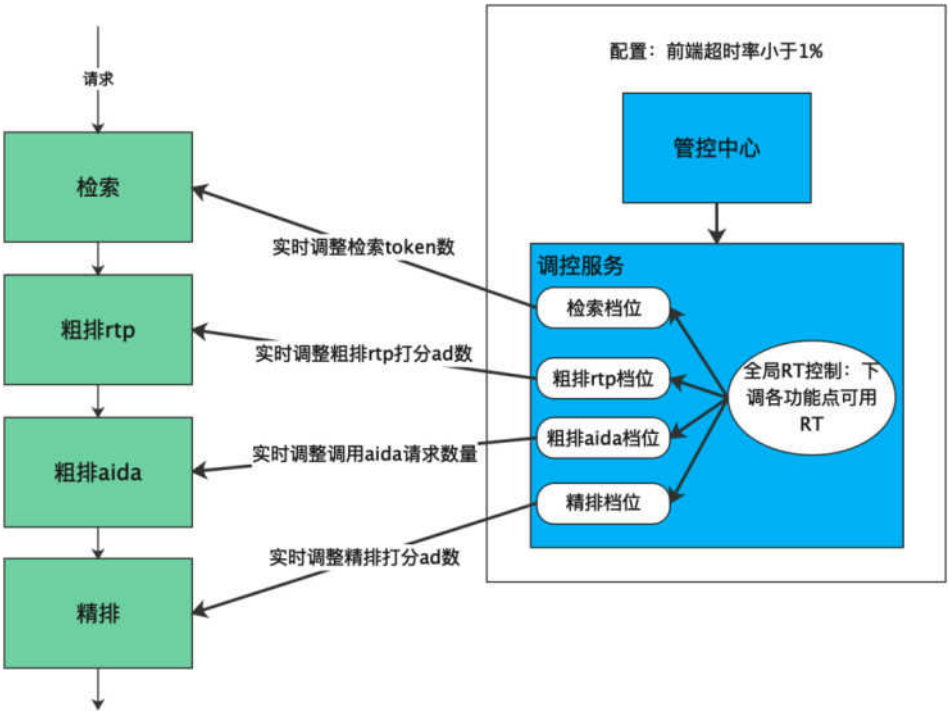


图 10　RT 控制功能方案

在全局 RT 控制中，档位下调时会优先下调边际收益低 (可参考 [2][3]) 调控点的可用 RT(均值)，该调控点可用 RT 下调后进而触发其档位下调，直到满足 RT 要求。

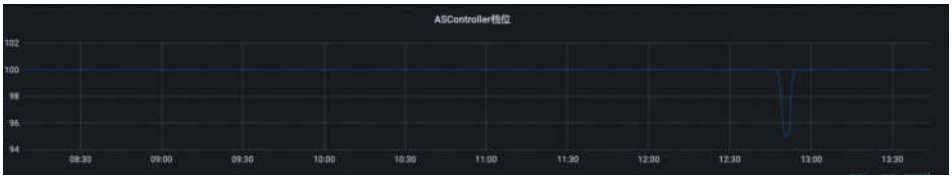
实战效果

当该场景某机房前端超时率上升超过 1% 时，那么该场景对应机房的全局 RT 控制档位将逐步下调，保证前端超时率低于 1%。超时率变化图：



图 11　日常场景 B na61 机房前端超时率

对应档位变化图：



今年在备战双十一压测过程中，前期未接入 RT 控制功能时，因为各场景流量都是数倍的增长，峰值附近系统水位非常高，部分机房场景 B 有几十个点的超时率。后面通过接入 RT 控制，超时率稳定在 1% 附近，顺利度过今年大促高峰。今年 11.10 大促高峰期间场景 B 控制效果图如下：



图 13　大促高峰期间场景 B na61 机房前端超时率

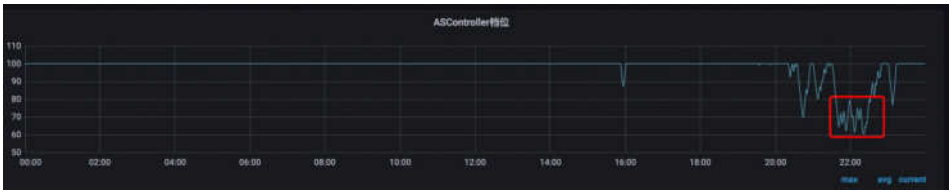


图 14　大促高峰期间场景 B na61 机房全局 RT 控制档位

另外全局 RT 控制有严格的实时报警以及档位波动记录，保障调控都在预期之内。

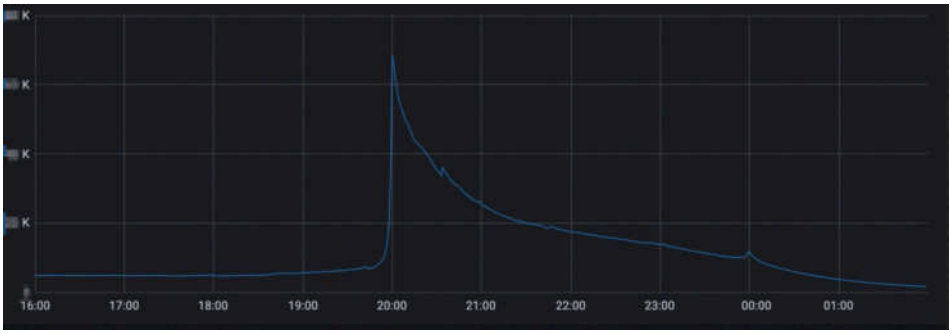
1.2 大促应用

1.2.1 大促峰值预调与快恢

功能描述：19:59:00 定时生效峰值档位集合以应对 10 倍的峰值流量，20:01:00 开启自动调控，进入快速恢复模式。随着流量下降，档位快速回升。所有功能点 3 分钟完成收敛。

实战情况

1）峰值实际流量



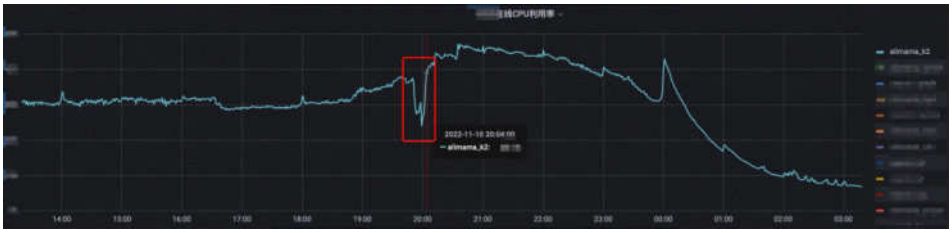
2）档位回升：3 分钟内基本恢复到满档



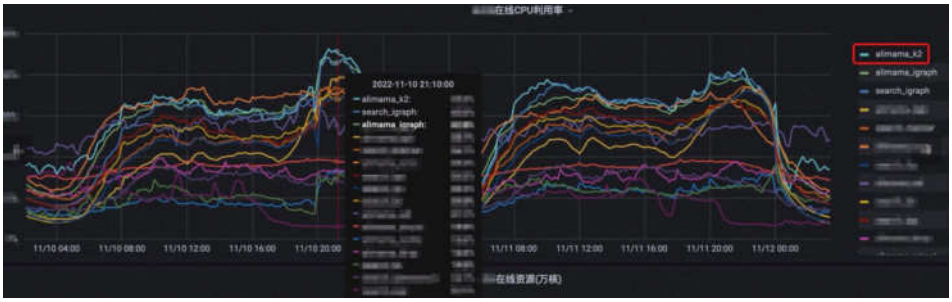
3）核心服务 cpu 回升



4）展示整体 cpu(20:01 后开启自动调 , 快速恢复 ,cpu3 分钟内完成快速拉升)



展示引擎 CPU 使用率长时间排名前列。



说明：该功能只针对流量大边际收益低的场景，边际收益高的场景完全自动调控。

1.2.2 定时设置档位上限

功能描述：为避免快速恢复模式下，部分模块出现超时率毛刺，将边际收益低的场景和模块，在快恢阶段设置档位上限 (设置了档位上限，但仍然会根据目标自动调整，所以会在局部波动)。实战情况如下：



图 20　20:00 附近购后粗排档位变化图

二、通用层

2.1 调控策略

1）dynamic container group regulator：动态容器分组调控器，支持通过 oops http 接口调整各分组机器比例，支持多目标设置，支持每个目标不同收敛区间。

2）multi goal negative feedback regulator：基于反馈的多目标流量调控器，支持多目标设置，支持不同机房设置不同目标。

2.2 metric 输入

新增 Metric 支持配置化接入，源头目前支持包括：黄金眼相关指标，blink 任务相关指标，tpp 相关指标，Eads 引擎 kmon 指标和 one-engine khronos 指标。

2.3 metric 处理策略

1）支持不同 metric 使用不同采集频率和采集时间段

2）均值策略，多路并发取最大均值策略等

2.4 灵活的定时功能

1）支持秒级别定时的功能，不同时间生效不同的 max 档位，不同的 fix 档位，不同的调控目标以及不同的档位模版。充分匹配广告业务场景的特点，如 0 点检索深度突增，大促整点流量突增等。

三、基础层

3.1 管控升级

1) 增加发布 diff 功能



图 21　版本对比功能

2）调控策略模版化

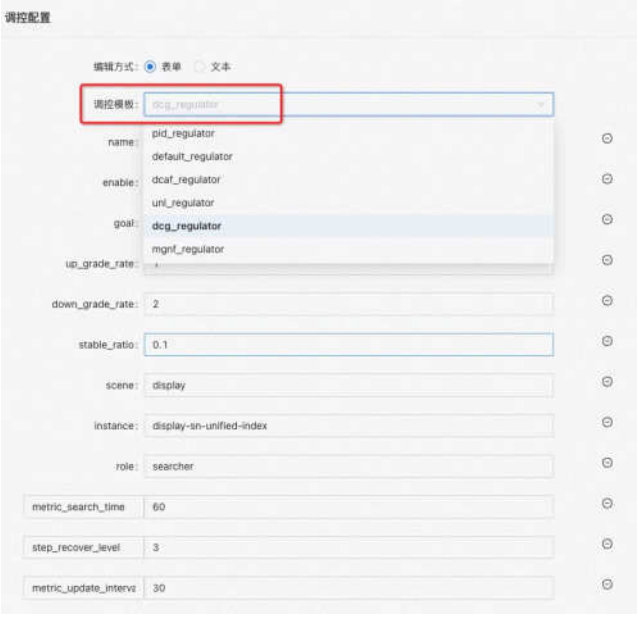


图 22　版本对比功能

3.2 视图升级

1）档位支持明细显示



图 23　档位明细图

2）档位历史曲线

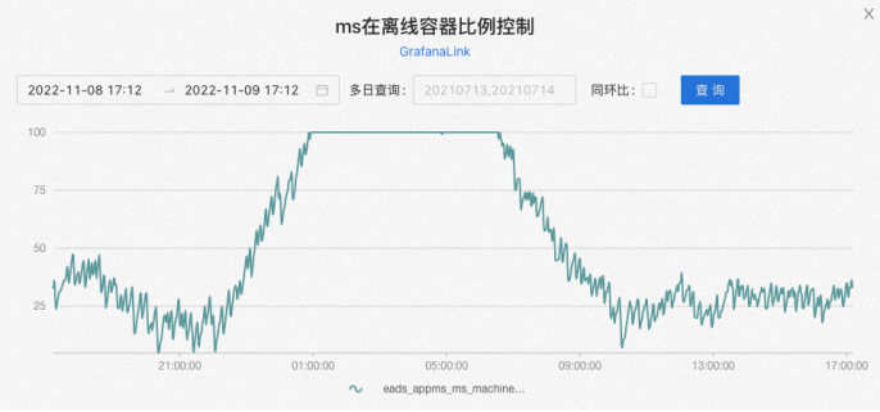


图 24　档位历史曲线图

3）支持档位快照功能，用于记录档位数据



图 25　档位快照功能

四、新的接入方式

4.1 整体介绍

下图蓝色部分是动态算力系统组成部分，绿色部分是业务方需要接入动态算力的服务。

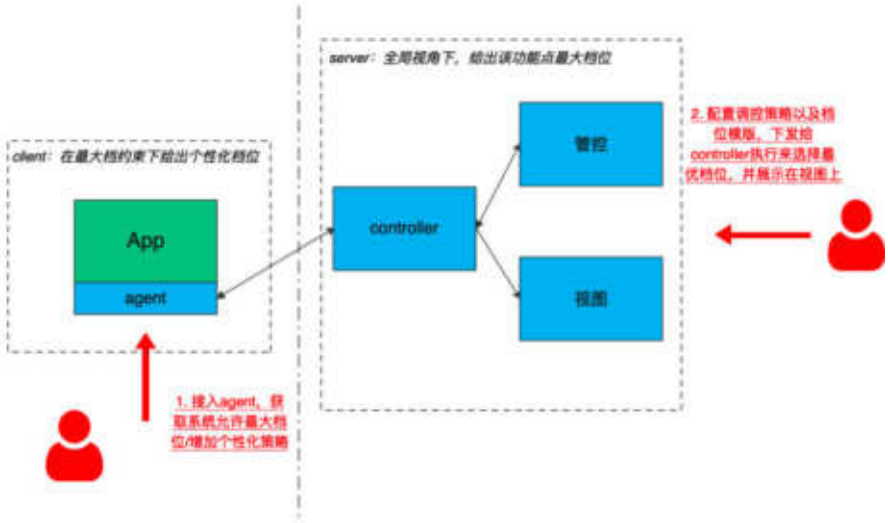


图 26　业务 App 接入

动态算力系统分两部分，分别是 client 端和 server 端。

4.1.1 client 端 (agent)

以 sdk 形式提供给接入方 (支持 c++,java)，主要负责跟 controller 交互，获取当前调控点 (由 controller 计算的) 最大档位，支持本地个性化调控策略 (个性化算力 dcaf，qscore 以及用户自定义策略实现不同用户不同档位)。

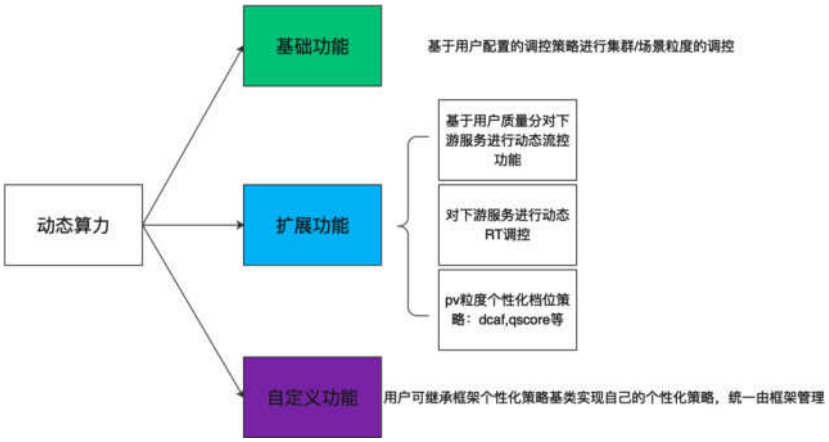


图 27　功能集合

4.1.2 server 端 (controller + 管控 + 视图)

server 负责接收用户在管控上选择的调控策略以及档位模版，并实时运行选择最优档位 (当前算力资源下该调控点整体的档位)，将档位数据下发给 agent，另外视图负责展示整个调控状态。

4.2 EADS 服务的接入

动态算力的 Agent 已经合并到 Eads 框架中，相比之前接入更加简便，并且配有详细的接入手册。

项名	旧版	新版
配置文件	两个	无
机房zk配置	明确指定	通过环境变量自动渲染
ID配置	每个应用需要单独申请APPID	同业务线所有服务共享APPID
参数获取代码	8行	1行
参数传递	不支持	支持op间和服务间传参
流控&RT控制	需要6行代码调用	配置化

表 1　agent 新旧版本接入对比

五、总结与展望

经过几年的发展，动态算力在算力分配策略上更加全面，在度量上也逐步精细化，应用的场景也渐多元化。相应的应用则可以总结为下图：



图 28　动态算力的应用

通过抽象封装，将常用的算力分配能力功能化，再加上轻简的使用方式，目前动态算力已具备平台化接入的能力。不过动态算力技术虽然已初成体系，但其离真正实现全局（跨场景，跨业务线，跨模块）最优算力分配和智慧引擎还有很长的一段路要走。未来希望继续探索算力、容量和效果之间的关系，尝试更优的决策算法进行算力的分配。另外在技术的通用性上，我们也会持续优化，进一步提升用户体验。

绿色智能技术方向上的探索在业界已然兴起，如何把有限的算力资源用好，利用通用的效率模型和体系化的方案解决好这个问题是一件非常有挑战且有价值的事情。而动态算力始终朝着让系统更加稳定，更加高效和智能的方向持续演进，期待未来在线引擎能够像变形金刚一样动起来。

关于我们：我们是阿里妈妈工程平台 – 引擎服务团队，致力于打造绿色智能的在线引擎。动态算力是一个共建项目，它是在各兄弟团队的通力合作下不断成长起来的，包括广告算法，业务引擎，引擎平台以及技术质量等团队。

参考

[1] 动态算力起源 – 阿里妈妈展示广告引擎的“柔性”变形之路：https://www.infoq.cn/article/wEaIlWI7076Id2bQptAq
[2] 阿里妈妈展示广告引擎新探索：迈向全局最优算力分配
[3] 阿里妈妈展示广告引擎动态算力再探索：面向业务收益的机器自适应调配

阿里妈妈智能诊断工程能力建设

作者：茂道、羲洋、君之、天柏

1. 业务背景

算法同学在日常工作中经常要面临一些耗时较多的临时工单，这类工单的问题类型五花八门，背后对应的原因也各不相同，例如广告主操作类问题、大盘流量波动问题、海选问题、粗排问题等。这类 Case 每次都需要耗费较长时间单独解决，没有办法沉淀相应的工具和知识体系，随之带来的是算法团队开发诊断代码工作量大、开发周期长、不宜维护等问题。

为了有效地持续提升工单处理效率，算法同学希望可以通过简易服务化方式，通过数据 + 指标 + 规则 + 服务化模式**快速配置** SOP 诊断链路，提升**自动化诊断能力**，最大化提升排查效率，并沉淀业务知识库，加快后续相似问题的诊断和响应速度。未来更期望将自动化诊断能力进行服务环节前置，如 xspace、袋小蜜，直接赋能广告主，并基于诊断和建设能力，升级营销诊断和袋小蜜等。因此，商家端智能诊断系统应运而生。



客服体验问题的解决思路

2. 建设目标

1）框架建设：希望通过 SQL 或者规则引擎 + 低代码方式提升业务侧迭代效率，提升算法同学的开发效率，迭代平台侧的的能力，沉淀组件和相关工程能力，整体工程资

源和数据资源可控。

2）数据建设：目前 replay 数据、操作日志、效果数据均由算法维护或者算法可灵活获取，从算法角度可以建立完善的广告诊断数据中心，标准化数据存储方式和数据获取方式。

3）诊断规则知识库建设：通过建设示范性 SOP，可以沉淀出大量可用的基础广告诊断规则（例如广告主余额不足、计划是否下线等）。同一数据诊断规则可以复用在多个 SOP 流程当中，用户可以基于诊断规则知识库搭建出复杂的广告诊断链路。

3. 技术方案

基于商家端引擎，我们开发了一套基于商家端框架的自动化 SOP 框架，将用户策略收口到商家端框架，可针对数据存储 & 管理、sop 微服务开发 & 测试 & 监控制定标准，提升开发和迭代效率；建立数据、指标、规则、服务的统一开发标准，依托 Dolphin 数据湖的能力进行数据存储和查询，规则引擎支持规则输入和 SOP 诊断链路编排，支持用户自助迭代。利用工程团队的技术优势，提升开发和迭代效率。

- **标准化**：数据存取标准化；服务接口标准化；核心算子 SQL 化、函数化；
- **可扩展**：数据的扩展性；规则的扩展性；
- **少代码**：数据读取高度抽象（SQL 化）、流程模块化 / 函数化、SOP 链路可视化配置；
- **可跟踪**：日志、监控；



商家端诊断引擎框架图

3. 规则引擎

- **标准化客诉流程：**通过规则引擎促使各个业务线的 SOP 流程统一为标准流程，遵循统一的 guideline。将 SOP 流程抽象为依赖什么数据，进行怎样的规则判断，得出怎样的结论，无法排查时应根据什么条件转交给不同的团队继续进行跟进。
- **为什么要引入规则引擎？**将业务决策逻辑从系统逻辑中抽离出来，降低系统间的耦合，使两种逻辑可以独立于彼此而变化，降低维护两种逻辑的维护成本，不同 sop 链路可以复用规则。
- **不同用户的使用场景不同，代码能力不同。**

3.1 引擎选型

引擎名称	场景	优点	缺点
Drools	1、业务代码和业务规则分离 2、适用于大型应用系统	1、功能完善 2、具有监控 3、操作平台等功能 4、支持DMN 5、文档完善	1、学习成本高 2、比较重，复杂度高 3、独立系统很难进行二次开发 4、以内存实现时间窗功能，无法支持较长跨度的时间窗
EasyRules	1、业务代码和业务规则分离 2、适用于大型应用系统	1、 轻量 2、 易学 3、支持复合规则 4、定义规则方式多样 5、支持复杂业务场景	1、不支持DMN 2、活跃度较低 3、文档较少
uniRule	1、业务代码和业务规则分离 2、适用于大型应用系统	1、轻量，使用起来方便。 2、有可靠性 3、学习成本比较低	1、功能简单 2、技术支持较少
URule	1、业务代码和业务规则分离 2、适用于大型应用系统	1、功能完善 2、支持规则集，决策树，决策表 3、文档完备（有视频教程） 4、易集成 5、易改造	1、社区不活跃 2、应用案例较少 3、源码应用技术栈相对落后

智能诊断场景需求：轻量开发、定义规则方式多种、支持代码模式和表达式规则，最终选择选择轻量、高效的表达式和代码结合的引擎 easyRules。

3.2 规则

低代码平台针对不同的使用群体，划分为三种模式：

1）简单运算

通过四则运算和条件运算，产出规则，具体是 metric 和 metirc 之间的四则运算表达式 和 逻辑表达式，产出为 true 或者 false。

```
if(100.0 * (competition_times - adReplayInfo.getSn_real_competition_times())
   / competition_times ◇ )>80
```

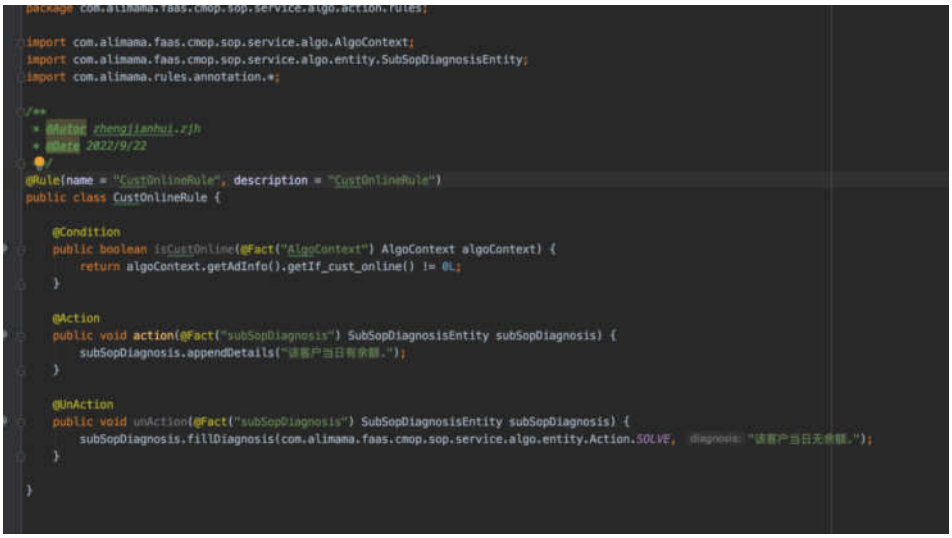
2）code 模式

对于具有复杂数据结构的运算，需要使用 code 模式，编写代码片段，使用 QLExpress 语法格式。

```
for (i = 0;i < bidwordList.size();i++) {
   bidword = bidwordList.get(i);
    keywordCateInfoEntity = keywordCatInfo.get(bidword);
    if (entity == null) {
        subSopDiagnosis.appendDetails(String.format("投诉词【%s】与 AD 的类目不符
.", bidword));
    } else if (!entity.getCate_map().containsKey(cate_id)) {
        subSopDiagnosis.appendDetails(String.format("投诉词【%s】与 AD 的类目不符
,推荐类目为【%s】.", bidword, entity.getMain_cate_full_name()));
    } else {
        goodWords.add(bidword);
        subSopDiagnosis.appendDetails(String.format("投诉词【%s】与 AD 的类目相符
.", bidword));
    }
}
```

3）java 代码

直接由用户编写 rule class，运行时动态加载。

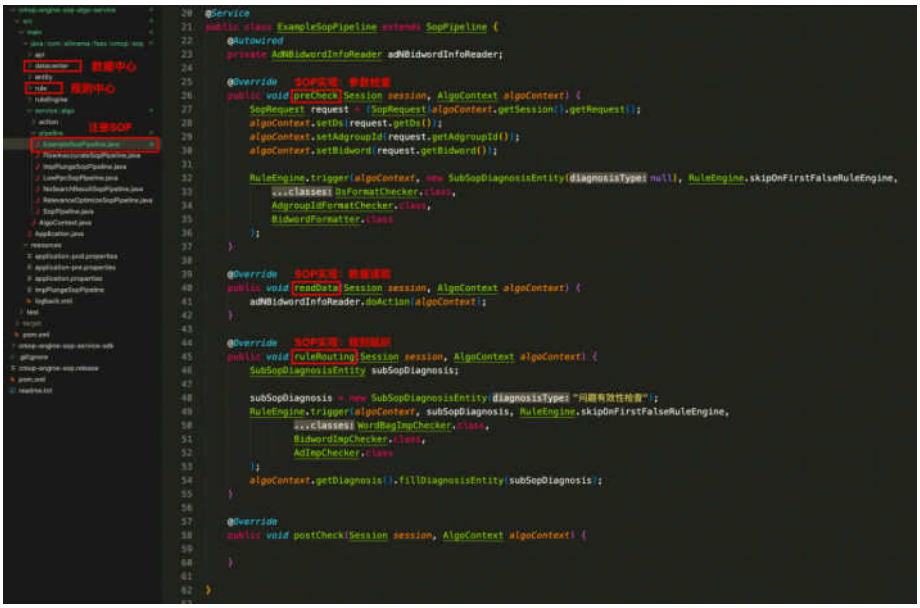


用户自定义规则编写示例

3.3 SOP 链路

将 SOP 链路划分为**参数检查、数据读取、规则组织、结果返回**，在不同阶段绑定不同规则集，进行诊断链路串联。

- **参数检查：**输入粒度规范、日期检查、输入关键词检查等；
- **数据读取：**指标读取、衍生指标读取等；
- **规则组织：**规则集绑定；
- **结果返回**：诊断结果、诊断话术。



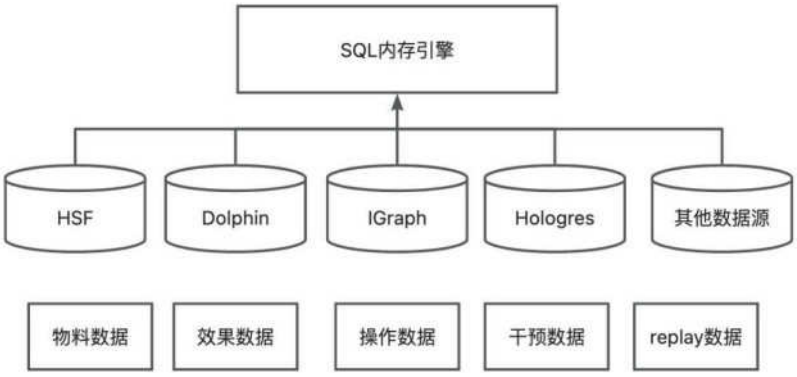
诊断链路编写示例

4. 数据中心

4.1 SQL 统一查询

同步查询

统一的 SQL 查询引擎对外提供统一的 SQL 语法（语法和 PG 语法保持一致），实现对 Dolphin、IGraph、Hologres、Http、HSF 等数据源的统一查询功能，同时也支持跨引擎查询功能，极大降低了用户的使用成本，使用存储在不同数据源的数据就像在 ODPS 上一样方便。



SQL 统一查询引擎框架图

异步查询

对于数据量特别大、查询模式支持异步化查询；

4.2 数据建设

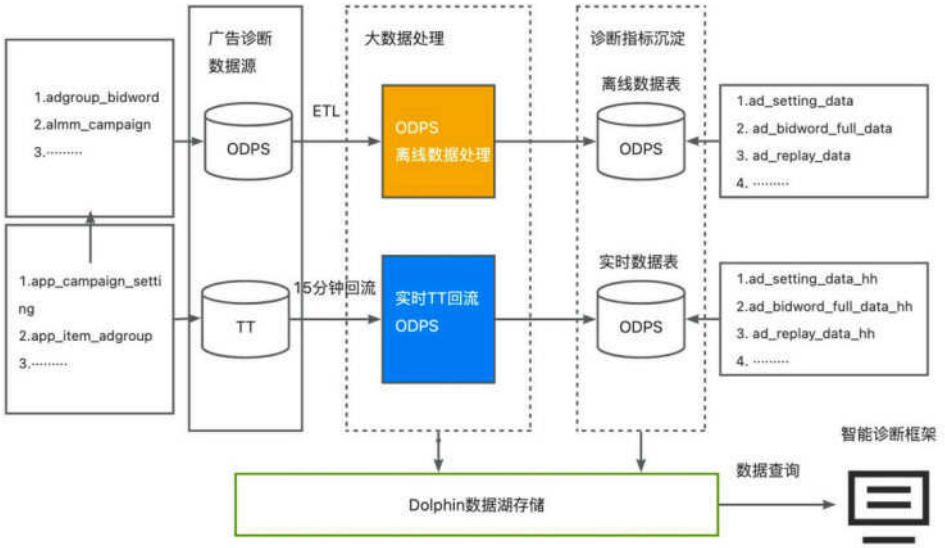
广告诊断系统数据可以划分为 6 大类：物料数据、效果数据、投放数据、干预数据、操作数据和人群数据。各类数据分别来自 BP、SDS、投放引擎、算法维护、操作日志、达摩盘等，数据存在分散、存储引擎多种多样、数据格式不统一、维护团队较多、部分数据实效性较差等问题，整合和产出一份格式统一、数据完备的广告诊断数据仓库十分必要。对 tickets 工单系统中的客诉工单进行统计，诊断所需要的数据粒度也分为 AD 粒度、关键词粒度、人群粒度和 QUERY 粒度。



广告诊断数据大图

实时、离线数据建设

- 离线数据：数据 ETL 产出 ODPS，通过极光平台 [2] 导入 Dolphin。
- 实时数据产出：TT 数据流回流 ODPS，通过极光平台 [2] 导入 Dolphin，产出小时级数据。



实时离线数据建设链路图

将存储在不同数据源的数据在平台进行注册，将诊断数据统一化为数据指标类型，用户可以在数据层定义衍生指标，支持四则运算以及复杂函数类型计算，沉淀出数据层的指标和衍生指标。

- 支持同一数据粒度下的多指标运算
- 生命周期管理
- 有效性巡检
- 指标数据探查

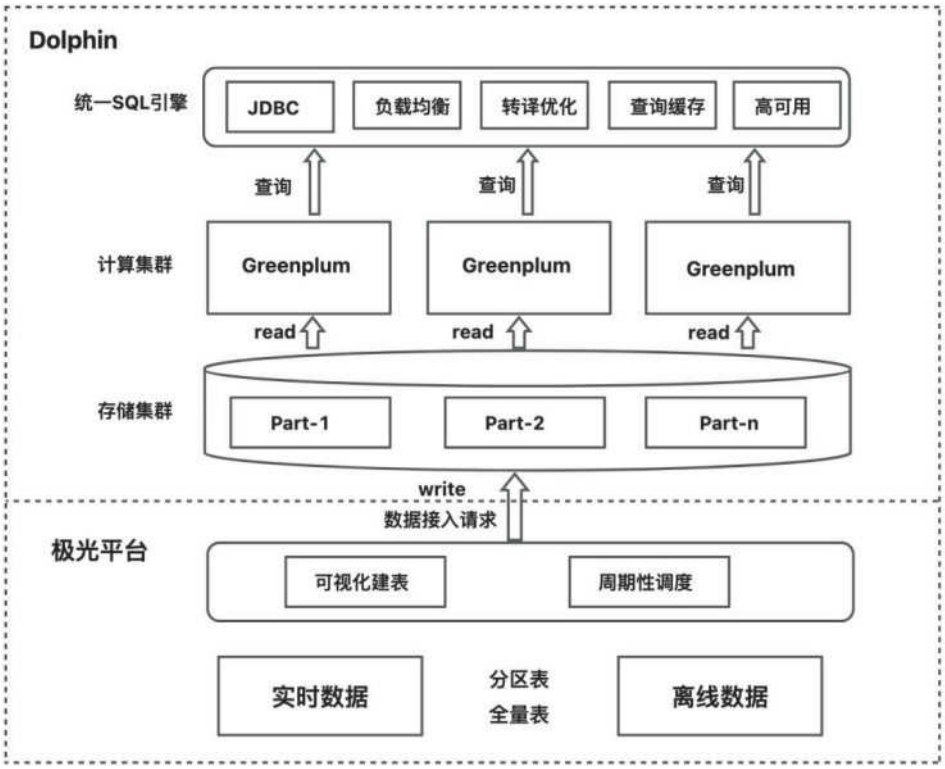
5. 加速引擎

广告诊断场景的查询具有**查询 QPS 低、查询周期长、单值查询**的特点，诊断场景对于查询延迟要求不高，实时数据与离线数据存储量较大，存储成本较高。针对这样的场景，我们通过数据库的外表技术，可以实现计算引加直接读取 HDFS 上的数据，多个计算集群，可以共享一份 HDFS 上的文件存储数据，实现数据的一写多读，HDFS 底层使用 HDD 存储数据，实现数据**成本**的大幅度**降低**，数据统一存储。

为了支持外表数据，索引下推查询，我们对 orc 代码数据读取部分做了大量深入优化，包括排序列的智能选择，动态 row group stripe size 和索引和数据的本地 cache。在直通车展现波动 SOP 场景中，90 天长周期 T 级别数据量查询可以达到

秒级延迟；在人群 SOP 场景中，Dolphin 在标签圈人百亿 ID 的交并差领先的技术优势能够很好地体现出来。

ODPS 表数据也可以直接通过极光平台 [2] 进行查询加速，极光平台 [2] 同时也支持原生 SQL 对各种数据源进行查询。



数据加速引擎架构图

6. 总结与展望

当前在直通车业务场景下，商家端智能诊断系统可以支持展现波动、流量不精准以及搜索无展现、相关性优化 SOP。展现波动 / 无展现 SOP 中 49% 给出诊断，流量不精准可 100% 给出建议，相关性优化可 93% 给出建议；覆盖直通车推广优化类工单 65%，准确率 >90%（推广优化类工单主要由技术同学负责解决，最复杂难解且数量多；覆盖率指能给出明确诊断或建议的比率），面向不同子问题的完结率和处理时长均有显著提升。

智能诊断系统由数据引擎团队与广告主赋能算法团队共建，希望可以通过低代码的智

能系统更好地帮助算法侧沉淀营销知识库，真正从平台视角来帮助广告主解决营销投放中可能存在的问题，同时也可以更及时地排查系统中的潜在风险。后续我们希望通过业务场景拓展，推向更多的工单类型，同时将服务环节前置，从根源上释放算法同学的压力，同时更好地赋能广告主；并基于诊断和建议能力，升级营销诊断和袋小蜜等。未来可以基于多维时序数据进行归因分析辅助诊断，在广告主进行提出工单之前就将问题定位，提高平台的智能性。

7. 附录

[1] **Dolphin**：面向营销场景超融合智能引擎，Dolphin 源自阿里妈妈数据营销平台达摩盘 (DMP) 场景，在通用 OLAP MPP 计算框架的基础上，针对营销场景的典型计算（标签圈人，洞察分析）等，进行了大量存储、索引和计算算子级别的性能优化，实现了在计算性能，存储成本，稳定性等各个方面的大幅度的提升。Dolphin 引擎作为商家端服务的核心基建，可以横向覆盖交互式 OLAP 分析，AI 算法计算，Streaming，Batch 等多个计算场景。

[2] **极光平台**：针对 B 端商家的统一服务框架和研发平台，主要有以下功能：1）统一服务框架：FAAS 函数服务，算法同学开发业务核心代码，工程团队负责基础功能，实现服务的快速开发，迭代，发布上线及低成本运维；2）统一研发平台：支持商家端算法特征的开发，管理，沉淀。模型管理、发布上线；3）统一计算引擎：算法核心通用算子下沉 Dolphin 引擎，实现业务逻辑和底层计算解耦，算子复用。

广告深度学习计算：向量召回索引的演进以及工程实现

作者：无蹄

问题定义

召回操作通常作为搜索 / 推荐 / 广告的第一个阶段，主要目的是从巨大的候选集中选出相对合适的条目，交由后链路的排序等流程进行进一步的择优。因此，召回问题本质上就是一个大规模的最值的搜索问题：

对于评分 $f()$ 和候选集 Z，给定输入 x，寻找固定大小的 Z 的子集 Y，使得 $\{f(x,y)\ ,\ y \in Y\}$ 在 $\{f(x,z),\ z \in Z\}$ 尽可能靠前。

这个问题本身非常简单直接，而其核心在于其巨大的规模。通常，在阿里妈妈广告场景中，召回的候选集规模能达到千万甚至是上亿的数量级。受在线系统较为严格的延时约束，在这样规模的候选集上通过暴力搜索找出最优解显然是不现实的。

因此，我们会对候选集 Z 构建索引，从而对搜索过程进行剪枝。尽管某些索引会导致检索出的结果并非最优解，但好在召回问题本身对精确度的 trade off 容忍度较大。首先，搜索 / 推荐 / 广告本身就是非确定性的，其最优解并不能严格定义；其次，后链路还会进行进一步择优，召回结果只要做到尽可能包含后链路的最优解，也就是保证召回率即可。

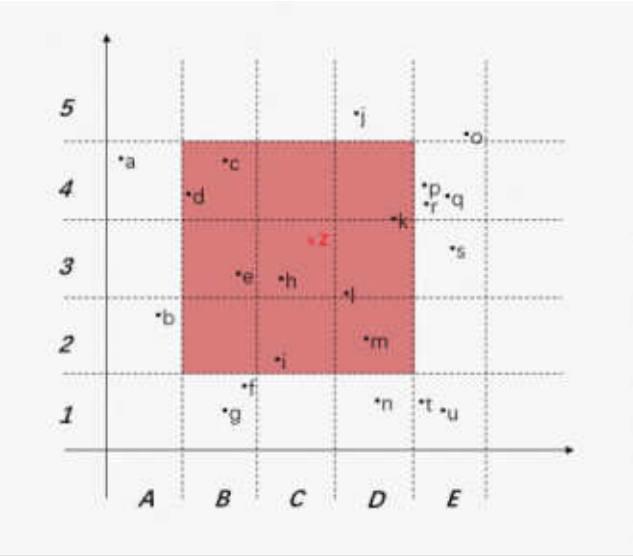
本文将介绍阿里妈妈广告召回索引的演进过程，以及在这一过程中的工程解决方案。

1. 分区索引（量化索引）

这种构建索引的方式是最直接的。简单来说，就是在每个维度上进行量化，从而把整个向量空间划分成若干个不同的分区。这样可以将向量划归到不同的分区，在需要进行剪枝时，可以先根据分区进行筛选：首先粗略地挑选出若干分区，再将归属于这几个分区的向量挑选出来进行详细的计算与排序。由此可以避免计算候选集中的全部向量。

如下图所示：对于寻找最近欧几里得距离向量的问题，在一个二维的向量空间上，我们对每个维度量进行 5 等分的量化，从而将整个空间划分成 25 个分区。对于输入向

量 z，我们可以首先筛选出分区 {B2, B3, B4, C2, C3, C4, D2, D3, D4}，然后对这几个分区中的 {c, d, e, h, i, k, l, m} 这 8 个向量进行距离计算和排序，相比于全局的比较搜索的 21 个向量，计算量缩小了 2.6 倍。

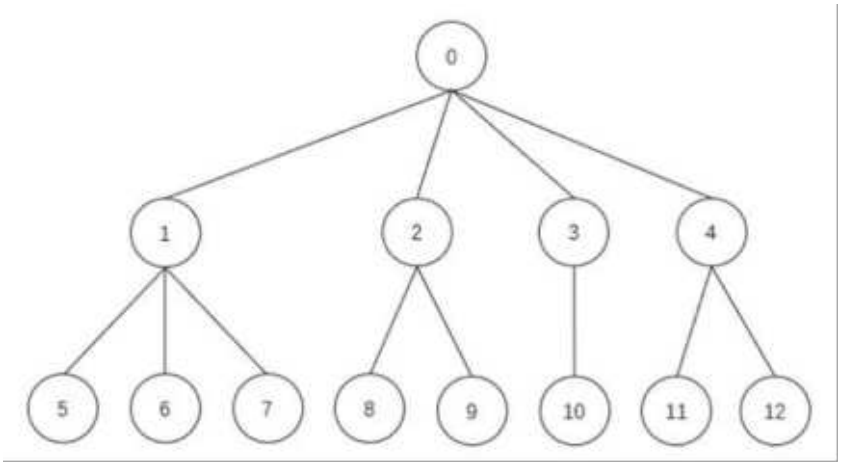


这种量化索引被广泛运用于传统的向量检索引擎，例如 faiss/proxima。这种索引对于平均分布的低维向量上针对距离的检索效果很好，但同时存在几个问题。首先是对于非平均分布的向量集，可能需要非均匀的量化策略，一方面增加了复杂性，一方面也未必有很好的效果；其次，对于比距离函数更复杂的评分函数，很难找到分区于函数结果之间的相关关系，很难做剪枝算法；最后，对于高维的向量，还存在高维稀疏的问题，导致分区数量远远大于实际向量的个数。举个例子，我们 NANN 模型中广告向量位 128 维，即使我们每个维度只进行 2 分，就有 2^128 个分区，远大于数百万的向量个数，导致其实很多分区里是没有向量的。为了解决这个问题需要对向量空间进行降秩，而降秩算法本身又引入了新的更复杂的问题。

2. 树状索引

为了规避量化索引的种种弊端，我们在 TDM 模型中采取用了树状索引。量化方法最主要的思想是要分类（分治），而它的弊端主要是来自于天然向量空间的限制。为了突破这些限制，我们可以更加“人工”的进行分类，而不是单纯以空间相对位置作为分类基准。这里我们就可以引入各种聚类算法，对向量进行聚类，并且可以在此基础上对聚类进行聚类（类似 linkage 算法）。由此，我们会获得一个树结构的索引。这

棵树上，叶子节点代表向量，上层节点表示聚类（或聚类的聚类 etc）。而检索的过程，就是在这颗索引树上从上到下进行 beam search。这种类型的索引，我们最早是在 TDM 模型中进行了运用，参考论文《Learning Tree-based Deep Model for Recommender Systems》。如下图所示，图中节点 5~12 表示向量本身，节点 1~4 则表示聚类。



在工程实践中，最早的 TDM 采用的“二叉树”的索引结构，深度大约 6~7 层（最上层数千个节点），且这个索引结构维护在引擎端，每一次 inference 都将导致若干次的 RPC，导致 RT 压力很大。后期改进中，我们首先将二叉树变成了多叉树，减少了树的深度。其次将索引结构用 tf 的 splits 结构表达，放到了模型中，在模型中计算，避免 RPC 开销，为模型增加的数倍的 RT 空间。这部分的详细叙述可以参考我们之前的文章《广告深度学习计算：召回算法和工程协同优化的若干经验》。

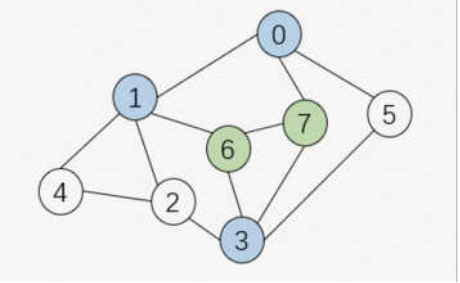
这种**树结构的索引有一个弊端：它的不同分类之间是互斥的，**也就是说它能表达的向量之间关系比较单一。对于电商，每个商品可能会有不同的属性，每种属性都可能和别的商品产生某种关联关系。对于多种关联关系的表达，树结构索引就显得有些力不从心了。

3. 图索引（HNSW）

为了解决树结构索引聚类之间互斥带来的问题，我们引入了图结构的索引 HNSW。与树结构索引用父节点表示聚类不同，HNSW 中每个节点都表示向量，而用编来表示向量间的关联。与树结构索引类似，在 HNSW 索引上的检索也是一个 beam

search 的过程，不同的树结构索引通常是从第一层开始检索，而 HNSW 上会随机选取节点作为检索的起点。此类索引我们运用在了 NANN 模型中，参考论文《Approximate Nearest Neighbor Search under Neural Similarity Metric for Large-Scale Recommendation》。

在 TF 中表达图的结构比树的结构更为复杂，这里我们采用了 ragged tensor 数据结构。ragged tensor 是 TF 原生支持的，可以表示一个行之间不等长的矩阵，我们用它来表达 HNSW 的邻接表，在上面执行 gather+unique 操作就可以表示 beam search 中 expand 的操作。例子如下：



对于这张图的结构，它的邻接表：

```
0: 1,5,7
1: 0,2,4,6
2: 1,3,4
3: 2,5,6,7
4: 1,2
5: 0,3
6: 1,3,7
7: 0,3,6
```

可以表达成ragged_tensor: [[1, 5, 7], [0, 2, 4, 6], [1, 3, 4], [2, 5, 6, 7], [1, 2], [0, 3], [1, 3, 7], [0, 3, 6]]，如果需要从 {6,7} 节点进行 expand，只需计算 unique(gather([[1, 5, 7], [0, 2, 4, 6], [1, 3, 4], [2, 5, 6, 7], [1, 2], [0, 3], [1, 3, 7], [0, 3, 6]], [6, 7])) = {0, 1, 3, 6, 7} 即可。

相比于树，图结构的索引由于并没有互斥关系，导致在 beam search 的过程中我们可能会重复访问某些节点，造成了计算资源和召回配额的浪费。因此我们在 beam search 检索过程中需要排除掉已访问节点，也就是维护一个访问节点的集合，通过 set difference 操作来进行去重 / 剪枝。对于上述的例子，这个操作意味着

{0,1,3,6,7}−{6,7} = {0,1,3}。

由于 naive 的 set difference 计算效率不高，我们采用了在全集上建立一个 bitmap 的做法：每个节点用一个 bit 标志是否被访问过，并在整个 beam search 的过程中都维护这一个 bitmap，将集合操作转变成简单的位操作。同时，为了避免 op 内 input/output 时的 memcpy，我们用 tf 里的 ref tensor（即 variable）来存储这个 bitmap，同时通过 temporary_variable 中 step_container 的机制来规避多并发时线程不安全的问题。

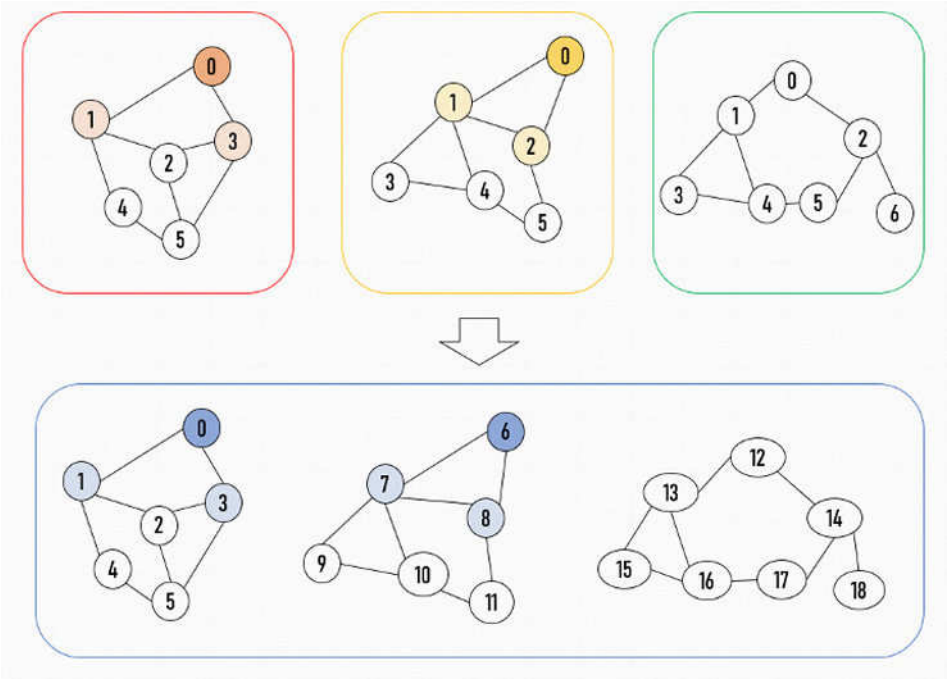
```
input  bitmap(temporary_variable),  nodes(vec_tensor)
result  =  []
for  n  in  nodes:
    if  bitmap[n]  ==  false:
        result.append[n]
        bitmap[n]  =  true
return  result(vec_tensor)
```

4. 多类目 + 多层次的 HNSW 图索引

随着 hnsw 图结构索引推广到更多的业务，我们也遇到了新的问题。在搜索广告场景中，业务上需要对商品类目进行强感知，这些类目之间并非互斥而还存在层级关系，而且大小分布极不均衡。算法在建模的过程中需要对每个类目单独构建 hnsw 索引，每次请求需要在若干被选中的类目上都进行 beam search 的检索。这对我们的系统提出的挑战是：如果我们在这些类目上依次进行独立的 beam search，整个检索过程将被重复数次，延时上必然满足不了要求。因此，我们需要某种形式的并行化，进行缩短整个检索的流程。

我们采用了 indirect（间接）的思想来处理这个问题。我们将多个类目上的 hnsw 图都统合起来，包括多层级的类目，都视作同一张 hnsw 大图，在这张大图上进行整体的计算。不同于上一节中的 hnsw 图，这张大图并不是完全联通的，而是由几个互相隔离的子图组成。而且在这张大图上，节点和广告并不是一对一的关系，而是多对一，即若干不同的节点会对应同一个广告，因此我们还会维护节点到广告的映射关系。对于每一次请求，我们会将 beam search 的起始点定在那些被选中类目对应的子图上，子图间的隔离将保证一个类目上的检索不会串到别的类目上。同时在每一轮 topk 的过程中我们会维护访问节点与不同子图计算结果的隔离关系，从而确保每个类目上的

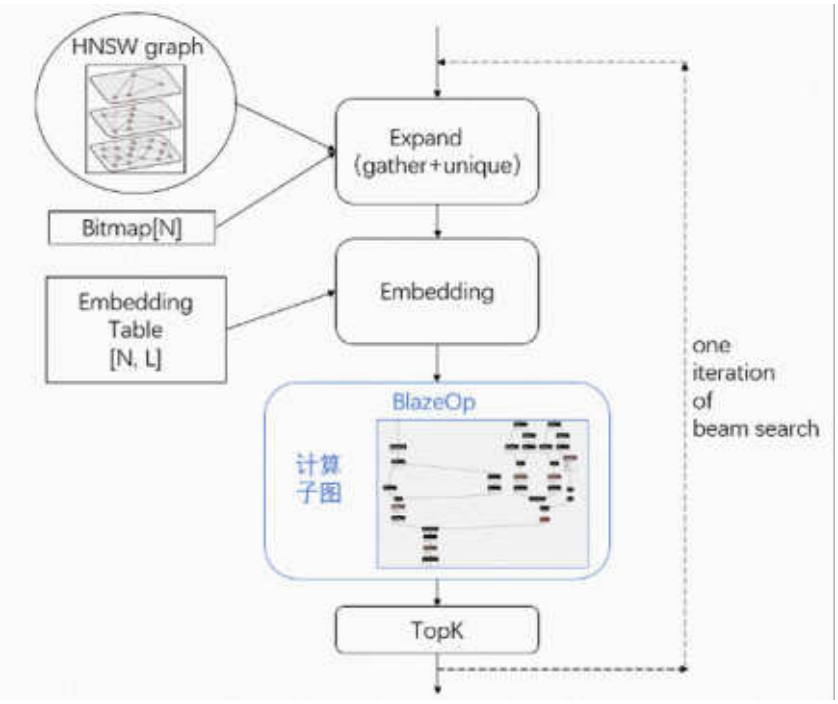
配额足够，保证类目的完整性。而这种隔离关系正好也可用通过 ragged tensor 来进行表达。



如图，我们通过对节点重新编号，将 3 个互相独立的图结构索引整合进同一张大图，三个子图仍然互不连通。对于其中独立索引上的两个独立的 beam search 过程：[0]->[1,3] 和 [0]->[1,2] ，而在新的大图上，就被整合进了同一个 beam search 流程：[[0],[6]]->[[1,3],[7,8]]。图中的 3 和 7 尽管是不同节点，但仍然可能表示同一个商品 / 广告 / 创意之类的对象，从而映射到同一个向量上。

5. 结合图化的创新

我们在图化模型服务引擎中提供了 BlazeOp 的抽象，将 tf session run 的整个过程封装进了 BlazeOp 中。由此，我们可以将模型运算图的一部分子图打包进 BlazeOp 所持有的独立 session 中。结合这一特性，我们对树结构索引和图结构索引这类 beam search 的模型进行了升级改造，将打分函数部分打包进 BlazeOp，从而实现了模型计算与检索过程分离。



这样做带来了以下优势：

1. **形成固定范式，方便算法迭代**。算法在更新打分模型的时候不需要去考虑索引结构，可以直接复用索引部分的代码，反之亦然。例如，可以直接通过替换 BlazeOp 中子图的方式将 A 模型中的打分函数运用到 B 模型中去。
2. **更易于进行打分模型的交付和优化**。首先，打分部分模型可以直接复用离线训练的图结构进行原生交付；其次，由于打分模型走独立的 session，避免 beam search 带来的 dynamic shape 的问题，采用传统的 padding 方案即可开启 xla，不依赖 auto_padding。最后，由于摘除了巨大 embedding table const 和复杂的 beam search 逻辑，打分的子图相对全图简化了很多，因此能很方便的开启 const-folding 之类的图优化。
3. **可以绕开含光 npu 系统对于图中只能存在一个 engine op 的限制**：由于对 engine op 进行编译时采用了静态的内存地址分配策略，同一个图中如果包含多个 engine op 会导致内存地址冲突。而将 engine op 置于 blaze op 的 session 中，则可以保证每个 session 中只存在唯一的 engine op，避免了内存地址冲突。通过 beam search 中对这个 session 的多次调用，实现了在一次图执行过程中执行多次 engine op 的目的。

此外，图化模型服务引擎也便于我们为索引升级新的特性。例如，通过对索引部分的 embedding table 定制动态的数据结构以及实现相对应的功能 op，配合 swift 流等中间件，我们可以实现 embedding table 以及 hnsw 图结构的实时化。这部分内容，我们将在后续的文章中详细介绍。欢迎感兴趣的同学关注及留言讨论。

引用

[1] Zhu H, Li X, Zhang P, et al. Learning tree-based deep model for recommender systems[C]//Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. 2018: 1079-1088.

[2] Chen R, Liu B, Zhu H, et al. Approximate nearest neighbor search under neural similarity metric for large-scale recommendation[C]//Proceedings of the 31st ACM International Conference on Information & Knowledge Management. 2022: 3013-3022.

Dolphin：面向营销场景的超融合多模智能引擎

作者：陌奈、弥卢、冷川

1. 背景

为提升易用性和降低使用成本，大数据技术逐步向 Serverless、一体化和智能化方向发展。为了更好的提升客户投放效率和效果，阿里妈妈自研了超融合多模智能引擎 Dolphin（以下简称“Dolphin”）。其最初定位解决通用 OLAP（OLAP 全称 Online Analytical Processing）在圈人场景计算性能问题，历经 5 年的技术发展与沉淀，目前已形成覆盖 OLAP、AI、Streaming 和 Batch 四大方向的智能超融合引擎，提供针对营销场景投前、投中、投后全链路的广告主工具和算法策略迭代。本文结合近年来营销场景生态发展梳理了 Dolphin 引擎技术演进过程，欢迎阅读交流。



投放场景

Dolphin 通过统一开放的技术架构，提供智能超融合一体化使用体验，实现使用 Dolphin SQL 就可以连接异构计算和存储引擎，屏蔽复杂的底层技术，通过 Dolphin SQL 解决业务场景下各类问题，实现业务逻辑和底层技术解耦，提供一体化高效的开发能力。

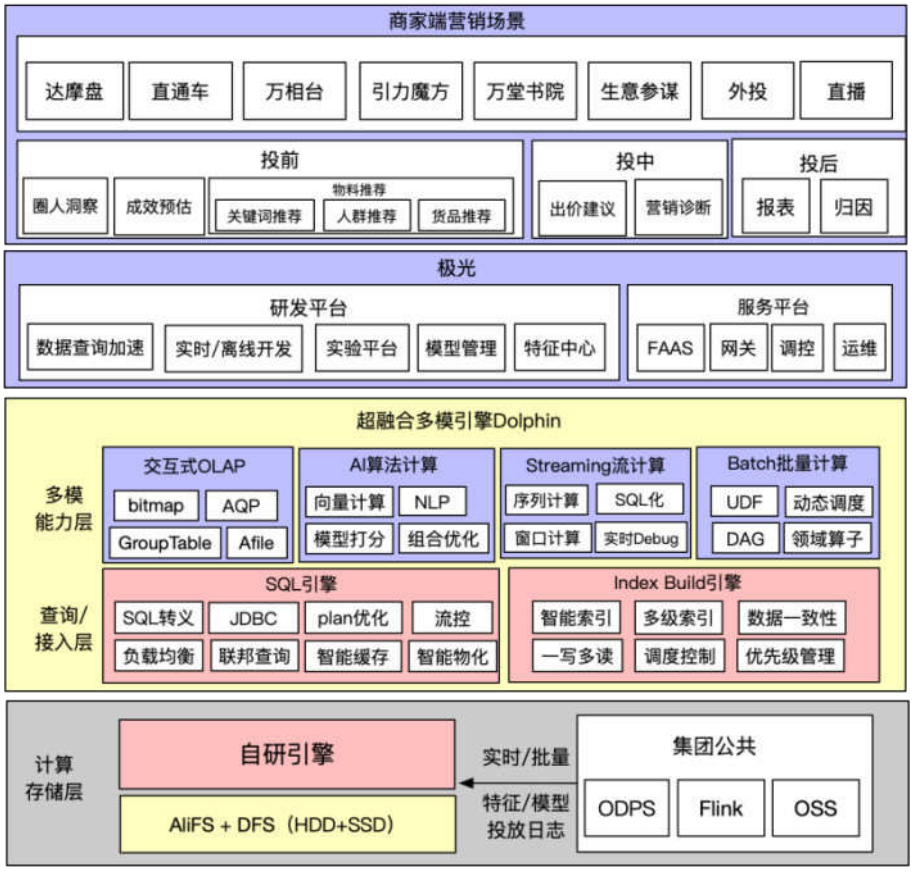
1.1 Dolphin 引擎技术业务大图

阿里妈妈营销场景复杂多样，包括达摩盘、搜索广告直通车和展示广告引力魔方等产品，用户规模和数据规模都是业界 Top 级别，Dolphin 引擎经过多年发展沉淀两方

面核心价值：

- **性能价值：**解决超大规模场景下通用引擎无法解决的性能问题。
- **效能价值：**降低通用引擎的使用成本甚至做到对用户透明无感知，从而提升开发迭代效能。

下图是 Dolphin 引擎相关技术和业务大图，主要分为商家端营销场景、极光（阿里妈妈交互式研发和服务平台）、Dolphin 引擎和计算存储层四个部分。



Dolphin 超融合多模引擎技术业务大图

2. Dolphin 内核能力

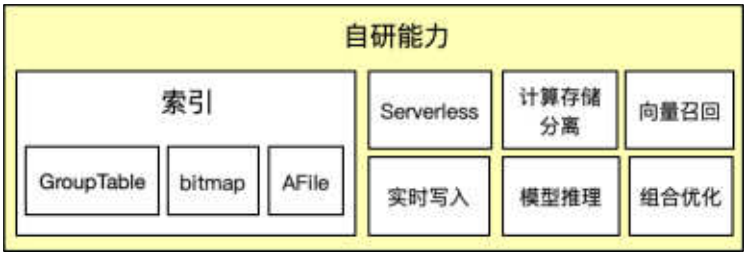
Dolphin 引擎最初源于 OLAP 计算场景，在 MPP 计算引擎基础之上构建，核心能力有五个方面，包括自研引擎、SQL 引擎模块、Index Build 引擎模块、智能计算和一写多读能力。

2.1 自研引擎

考虑到查询性能、稳定性和生态完善程度，我们选择自研一系列能力：

- 计算存储分离，基于集团云基础设施对计算存储解耦，实现云化部署，对计算存储动态管理。
- 支持 bitmap、GroupTable 和 AFile 索引，对超大规模数据计算性能加速。
- 支持向量召回计算，支持高并发、高性能向量在线计算和离线批量计算。
- 支持模型推理打分，支持部署 AI 模型进行高并发、高性能在线推理打分计算。
- 支持实时写入，支持高性能实时数据写入能力。

基于引擎自研的能力，不仅扩展引擎能力边界，更减少多引擎开发及维护成本。

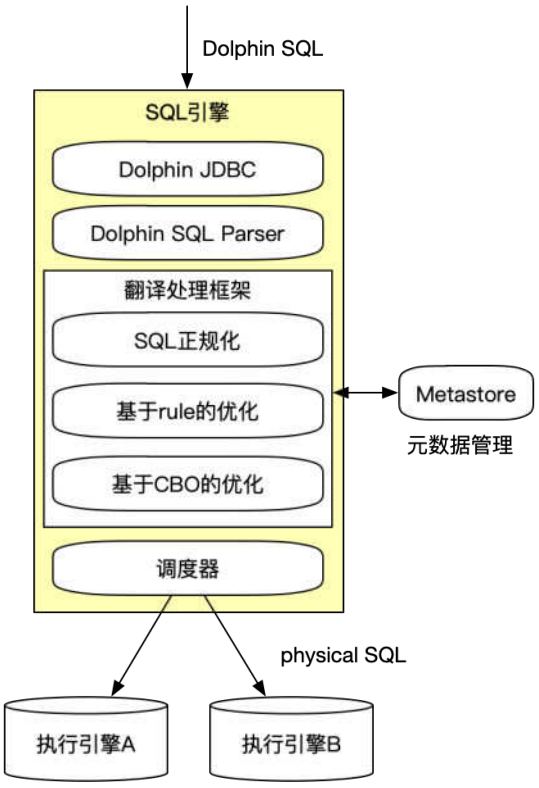


2.2 Dolphin SQL 引擎

SQL 引擎核心设计目标是解耦**业务 SQL** 和**物理执行 SQ**L，通过 SQL 转译让业务 SQL 转化为物理执行 SQL，用户对底层透明，给底层提供非常大的技术扩展空间，让使用 SQL 开发 AI、Streaming 能力成为可能。SQL 引擎的能力主要包括转译、执行计划优化、负载均衡、物化及联邦查询能力。

我们自研 SQL 编译器，实现让复杂 Dolphin SQL 转译为底层物理执行 SQL，这里主要分为 3 个部分：

- **Dolphin JDBC：**主要 SQL 服务入口，为了实现更接近数据库的使用体验，我们实现 Dolphin JDBC 驱动，让业务对接相对 RPC 服务也更简单。
- **Dolphin SQL Parser：**使用性能更高的 fastsql，将 SQL 解析为 AST 树。
- **翻译处理框架：**负责对 AST 数进行分析，包括检查元数据、绑定物理信息、基于规则 +CBO 的执行计划优化，生成 physical SQL 提交给执行引擎。
- **调度器：**根据物理执行计划决定哪部分算子需要在哪个引擎执行，控制联邦查询过程。



查询 SQL 转译过程

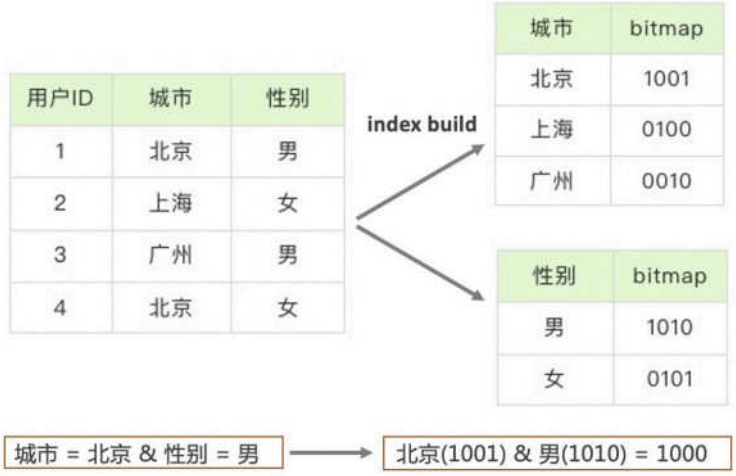
2.3 Dolphin Index Build 引擎

为解决通用引擎在超大规模圈人场景性能瓶颈，我们使用 Index Build 引擎实现对数据的索引构建，主要有 Bitmap、GroupTable 和 AFile 索引，然后将索引导入在线 MPP 引擎提供查询服务。

2.3.1 Bitmap 索引

我们在 18 年底启动使用 bitmap 计算升级方案，通过对原始数据进行 build 成 index 索引导入在线引擎，实现用户使用通用 SQL 就可以进行毫秒级圈人计算，新的 bitmap 索引实现部分场景数据压缩高达 50 倍，计算性能提升高达 15 倍，同时可以达到数百 QPS 毫秒级并发查询性能。

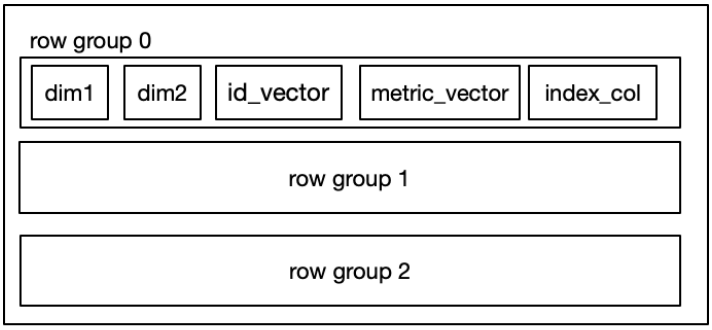
下图所示是把原始结构表转化为 bitmap 结构索引表，就把传统 scan+filter 计算转化为 bitmap 与运算，参与计算的数据量更小且可以利用 CPU 指令计算加速。



bitmap 计算案例

2.3.2 GroupTable 索引

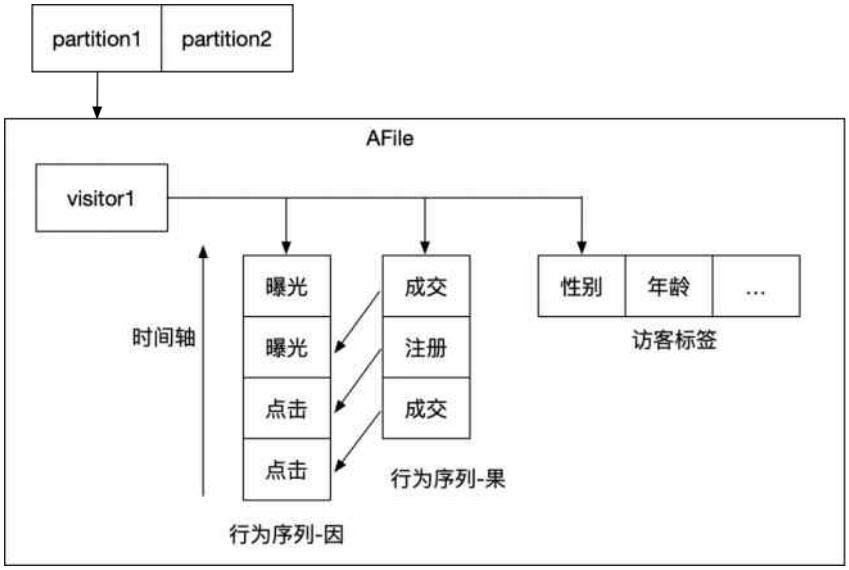
在达摩盘标签圈选场景中，我们使用 bitmap 加速标签圈选已取得显著业务收益，然而在用户 ID 粒度计算的场景（group by ID + having）及 ID join 场景则无法直接应用，这类场景的计算量通常非常大，单表容量可达到百 T 级别，通用引擎计算的时间和存储成本很大，基本很难支持大规模场景，于是我们提出 GroupTable 索引结构，使用向量化 + 索引 + 压缩的方式进行存储，实现部分场景数据存储节省高达 60%+，计算性能提升 30 倍 +，该结构也可以跟 bitmap 结构进行混合计算，基本覆盖大部分计算场景。



GroupTable结构

2.3.3 AFile 索引

在超大数据规模的用户行为分析和广告归因分析场景，因为涉及数据 join 关联，通用计算引擎很难达到秒级交互式查询性能，故提出 AFile 索引结构，通过预计算的方式提前做好数据处理，从而达到计算性能提升 1 到 2 个数量级效果。



AFile 结构

2.4 智能计算能力

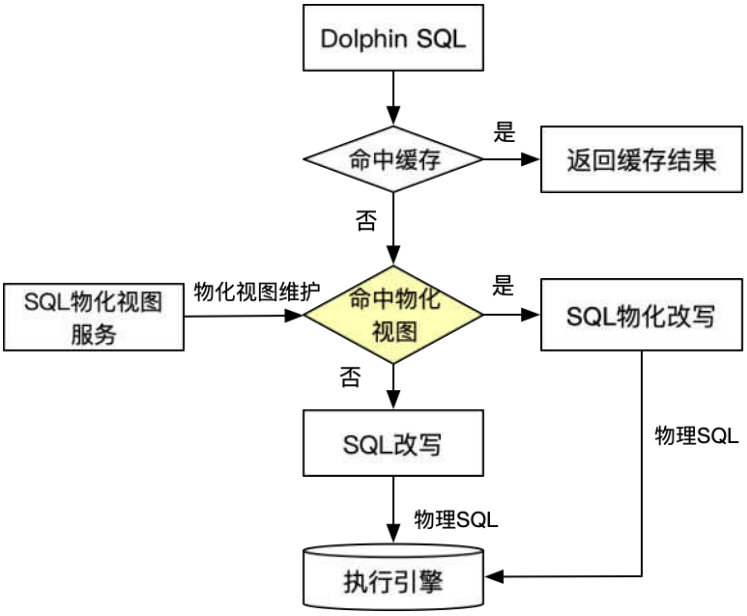
在算法广泛应用的今天，如何利用算法让系统更智能化从而提升性能和降低成本，已经成为业界探索的热点，近两年我们基于 Dolphin 在智能化方面已经有一些探索实践。

2.4.1 智能物化

在营销场景下每天有数百万分析查询请求，如何准确物化数据，用空间换时间提升查询性能成为重要的研究方向，我们首先从统计学角度入手，从 SQL 的 query block 粒度进行物化，实现对大部分复杂耗时的请求进行物化优化。

但统计学方法会有不足，例如通常是基于历史数据统计分析，而用户行为有周期性，尤其在电商场景，每年都会有一些如 38、618、双十一和年货节大促节点，这些行为特点使用统计方法很难捕捉，更适合使用机器学习方法，我们会结合用户查询行为序

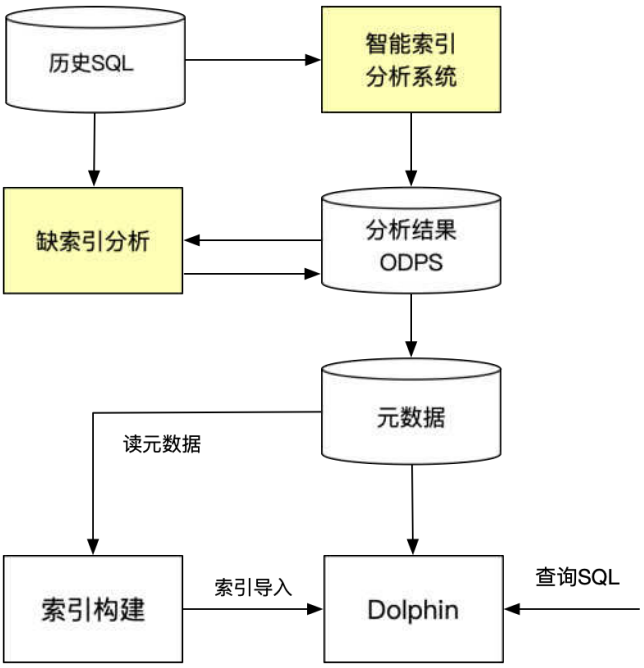
列数据进行建模，使用模型预测未来会到来的热点查询，从而提前物化来提升效果。



SQL 查询流程

2.4.2 智能索引

在达摩盘场景，每天查询 SQL 覆盖数千张表，这些表都是业务方配置接入，然而业务方不用关注如何优化表来提升性能，为提升用户体验我们向业务方屏蔽表的索引优化。这么多表的索引构建很复杂，分区列选错了没有优化效果，索引建多了会造成存储冗余，索引建少了查询性能会有影响，因此我们建立了一套系统化的方案，使用表的统计数据和历史查询信息，采用启发式算法来自动选择，达到智能选择分区列和索引的目的。

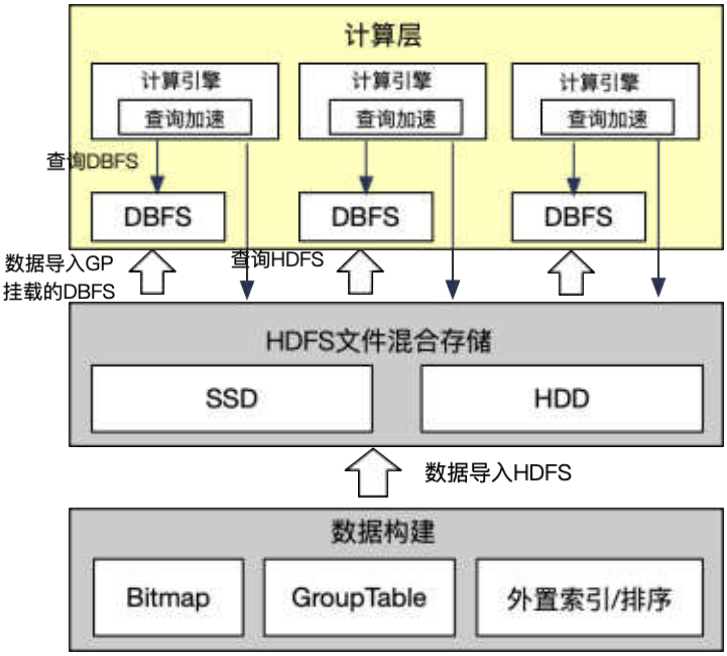


智能索引架构图

2.5 一写多读能力

Dolphin 底层使用 MPP 计算引擎，通过 DBFS（DBFS 是数据库场景的云原生共享文件存储服务）高性能 SDD 存储数据，高可用环境下会使用多集群冗余备份，为了减少数据冗余存储开销，我们采用一写多读方案，将部分存储占用大且查询量相对较少的表存在 HDFS，存储占用少但查询量大的数据存在 DBFS，这个思路类似冷热存储，但我们的工作不仅是数据跨介质存储，更多是在读冷数据优化方面。

为了保证查询 HDFS 的性能，我们做了一系列查询加速优化，如列裁剪、缓存和下推等，基于该方案，我们实现在查询性能不变的情况下，因为使用 HDD 存储大表数据，实现整体存储成本节省 70%+。



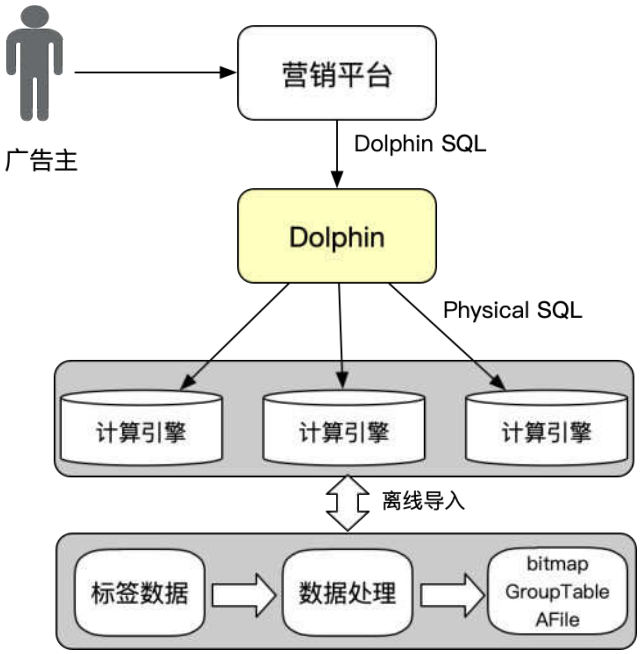
一写多读架构

3. Dolphin 领域能力

在业务需求多样的营销场景，Dolphin 作为超融合多模引擎，基于 SQL 和 index build 引擎核心能力，支持的能力范围已经包括 OLAP、AI Service、Streaming 和 Batch 四个方向，基本实现绝大部分需求都可以使用 Dolphin 一体化完成。

3.1 Dolphin OLAP

Dolphin 底层引擎最初就是服务于达摩盘 OLAP 分析查询业务，其在 OLAP 方向的沉淀最久，基于 Bitmap 和 GroupTable 索引的查询加速方案可以实现百毫秒、百 QPS 和万亿级数据精确圈人查询，不仅服务于阿里妈妈，还跨部门支持 10+BU 的圈人洞察业务；此外基于 MergeTree 表引擎的方案可以支持高性能报表查询和归因计算。



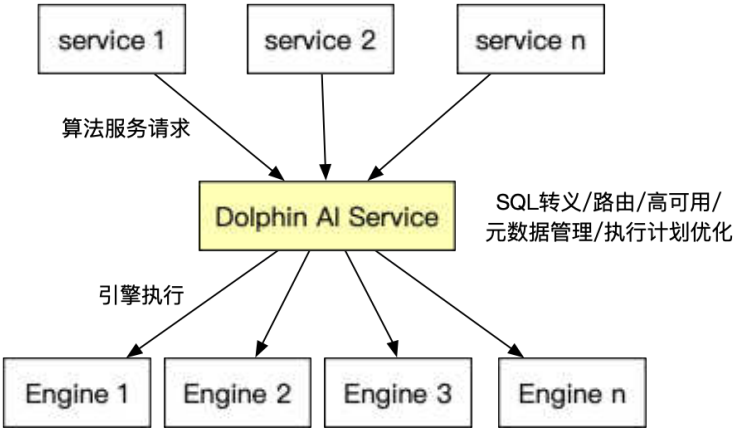
OLAP 计算架构图

3.2 Dolphin AI Service

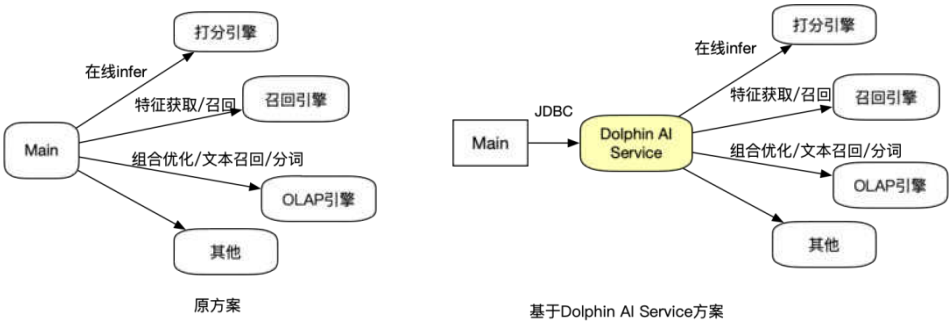
Dolphin 最初主要解决圈人问题，例如标签圈人、关键词圈人和 LBS 圈人等。随着商家对拉新、转化等营销诉求不断提升，需要在人群基础上进行更智能的分析挖掘，从而衍生到算法场景。

以往算法业务涉及链路很长，包括预处理、召回、排序和打分，往往需要跟多个独立系统进行交互，新的业务从数据接入、引入客户端到 debug 测试，环节繁杂，调试成本高，因为涉及多个系统交互，后续迭代成本也相当高。

然而单位时间内完成的迭代次数就是算法业务生产力，因此我们提出 Dolphin AI Service，基于不同独立系统之上构建一个 SQL 服务层，将业务逻辑和各系统进行解耦，提供统一的 Dolphin SQL 语法，极大降低算法同学学习和开发成本，实现使用 SQL 就可以完成业务开发及在线调用。



中间层设计思路引入中间层之后，算法开发业务只需要对接 Dolphin AI Service，使用 SQL 就可以完成预处理、向量召回和模型打分等计算需求。



下图所示是算法用户使用 SQL 实现关键词推荐的计算流程，包括词的预处理、模型打分，向量 TopK 召回和组合优化计算，所有计算都使用 SQL 表达，实现学习、调试及开发效率显著提升。

```
-- 词归一化
SELECT normalize('洗衣机排水');

-- 深度模型打分
SELECT score
FROM rtp_dolphin.dolphin_model_test
WHERE item_list in (1,2,3)
  AND qinfo = '${qinfo}'
  AND context = '${context}'

-- topk向量召回
SELECT id,
       euclidean_distance(emb, Array(0.123,0.321,0.212)) AS distance
FROM id_embedding_table
ORDER BY distance ASC
LIMIT 100;

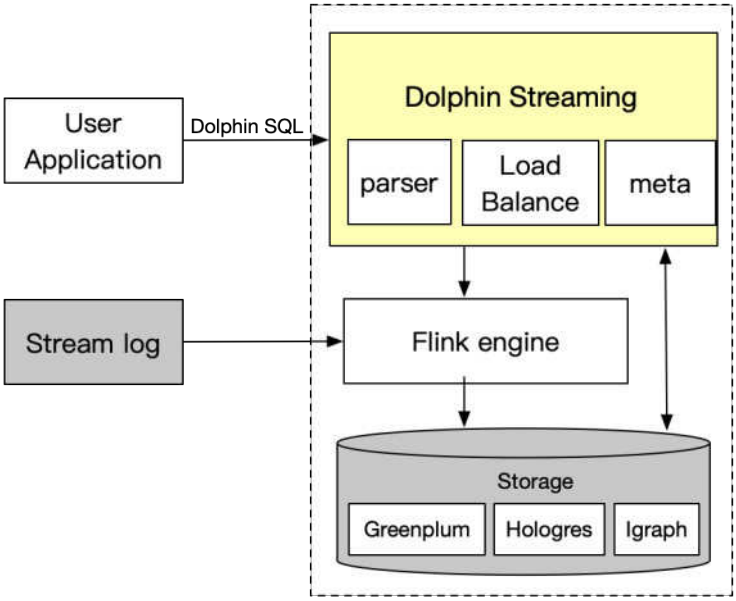
-- 组合优化
SOLVESELECT quantity IN (SELECT * FROM item) as u
MAXMIZE (SELECT SUM(quantity * profit) from u)
SUBJECTTO (SELECT SUM(quantity * weight) <= 15 FROM u),
  (SELECT 0 <= quantity <= 1 FROM u)
USING solverlp;
```

SQL 执行 AI 在线计算

3.3 Dolphin Streaming

在阿里妈妈商家端算法场景，算法业务迭代使用的数据是离线 T+1 甚至 T+7 更新频率，为更全面、实时的挖掘潜在需求，利用实时行为及反馈帮助广告主更好诊断选择，我们在已有引擎基础之上构建出 Dolphin Streaming，提出像开发数据库一样开发实时作业（DB for Streaming），基于该理念的设计特点包括：

- **底层透明：**用户无需感知底层 Flink 引擎和存储，极大降低学习和开发成本。
- **极简 SQL：**设计的 SQL 开发语法屏蔽不易理解的实时开发术语，例如 TUMBLE，HOP。
- **流程一体化：**打通从实时数据开发、测试到上线读取全流程，实现用 SQL 开发的特征也用 SQL 读取。



Dolphin Streaming架构

Dolphin Streaming 提供数据库 SQL 操作体验，结合 Flink 成熟大规模计算能力，屏蔽存储，让实时数据开发和查询一体化，是**用户体验**和**系统性能**都能达到较高水准的架构方案。

基于 Dolphin Streaming 开发实时作业分为三步：

- **定义输入源**，一般为实时数据源，只需要第一次定义，后续复用。
- **定义输出表**，可以写入实时数据源，也可以直接落盘存储，用户无需感知底层存储引擎。
- **定义计算逻辑**，这里主要是对输入源进行处理转换，数据写入输出源。

集群: dolphin_kgb_cmop_str...

```
-- 定义输入源
create table task_input_wl_ri_test
(
  user_id String,
  task_id_pv String,
  task_id_ipv String,
  behavior_time String
) with(
  bizType='tt',
  topic='task_wl_ri_test',
  pk='user_id',
  timeColumn='behavior_time'
)

-- 定义输出表
create table task_output_wl_test (
  time_id String
  user_id_list String
) with (
  bizType='feature',
  pk='time_id'
);


insert into task_output_wl_test
select window_start(behavior_time) as time_id,
       concat_id(true, user_id) as user_id_list
from task_input_wl_ri_test
group by window_time(behavior_time, '1 MINUTE');
```

实时作业开发流程

运行作业后就可以直接使用 SQL 查询特征结果：

集群: dolphin_kgb_cmop_str...

```
       concat_id(true, user_id) as user_id_list
from qianniu_sop_task_wl_ri_test6
group by window_time(behavior_time, '1 MINUTE');

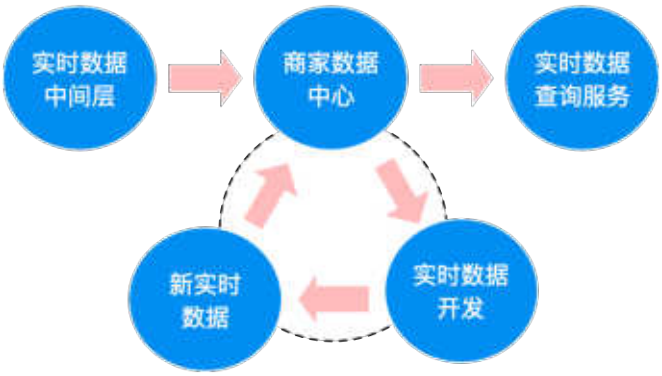

select time_id,user_id_list from
qianniu_sop_task_wl_test6
where time_id in ('202201121510')
```

运行日志 | 结果[1] ×

	A	B
1	time_id	user_id_list
2	202201121510	143786037,2212273210258,745148460,3368439272,267787

SQL 查询实时特征

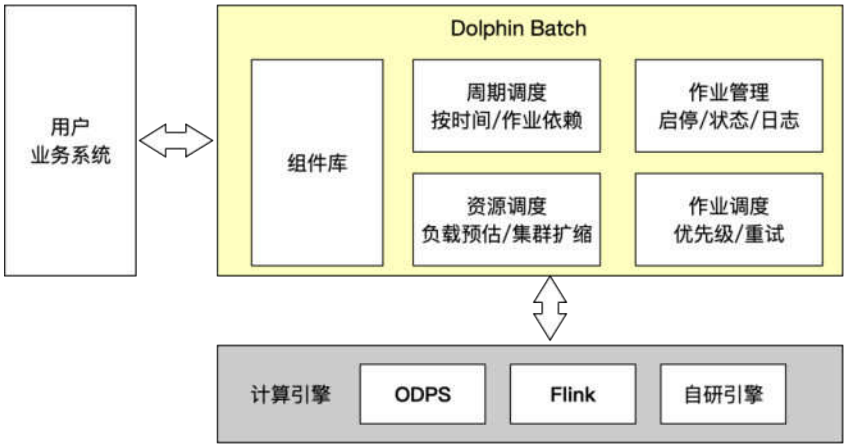
此外为解决数据分散使用困难问题，我们还基于 Dolphin Streaming 构建实时数据体系，让数据在体系内循环沉淀，用户可以在商家数据中心查询需要的上游实时数据，然后使用 SQL 实现实时特征开发及生产，实现用户只需要懂 SQL 就可轻松完成实时作业开发。



数据循环流动

3.4 Dolphin Batch

在各业务场景离线计算是非常普遍需求，集团 ODPS 可提供通用数据离线计算能力，但难以高效满足领域计算场景，很多相同业务逻辑的需求都是各自独立开发，大量重复建设，例如离线批量向量召回，离线批量打分等，因此我们构建出 Dolphin Batch 解决领域计算场景重复建设成本问题和性能问题。



Dolphin Batch 架构图

基于 Dolphin Batch 我们在交互式 web 界面 5 分钟就可以完成一个向量召回作业的配置，该能力通过产品化不仅支持妈妈内部团队，还支持其他 BU。我们不仅实现领域能力产品化复用，还通过使用自研引擎的向量召回方案，实现性能和作业调度稳定性大幅提升。



离线批量向量召回

4. 总结及规划

这些年来大数据技术在产品使用和技术演进层面都在向超融合数据库发展，技术方面如事物、向量计算、索引、冷热存储、物化、实时读写和模型应用等。大数据的使用从用户角度看就是一体化数据库系统，算法工程师和数据科学家可以使用 SQL 完成任意规模数据处理、实时作业开发和算法开发等业务需求，其背后底层是极其复杂的工程和算法实现。

Dolphin 未来将持续基于 Serverless、一体化和智能化方向不断升级，面向用户需求扩展业务边界，让用户体验简单高效，屏蔽复杂流程和性能问题，通过技术推动经营增长。

阿里妈妈工程平台智能分析引擎团队 – 系列文章：

阿里妈妈 Dolphin 分布式向量召回技术详解
阿里妈妈 Dolphin 智能计算引擎基于 Flink+Hologres 实践
Dolphin Streaming 实时计算，助力商家端算法第二增长曲线
面向数智营销的 AI FAAS 解决方案
FAE：阿里妈妈归因分析与用户增长分析引擎
开源 greenplum 向量计算库：https://github.com/AlibabaIncubator/gpdb-faiss-vector

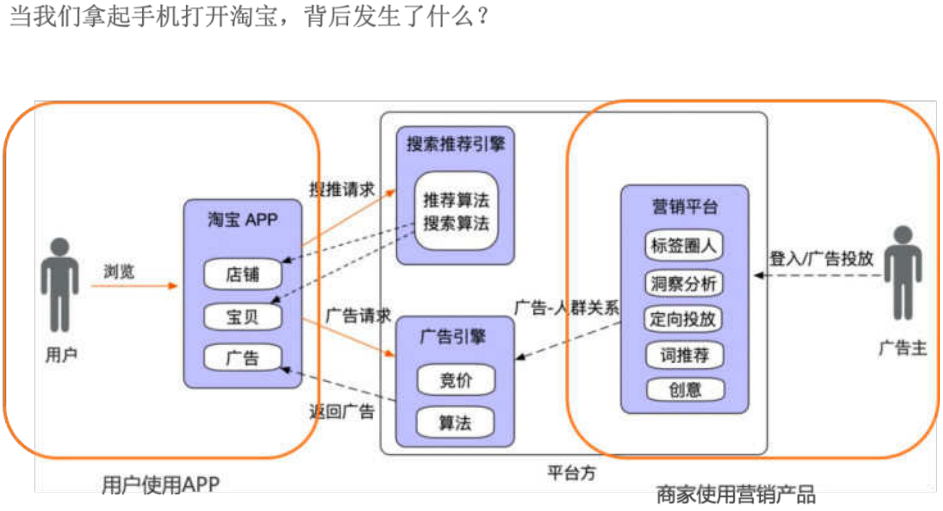
阿里妈妈 Dolphin 智能计算引擎基于 Flink+Hologres 实践

作者：陌奈

阿里妈妈数据引擎团队负责广告营销计算引擎 Dophin 的开发，目前支撑百万级广告主的营销产品，支持万亿级数据毫秒级交互式人群圈选、洞察分析及百亿级数据秒级广告效果分析，同时支持 OLAP、实时、离线及 AI 超融合计算能力，为商家端产品万相台、直通车、超级推荐和达摩盘等营销产品提供极速的数据探索能力。

1. 阿里妈妈 Dolphin 智能计算引擎介绍

用户打开淘宝 App 时，后台会有两种类型的请求，第一种类型是满足用户诉求的自然推荐，第二种请求是满足用户和商家综合诉求的广告推荐。例如打开淘宝看到某品牌，是因为该品牌使用阿里妈妈营销产品圈选人群进行广告投放，被圈选的人会看到该广告。

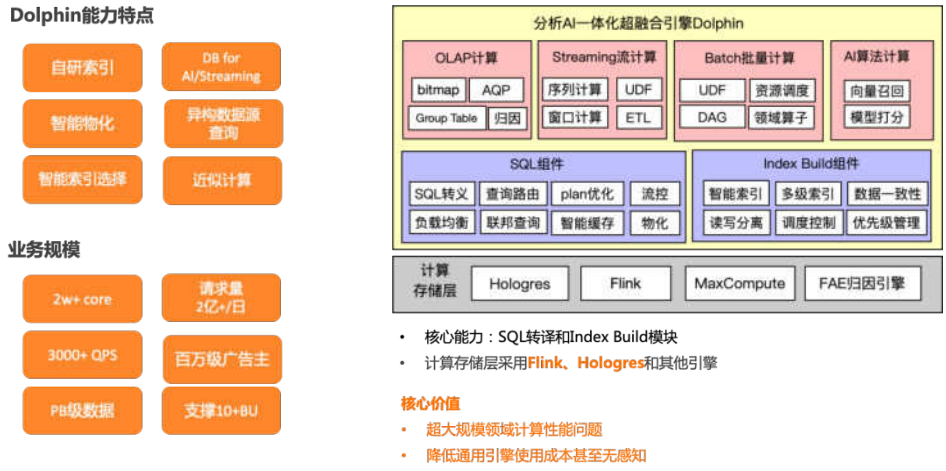


商家端营销产品的主要目标是服务于广告主，帮助广告主进行人群投放，从而提升经营效果。此类营销产品覆盖的场景非常广泛，包括人群圈选、洞察分析、Lookalike、人群推荐等场景。这些场景会有 OLAP 分析、AI 算法和实时特征计算的基础能力需求，基于这样一个数据 + 算法综合能力需求背景下，阿里妈妈自研了 Dolphin 计算引擎。



Dolphin 引擎是一个分析 AI 一体化的超融合引擎，拥有 OLAP 分析计算、Streaming 流计算、Batch 批量计算和 AI 算法计算四个领域能力，这些能力基于 SQL 组件和 Index Build 组件构建：

- SQL 组件的主要能力是 SQL 转义、路由、负载均衡、联邦查询。
- Index Build 组件主要负责智能索引、多级索引（Bitmap 索引、时间序列索引等）、调度控制等。



Dophin 引擎提供了自研索引、智能物化、智能索引选择、异构数据源查询和近似计算等几个优势功能。其中：

- 智能物化：智能物化指能够自动将 SQL 转化为物化视图，无需人工手动操作。使用深度模型对业务历史查询 SQL 进行时序分析，比如哪些广告主在什么时间周期的数据使用频率更高，可以选择将高频的使用数据进行物化，以提高数据查询效率。
- 智能索引：大多数业界的做法是为查询建立全索引，而智能索引要做的事情就

是分析 SQL 查询语句，判断条件命中率，从而推荐不同的索引，推荐目标为让索引对数据查询的过滤量最大，同时避免建立无效索引占用。

- 近似计算：根据统计的结果用近似的算法来计算，主要针对大规模数据，计算其近似结果。

目前，Dolphin 引擎支撑的业务规模为 2w+core，每日请求量 2 亿 +，QPS 为 3000+，支撑百万级广告主、存储了 PB 级数据，涵盖阿里 10+ 核心 BU 的核心场景如人群圈选、洞察分析等场景，业务方可以通过 Dolphin 引擎就能非常高效且低成本的进行数据探索。

Dolphin 引擎主要解决两个问题，一个是超大规模场景下使用通用计算方法存在的性能瓶颈问题；第二个就是降低业务方使用引擎成本，甚至做到对底层引擎无感知。很多通用引擎并不能直接解决业务的性能问题，因此需要对数据做索引以实现查询优化。因此我们在已有的引擎基础上建设了 Dolphin 引擎。那从 Dolphin 引擎架构图我们也可以看到，引擎的底层计算存储层主要由 Flink+Hologres 来实现，这就相当于，广告主每一次在 Dolphin 引擎上的计算，最终都会被转换成由 Flink+Hologres 来完成，那么我们为什么选择 Flink+Hologres 这套架构呢?

2. 为什么选用 Flink+Hologres ?

选择 Flink+Hologres 这套架构主要来源于对底层引擎的强需求：

1. 高性能

广告场景对延迟有着很敏感的要求，如果底层引擎的性能不足，即使上层应用再做优化，也会导致计算性能大打折扣，从而导致广告投放等不精准。

2. 可扩展

底层引擎需要足够的可扩展，这样上层应用才能更加灵活的去承接新的业务、新的场景，应对业务高低峰期等情况。

而 Flink 和 Hologres 能够满足以上两个业务需求：

- 实时引擎 Flink 支持低延迟，支持自动调优，且技术稳定成熟。用户在 Dolphin Streaming 平台上提交 SQL 时，SQL 会被转译为 Flink SQL，提交给 Flink 引擎，这里会利用 Flink autoscale 自动配置调优功能来自动调整，降低可了用户对 Flink 的学习成本和调优维护成本。

- Dolphin 引擎与 Hologres 更有着更长的合作历史：在 2019 年 Dolphin 引擎与 Hologres 共建了 Bitmap 计算能力，实现业界公开信息里最大规模、最高性能、最低使用成本的圈人能力；2020 年共建了千万级人群中心，实现广告人群统一管理；2021 年，Dolphin 引擎支撑了算法业务场景，使用 Hologres 向量计算能力支撑算法业务，主要支撑推荐算法里粗排召回计算环节；2022 年，支持了算法实时特征开发能力，这里就运用到 Hologres 实时写入和点查能力，实现了更高效的实时开发。最终，我们将 Hologres 的所有能力整合在一起，形成了超融合一体化引擎能力。



引擎实现细节 1：如何解决超大规模 OLAP 计算能力

挑战：

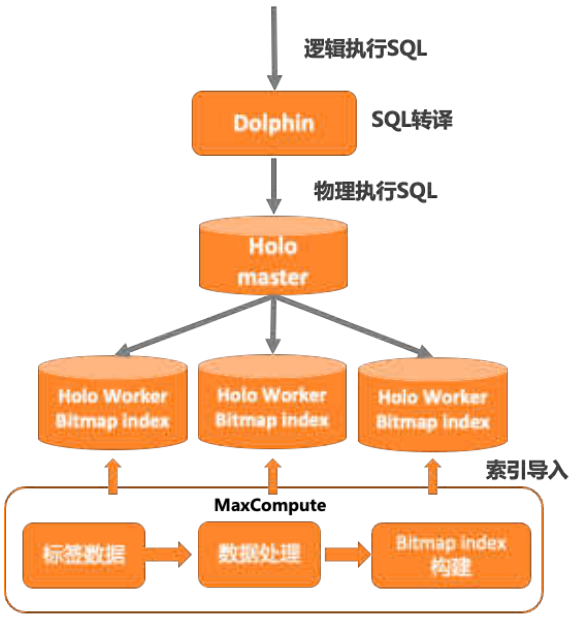
Dolphin OLAP 计算的核心在于解决超大规模问题。阿里妈妈广告场景上存放着大量的数据，为了让广告主有更好的用户体验，需要支持更复杂的计算逻辑和更快的计算速度。其中典型的场景有单 SQL 几十张表 Join、单表最高万亿行规模、单表基数最高百万级和万级标签日更新。

解决方案：

例如广告场景上有很常见的数据表：用户基础表（性别年龄）和用户店铺表（用户、店铺类型），当需要查询 20、30 岁且逛过某品牌的用户数量时，如果数据量很少，通用引擎可以很快得出结果；但如果 SQL 涉及 Join 的表有几十张，而且还可能存在万亿级表，此类情况下通用计算引擎无法完成计算。**因此，我们基于 Hologres 共建了一套**

Bitmap 计算方案。具体方案流程为：

1. 方案查询流程：用户输入逻辑执行 SQL ，Dolphin 引擎将用户逻辑执行 SQL 转译为物理执行 SQL ，然后传递给 Hologres 执行。
2. 方案索引构建流程：MaxCompute 将标签数据进行预处理，然后将它构建为 Bitmap 索引，再写入到 Hologres，即可实现的 Bitmap 查询。



SQL转译执行流程

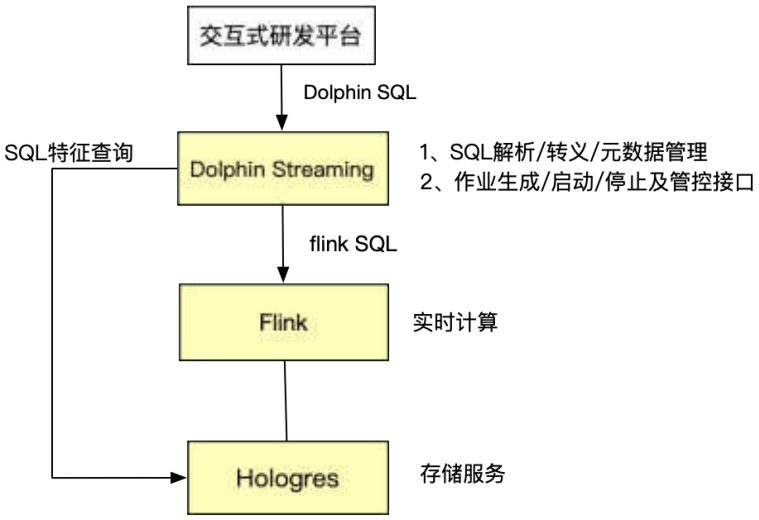
通过这样的 Bitmap 方案，能够让查询拥有更好的性能和更低的存储，在超大规模 OLAP 计算场景中，支撑了 200+QPS ，平均百毫秒查询性能，以及万亿行数据秒级精确计算，高效支撑用户交互式分析低延迟的用户体验。

引擎实现细节 2：如何实现低成本的实时开发能力

广告场景通常都是实时计算，这里我们通过 Flink 来支持是非常方便的，而我们的面向的用户通常都是不同岗位，比如算法、运营等，假如全部都用 Flink 去开发任务，那么对于上层应用的同学来说就会额外增加非常多的学习成本和压力，比如既要学会 Flink SQL，还要学会 Hologres SQL，流程和操作都非常繁琐。为了降低用户的学习成本，提升开发效率，Dolphin 使用 OpenAPI 做了丰富的实践，开发了一套 Dolphin Streaming 实时开发平台，通过 Open API 直接以服务接口调用的形式调

用 Flink 提交作业、暂停作业、管理作业等。

Dolphin Streaming 将 Hologres 和 Flink 做了封装，对用户暴露更简单的开发接口 Dolphin SQL 。用户在阿里妈妈交互式研发平台上提交 Dolphin SQL，SQL 变会自动通过 Dolphin Streaming 进行处理，做 SQL 解析及转译，将 SQL 通过 OpenAPI 发送给 Flink，拉起作业做执行。执行完后，数据会实时写入到 Hologres，然后通过 Dolphin SQL 将写入 Hologres 的特征直接查询出来，无需再考虑存储、配置认证信息、token 信息等，只需像使用数据库一样开发实时作业，整个流程非常顺滑简单，大大提高开发效率。



demo1：计算用户最近 50 条行为序列

用户最近 50 条行为序列是算法序列模型里常用的特征，一般需要开发行为序列特征，如果用 Dolphin Streaming 开发，只需简单三步：

1. 第一步，定义数据源表。Biztype 可直接填写为 tt ，tt 是阿里的实时数据源。这里如果要写 Flink SQL，则需要登入 tt 管理平台，查询 topic 并订阅 subID。
2. 第二步，定义输出表。Biztype=feature 代表写入到 Hologres，然后填好 PK 参数即可。
3. 第三，定义计算逻辑。SQL 执行完之后，数据源源不断地写入，通过 select user_id，product_id from ** where user _id=** 即可查询用户特征。

```
create table qianniu_icon_behavior_wl_ri
(
  user_id String,
  product_id String,
  label String,
  behavior_time String
) with(
  bizType='tt',
  topic='qianniu_icon_behavior_wl_ri',
  pk='user_id',
  timeColumn='behavior_time'
)

-- 定义输出表
create table qianniu_icon_behavior_wl (
  user_id String,
  product_id String,
  label String,
  behavior_time String
) with (
  bizType='feature'
  pk='user_id'
);

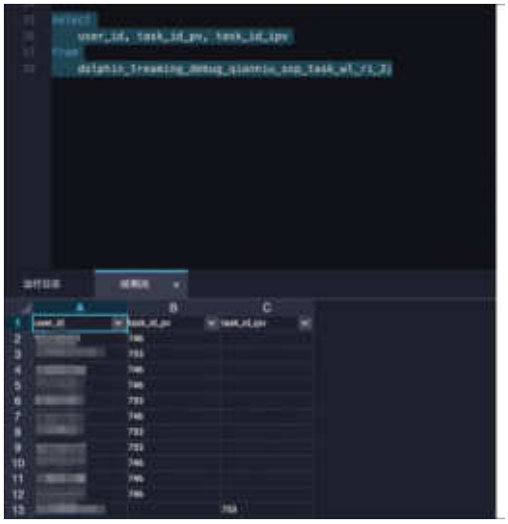
--定义计算逻辑
insert into table qianniu_icon_behavior_wl
select user_id,
       concat_id(product_id, behavior_time, 50) as product_id,
       concat_id(label, behavior_time, 50) as label,
       concat_id(behavior_time, behavior_time, 50) as behavior_time
from qianniu_icon_behavior_wl_ri
group by user_id;

-- 查询特征
select user_id, product_id, label, behavior_time
from qianniu_icon_behavior_wl
where user_id in ('3330415300');
```

demo2：实时 Debug 功能。

在实时开发时，经常需要查看上游数据源，以往的方式通常需要定义一个 print 输出源，然后定义输入源和执行逻辑，将数据写到标准输出，再通过查看日志才能获取到上游数据源。而我们实现了更简单的方式。

通过 create table 形式注册一张表以后，执行 select user_id from 某表，结果即直接展示表的明细。



通过 Dolphin Streaming，我们可以非常高效的将复杂计算逻辑进行自动封装与转换，用户无需自己写 SQL，也不需要去学习多种开发语言，就能非常高效的拿到想要的数据，大大降低了学习成本和使用门槛，同时也节省了开发效率。

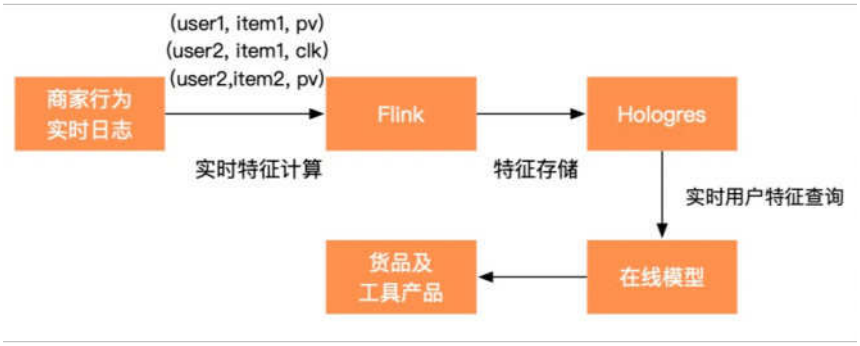
3. 业务场景实战

场景 1：实时营销推荐

实时营销推荐是广告中最常见的场景，在该场景下，最大的痛点在于：广告主在使用营销平台时，常常面对如何推广、通过何种渠道推广等问题。

基于这样的用户诉求，从算法角度为广告主解决该问题：通过广告主点击某些信息、某些广告点位时，判断广告主意向，结合意向和广告主本身商家店铺和商品信息，为其推荐能提升经营效果的商品以及效果更优的投放渠道，从而让广告主的投放体验更好。

基于以上需求，我们开发了一套用于捕获用户实时行为的作业：通过 Flink 计算商家实时行为日志，存储到 Hologres，然后在线模型直接读取特征，通过实时特征提升模型的推荐效果。



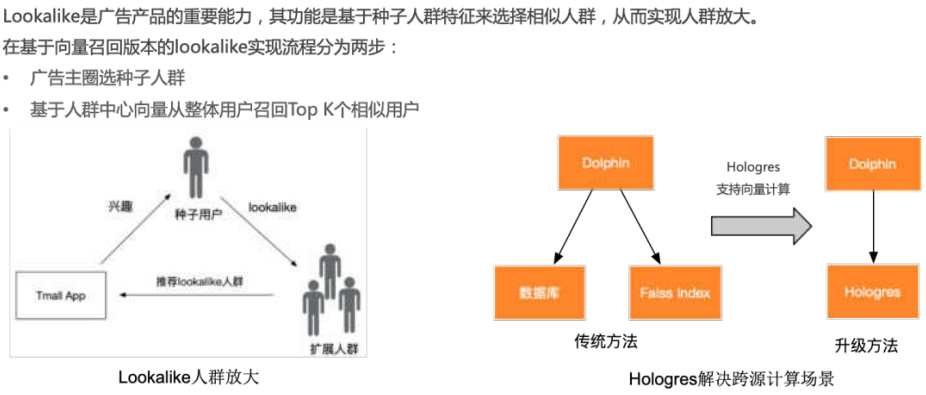
Dolphin Streaming 方案主要使用了 Flink 实时计算、Hologres 实时写入以及行表的点查能力，使整体开发效率提升三倍以上，推荐效果更佳。

场景 2：向量召回计算

在算法里万物皆可向量表示，尤其是在推荐算法和召回流程中，经常使用向量召回获取 Top K 相似对象，在阿里妈妈向量召回的场景中，我们使用 Hologres 的向量召回能力，以 Lookalike 场景为例具体说明：

Lookalike 是广告产品的重要能力，核心是基于种子人群特征选择相似人群，常应用在拉新场景。以电商场景为例，其原理可以抽象理解为针对已经在店铺有过行为的用户，分析其特征，寻找与之特征相似的用户，将宝贝推广给此类用户，从而促进店铺新用户增长。基于向量 Lookalike 算法实现过程如下：

- 第一步，广告主圈种子人群。
- 第二步，基于种子人群计算中心向量，再通过中心向量从整体用户里召回 Top K 相似的用户。



传统算法下，Dolphin 会使用没有向量召回能力的传统数据库，将数据导入到数据库中。先由 Dolphin 查询数据库，计算种子人群中心向量，然后通过 Faiss 将 Top K 向量查出。传统方案整体运维和管理成本较高，因此我们对其进行了升级，直接使用 Dolphin 调用 Hologres，因为 Hologres 能够同时支持数据库功能和 Proxima 向量功能，简化了计算流程。

基于 Hologres 的向量召回能力，我们开发了实时向量召回和批量向量召回能力，用户直接输入 SQL 即可调用底层 Hologres，Dolphin 封装了容灾和负载均衡等重要能力，简单地填入参数即可完成批量向量召回的计算。其中

- 实时向量召回：支撑 1000+QPS，平均延迟 50ms+，支撑了直通车、万向台、达摩盘等多个商家段营销业务场景
- 批量向量召回：目前已对外产品化，提升开发效率 3 倍 +，有效支撑达摩盘、直通车等多个算法业务

4. 总结

基于 Flink+ Hologres 的强大能力，我们得以建设更贴近业务领域的超融合一体化 Dolphin 引擎，主要包括基于 Bitmap 的高性能 OLAP 计算、更简单灵活的实时开发能力以及基于 Hologres 强大的 AI 向量召回能力。

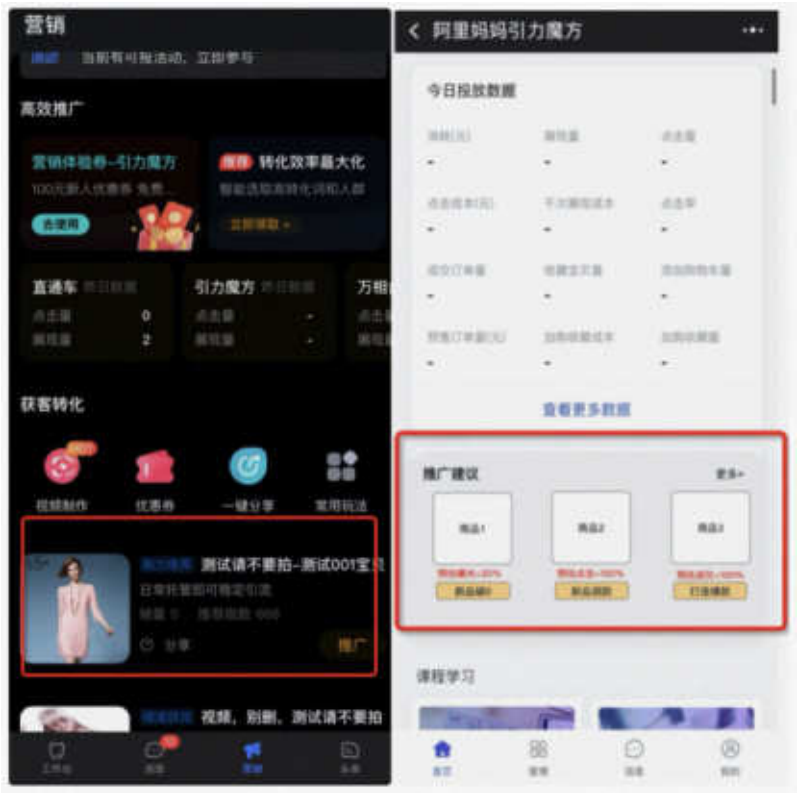
未来，我们会在智能化、一体化方面继续探索，不断提升用户体验。

Dolphin Streaming 实时计算，助力商家端算法第二增长曲线

作者：陌奈、茂道

1. 背景

随着业务朝向精细化经营增长，阿里妈妈商家端营销产品更加聚焦客户投放体验，旨在帮助商家提升经营效果，在变化的市场中找到确定增长。近年来，商家端算法业务使用的数据是离线 T+1 甚至 T+7 更新，为进一步捕捉用户意图，更全面实时的挖掘潜在需求，利用实时行为及投放效果帮助广告主在成效预估、货品工具推荐等业务有更好效果，阿里妈妈数据引擎团队从 21 年开始在数据实时化开发方面进行探索尝试，从实时角度助力商家端算法第二增长曲线。



货品工具推荐场景

2. 业务问题

与用户端（C 端）相比，商家端（B 端）算法业务更具多样性，但对实时数据的使用还处于启蒙阶段。目前面向 C 端的实时开发服务已经很成熟，但开放的能力比较基础，且这些能力主要面向工程同学，但在实际 B 端场景中，因算法工程支持资源有限，而算法同学自己直接开发实时作业成本较高，不仅需要学习了解上游实时数据源订阅信息，还需要了解不同存储引擎选型等工程技术支持，例如 Igraph（阿里集团内部 KV 存储引擎）、Lindorm（阿里云多模存储引擎）和 Hologres（阿里云 HTAP 存储引擎）等，所以需要有一个更算法友好的开发平台，**实现让非工程同学也能轻松开发实时作业**。

那么，对于算法同学什么开发方式最简单？因为算法同学对 SQL 非常熟悉，每天大量工作都在 Dataworks（Dataworks 是阿里集团大数据开发平台）完成，所以能让实时作业 SQL 化开发是平台确定方向。目前 Flink 已经可以提供 SQL 化开发，但仅提供基础实时计算开发能力，存储方面需要自己选择，对于非工程技术人员仍有较高的学习成本，故期望如下能力：

- **屏蔽底层细节的 SQL 化开发**，不仅开发 SQL 化，还可以帮助用户**屏蔽底层存储和上层数据源配置**信息，降低学习及开发成本；
- **统一的数据中心**，从实时开发的数据获取、开发调试及上线 **End2End 一体化**，提升开发效能。

3. 业界解决方案

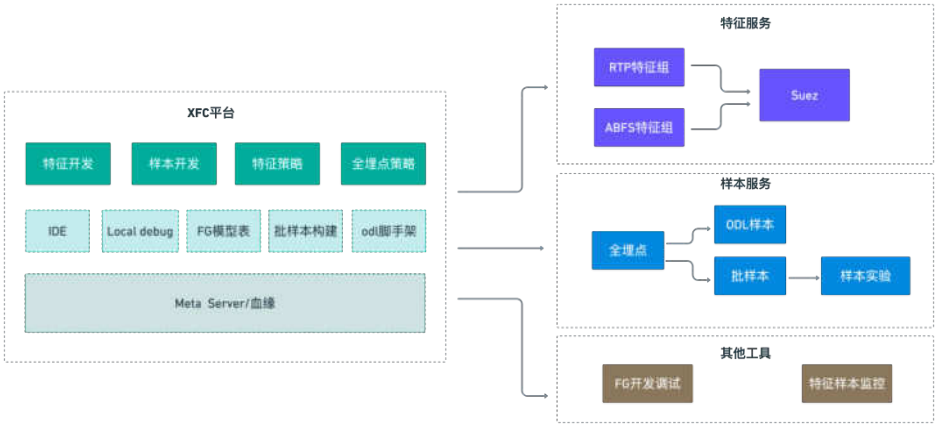
如何更高效开发实时作业，业界有很多尝试和探索。

3.1 集团内部解决方案

在集团内部，经常使用的实时化产品有 AMC 特征中心和蚂蚁特征服务平台等，它们体系化建设完善且功能全面，但大多是工程同学使用，有一定开发使用成本。

3.1.1 AMC 特征中心

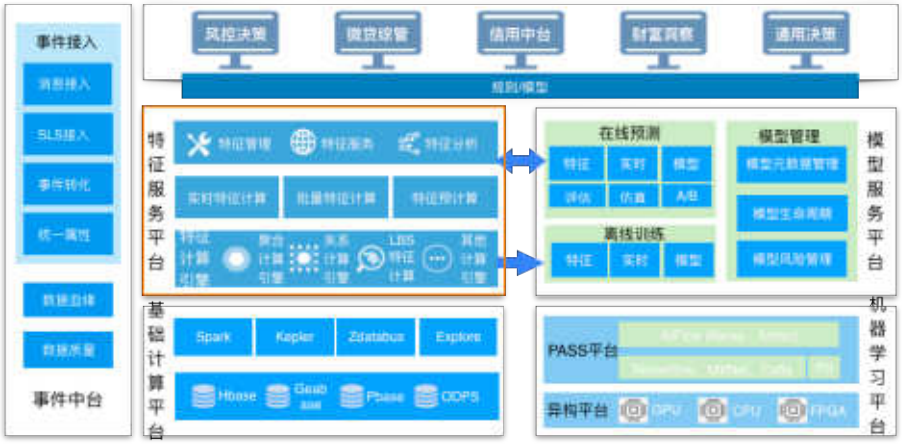
AMC 是特征样本平台，解决主搜场景算法同学特征迭代遇到的问题，提供复杂特征开发和统一特征管理问题。



在复杂特征开发方面提供 TableApi，支持算法自助开发复杂特征，该方案灵活性比较高，但是从算法开发体验和 debug 角度看，成本仍然较高。

3.1.2 蚂蚁特征服务平台

蚂蚁提供全平台统一的特征服务平台，提供特征管理、服务、分析和计算等能力。



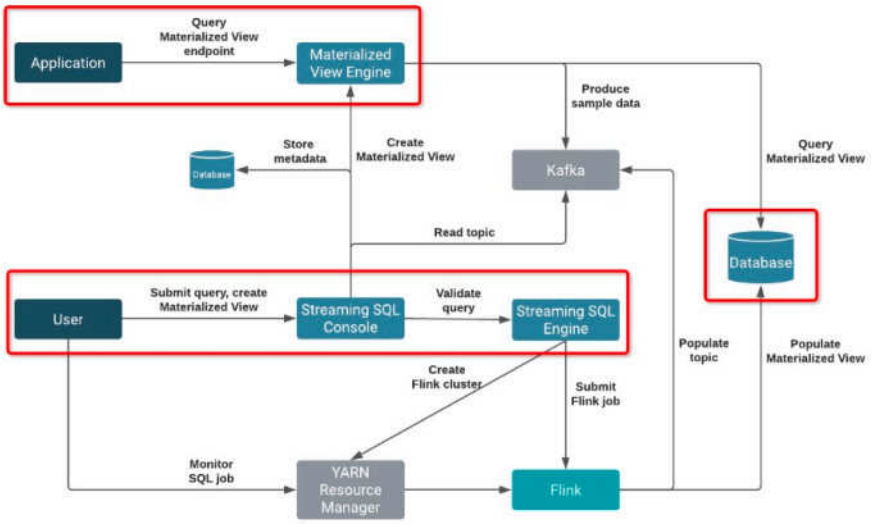
为了简化计算流程，平台提出**特征 SQL 语法**，该语法使用类似 SQL，但区别较大，对新开发同学学习成本不低，因为新语法有很多非通用概念。

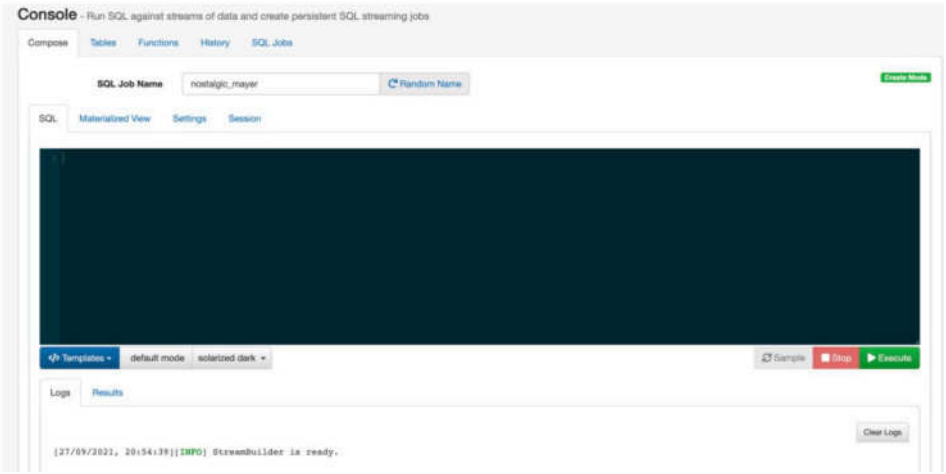
3.2 外界解决方案

调研发现外界有很多类似的设计，主要有老牌云厂商 Cloudera 的 SSB 和新兴公司 RisingWave。

3.2.1 Cloudera Stream Builder

Cloudera 在实时开发方面有成熟商业化实践，Stream Builder 整体上是基于 Flink 和 Database 封装形成实时开发平台，使用 SQL 进行开发，数据经过 Flink 处理写入 database，从 Database 读取数据，架构图如下。





上图为用户开发页面，所有的数据源和存储库都抽象出来，配置一次即可，无需每个 job 都配置一次，此外该平台还提供 sample 能力支持数据 preview，大幅提升数据开发效率。

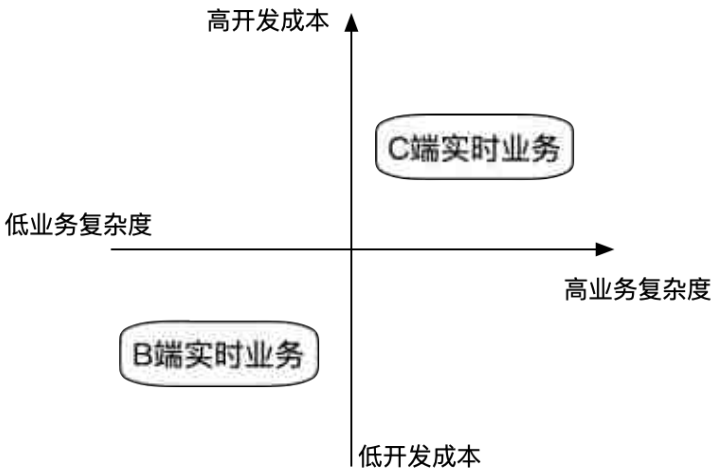
3.2.2 RisingWave

RisingWare 在 2021 年底开源，主打实时数据库，不仅包括实时计算能力，还有自己的存储，提供 PG SQL 语法，用户可以像使用传统数据库一样开发实时作业，整个流程就像操作传统数据库，**所写即所得**。

它和我们系统设计的目标非常一致，提供通用的 SQL 语法，且使用方面不用过多考虑数据来源和存储选择，学习成本、开发、调试及上线都一体化，是用户体验和开发成本更优的方案，但 RisingWave 还在 dev 阶段。

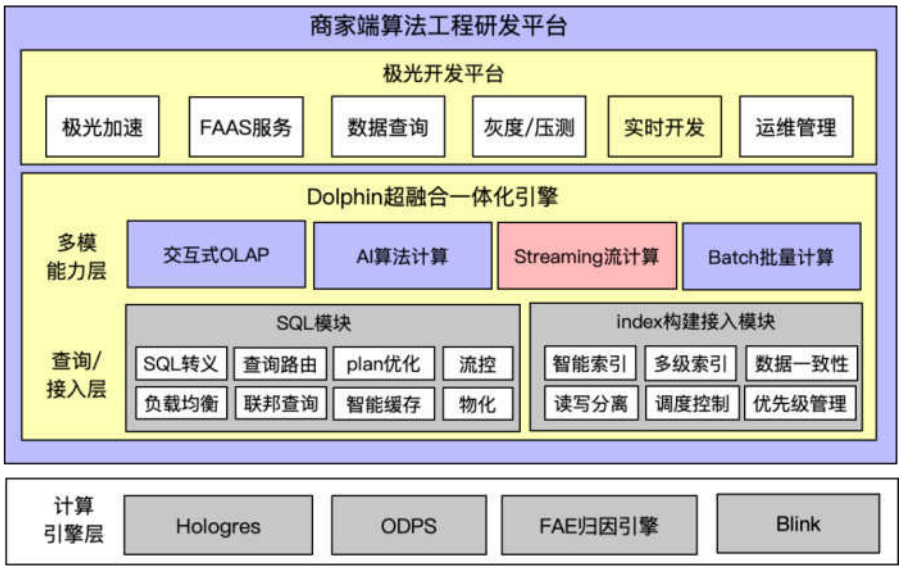
4. 技术方案

当前集团提供的实时开发方案多面向工程技术同学，具有相对灵活的控制能力，可解决超大规模复杂场景；但在广告商家端场景，面对百万规模的用户实时行为，业务更看重**开发迭代效率**，期望把复杂工程细节屏蔽，节省人力成本提升迭代效率。



Dolphin 引擎是阿里妈妈数据引擎团队自 2018 年底研发的超融合一体化计算引擎，在面向商家端营销产品场景下经过多年发展，已经从最初的 OLAP 计算延伸到 AI 计算、实时计算和批量计算，让业务迭代效能达到较高水位。

引擎基于在 SQL 领域的基础能力，在 2021 年中研发出 Streaming 框架能力，填补实时计算能力空白，解决实时计算高开发及维护成本问题，下图是 Dolphin Streaming 在整体 B 端工程解决方案的定位，主要是基于计算存储引擎以及 SQL 引擎能力构建。



4.1 设计思路

Dolphin Streaming 设计目标是**像开发数据库一样开发实时作业（DB for Streaming）**，让算法等非工程用户也可以轻松开发实时作业，具体包括：

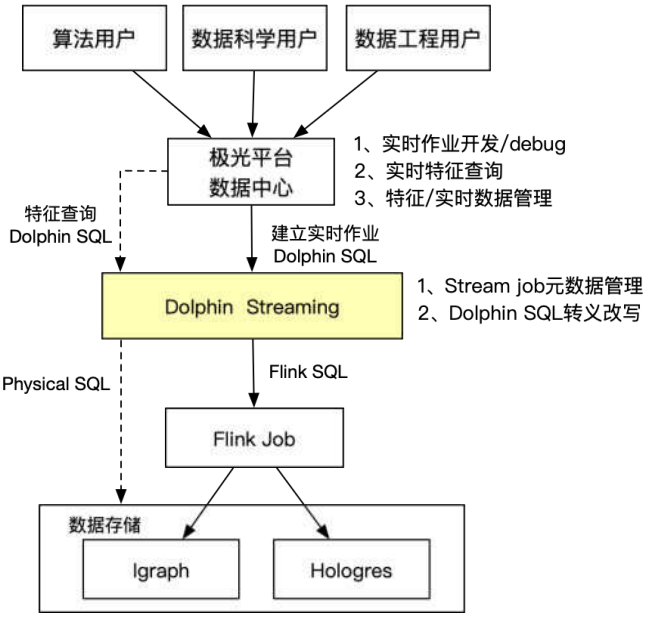
- **极简 SQL 语法：**屏蔽有理解成本的实时开发术语，如 TUMBLE、HOP 等；
- **底层技术无感知：**为算法用户及其他非工程用户提供一套开发平台，无需过多感知上下游数据源和存储；
- **流程一体化：**打通从实时数据开发、迭代到上线读取全流程。

该设计跟 Cloudera、Risingwave 有很多相似，但也有区别：

- 不仅关注实时开发平台，还关注用户对**实时数据获取**、开发及高效复用；
- 设计**简化 SQL 语法**，屏蔽 TUMBLE、HOP 等概念；
- 不仅关注数据开发产出，还关注用户开发后直接**上线数据流程**。

4.2 架构图

我们开发了从数据、计算引擎到上线一体化方案，端到端高效实现数据开发、存储、debug 查询及上线整个流程。

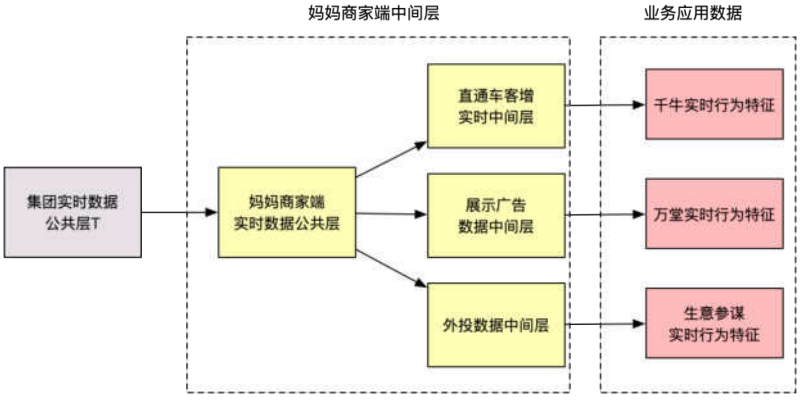


4.3 数据层

目前阿里妈妈广告商家端数据散落在各个团队，我们期望面向商家端场景搭建**数据中心**，让离线和实时数据被更好的管理、复用。具体设计如下：

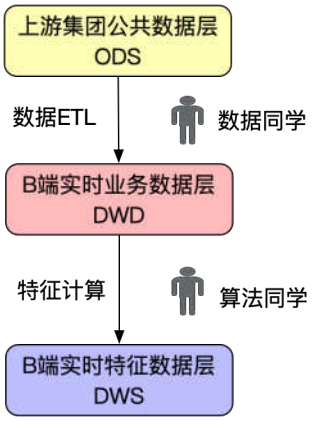
- **将实时行为标准化**，沉淀实时、离线特征基础设施，提升数据复用，减少重复存储和开发。
- **建立实时数据地图**，降低实时数据查找、管理及维护成本。

建立商家端数据中心，并跟极光开发平台结合，形成集团公共层、妈妈商家端中间层到业务应用层三层数据体系。

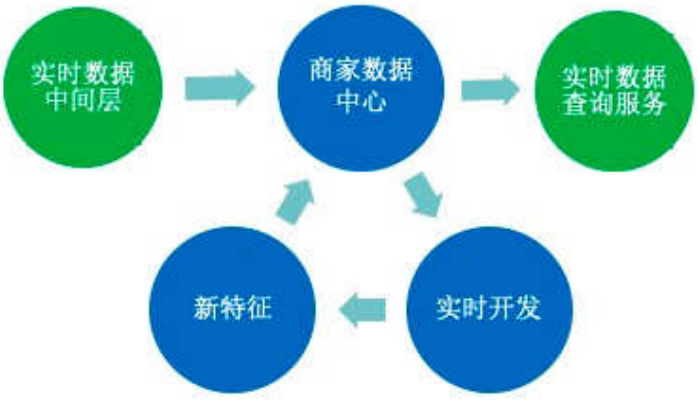


此外，我们还建立了面向商家端算法实时开发的协作模式：

- 数据同学负责将集团公众层非结构化数据 ETL 为结构化的实时数据中间层；
- 算法等非工程同学自主使用 Dolphin SQL 将上游中间层实时数据进一步开发聚合为所需要的特征数据；
- 特征数据会进入商家数据中心被复用。



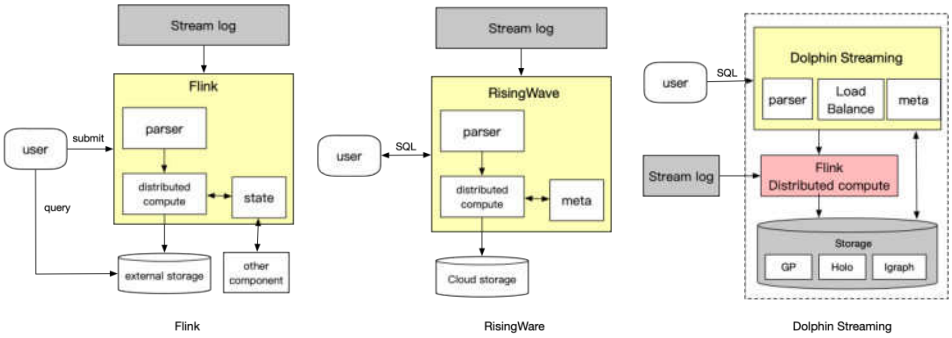
数据同学开发的中间层数据进入商家数据中心，数据一方面可被直接被查询，另一方面可以给算法同学开发特征使用，开发的特征会进入商家特征中心，从而形成数据闭环和复用，如下图所示。



4.3 Dolphin 引擎

Dolphin Streaming 在实现方面基于 Flink 实现，使用 OpenAPI 实现对 Flink job 的创建、资源配置和启停；在存储方面使用 GP、Hologres 和 igraph，该方案既利

用现有引擎支撑大规模场景的成熟能力，又让使用体验像数据库一样简单。下图是 Flink、RisingWave 和 Dolphin Streaming 的架构特点：



这里主要对比 Dolphin Streaming 跟 Flink 的区别：

- Flink 定位实时计算引擎，没有自己的存储，用户在使用必须定义外部存储
- Dolphin Streaming 屏蔽计算和存储引擎，计算引擎使用 Flink，存储使用不同引擎
  - 用户无需感知 Flink SQL 语法，只需要使用更简单，更接近 SQL 标准的 Dolphin SQL
  - 用户开发无需感知底层存储类型、表配置等信息

Dolphin Streaming 提供数据库 SQL 操作方法，使用 Flink 成熟大规模计算能力，屏蔽存储，**让实时数据开发和查询一体化，是结合用户使用体验和性能的不二之选**。

4.3.1 SQL 转译

Dolphin streaming 提供一套面向算法用户友好的 SQL 语法，屏蔽数据源信息、中间结果处理、输出信息，让算法用户开发实时特征就像在 Dataworks 上开发离线计算作业一样简单。

(1) 实时作业开发

新作业的整体开发流程分为定义输入源、定义输出源和定义计算逻辑三个部分，定义数据源只需要执行一次全局可用，无需每个作业都重复定义。为了屏蔽底层复杂的实时计算语义理解问题，我们设计更简化的 UDF，让开发更简单，例如：

- window_row(amount)，窗口函数，按时间排序取最近 amount 条行为。
- window_time(timecloum，timeunit) 按照单位时间进行指标序列聚合

（2）实时数据查询

传统方案特征查询都是使用存储引擎对应的 client 来查询，不同引擎查询方法都不一致，这里我们使用 Dolphin SQL 屏蔽底层查询引擎，使用统一的 SQL 语法查询，不仅降低开发成本，还让特征 debug 调试更简单，调试完的 SQL 可以直接在线上使用。

很多场景都存在实时特征和离线特征一起作为请求入参，一般都是分别查询实时和离线特征再拼接，为提升效率，Dolphin SQL 支持在 SQL 查询就可以让实时特征和离线特征 join 实现参数拼接，极大提升开发效率。

```
SELECT a.id, a.action_list, b.city, b.level
FROM realtime_feature_table a
JOIN offline_feature_table b ON a.id = b.id
```

4.3.2 作业调度

Dolphin Streaming 通过 openApi 进行作业进行调度运维，包括：

- 作业创建、执行计划生成、作业启动
- 作业停止、作业暂停、作业状态查询

4.3.4 Debug 功能

传统使用 Flink debug 都是写一个 job 打印数据记录，需要写一个完整的 job，效率较低。为了让用户 debug 更简单支持用户使用 select 语句探查实时数据源表，进行快速 ETL 开发，**无需提前定义**上游源数据，相较 Flink 非常高效。

5. 应用示例

作为整体端到端方案，我们实现极光开发平台打通 Dolphin Streaming 能力，可以直接在极光开发平台（极光是阿里妈妈商家端数据开发平台）进行数据管理、实时作业开发、debug 调试及运维管理，让实时数据问题在这里一站式解决。

5.1 数据开发

用户直接基于上游中间层实时数据进行开发，定义好输入输出源，然后定义计算逻辑即可，上游 TT 的 subId，accessId，accessKey 信息都在 SQL 转译阶段生成，用户无感知，下游存储表信息也都是自动创建和生成。

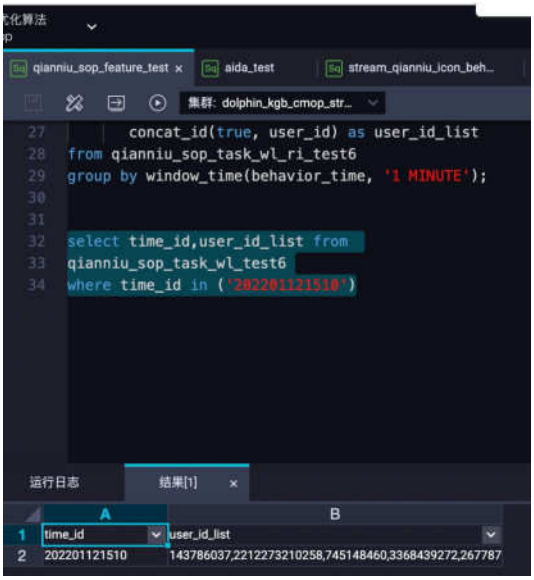
```
create table task_input_wl_ri_test
(
  user_id String,
  task_id_pv String,
  task_id_ipv String,
  behavior_time String
) with(
  bizType='tt',
  topic='task_wl_ri_test',
  pk='user_id',
  timeColumn='behavior_time'
)

create table task_output_wl_test (
  time_id String
  user_id_list String
) with (
  bizType='feature',
  pk='time_id'
);


insert into task_output_wl_test
select window_start(behavior_time) as time_id,
       concat_id(true, user_id) as user_id_list
from task_input_wl_ri_test
group by window_time(behavior_time, '1 MINUTE');
```

5.2 特征数据查询

开发好特征数据之后可以直接查询实时结果，验证计算逻辑，如果没问题该查询 SQL 可以在线上直接查询 Dolphin 使用。



5.3 数据探查

当上游 TT 表已经提前注册好，直接像数据库一样 select 查询表就可以实时获取探查结果，在数据 debug，数据探查方面非常实用高效。



6. 业务收益

Dolphin Streaming 支撑商家端算法实时特征开发，包括阿里妈妈直通车、引力魔方成效预估及如意货品推荐等场景，已取得显著收益。

- 支持客增如意货品推荐服务顺利经历本次**双十一**大促考验，通过对客户行为序列的精细化兴趣建模，应用到 5+ 场景，实现拉新、活跃和新建计划客户数增长显著，整体实时特征在线查询 QPS 达到 6000+，实现 Dolphin 引擎查询业务量翻倍增长。
- 通过引入实时广告效果数据，支持万相台 MCB、千牛小程序及直通车智能计划等 3+ 场景成效预估，其中万相台 MCB 带来 cost 及 ARPU 值显著提升。

7. 总结

在大规模实时场景，数据开发需要专业的工程技术同学支持，优势是可以达到极致性能；但在大多数普通规模实时场景中，如何简单高效的开发、迭代、测试和上线是业务方更关注的因素。

通过 Dolphin Streaming 提供面向算法等非工程同学的实时开发 **DB for Streaming** 解决方案，实现从数据获取、特征开发到特征上线整个流程**一体化**完成，在广告商家端算法场景不仅节省了算法到工程之间的沟通时间，还降低了开发人力成本，让业务迭代效率更高，实现规模化提升实时研发效能。

用户使用越简单，关心的内容越少，越是需要**背后大量的用户理解和研发工作**，未来我们会继续以**用户体验和研发效能为中心**，用技术提升商家经营增长。

alimama · tech



欢迎微信关注「阿里妈妈技术」

联系我们：alimama_tech@service.alibaba.com