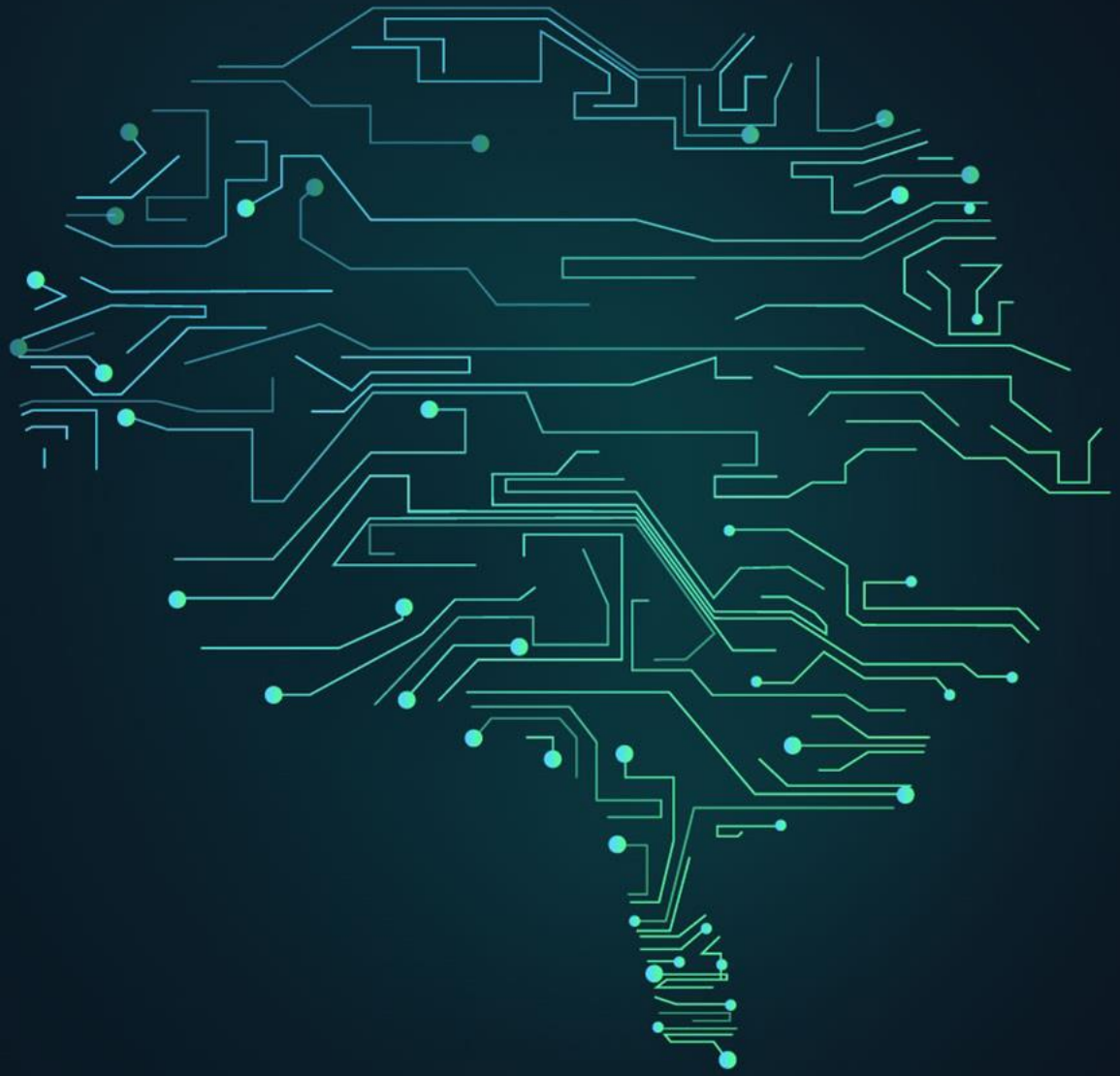
深度学习模型质量



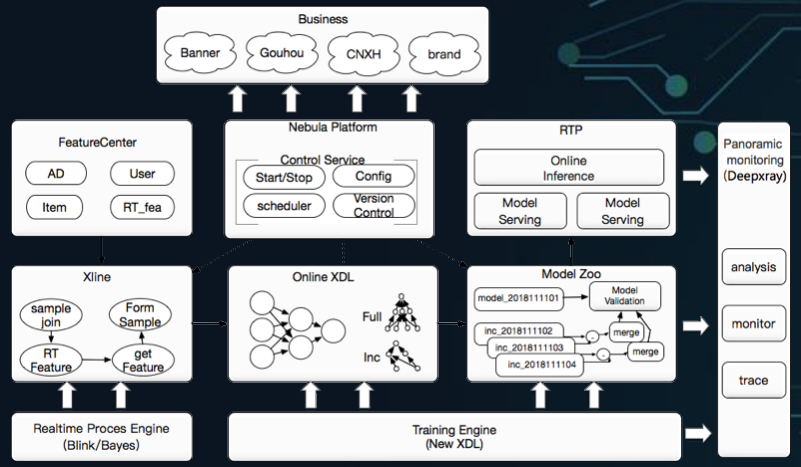
主讲人：有虑
阿里巴巴高级开发工程师

目录
Contents

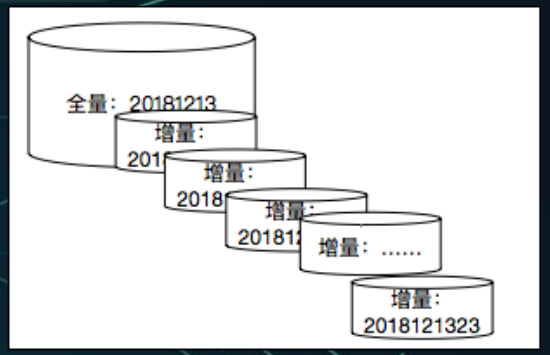
01 章节 PART

算法质量的挑战

计算广告领域--算法质量的挑战



大规模数据



✖ 50+模型

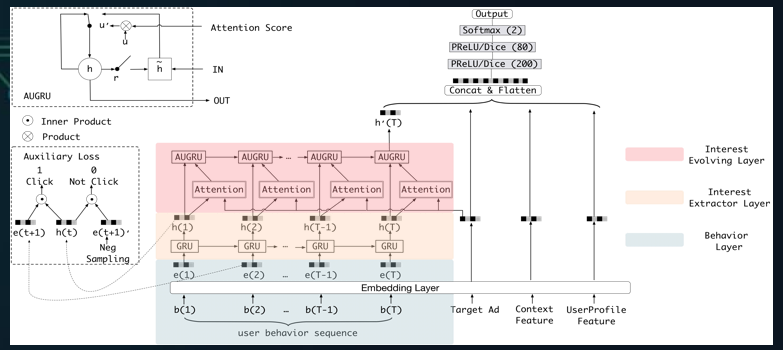
千亿样本　万亿特征　TB模型

模型拆分与融合

广告搜索与定向全业务迁移

端到端实时化演进到组件实时化

实时化系统 ⟷ 黑盒化过程



丧失功能独立评估能力

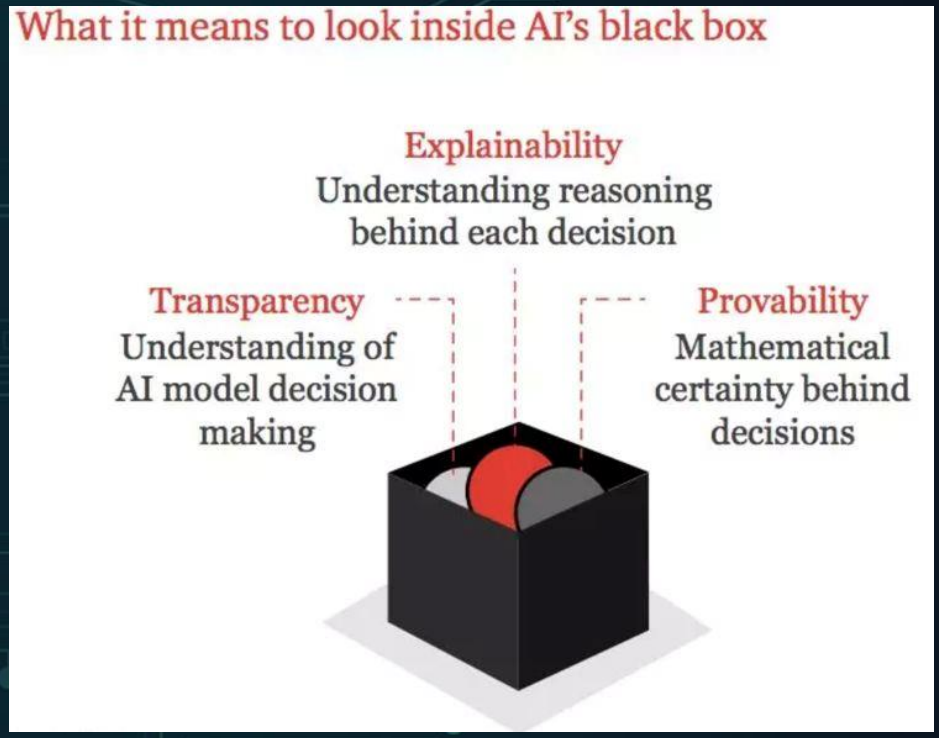
前沿技术/风险未知

章节 PART 02

数据透出与可视化

AI的“黑盒”特性

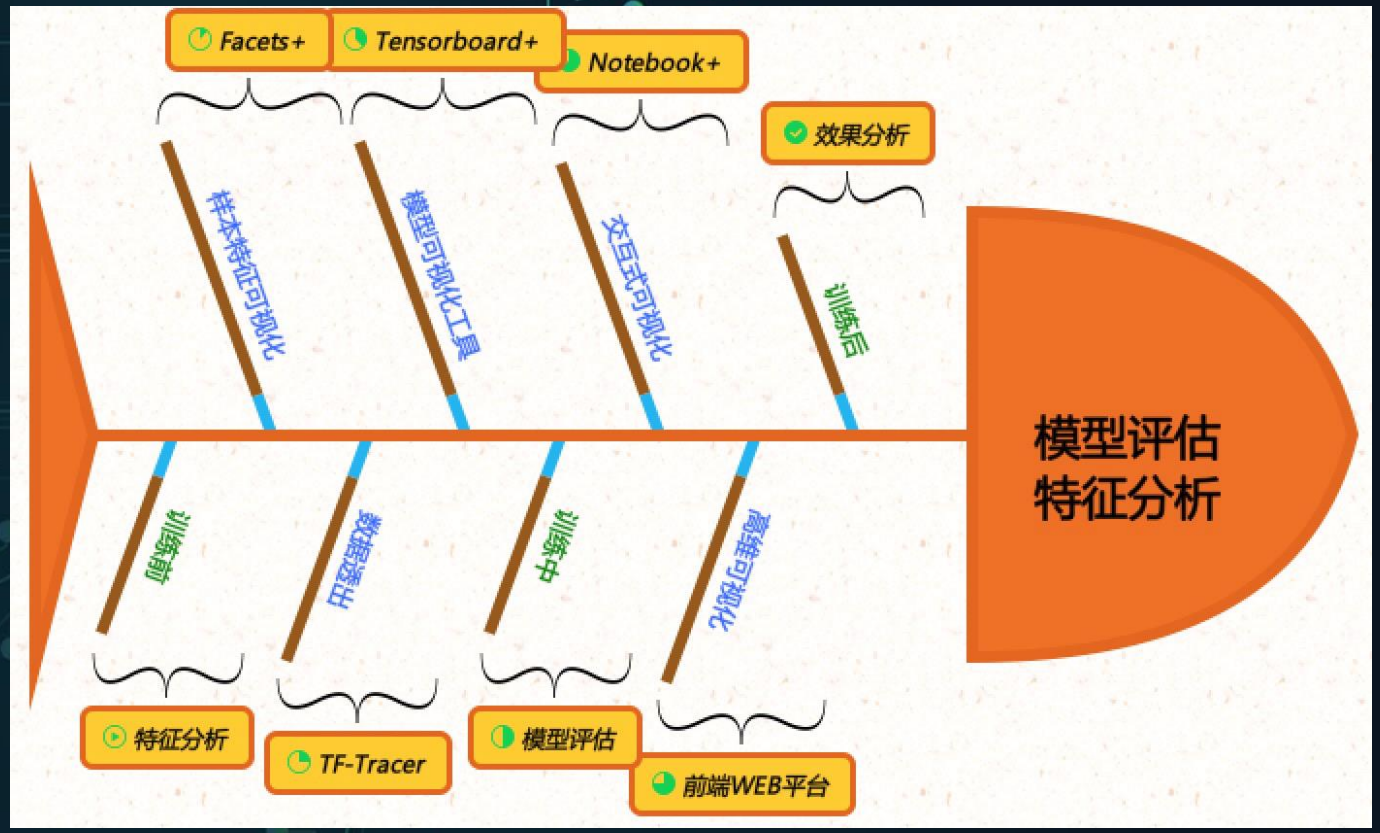
- 深度网络对机器学习研究和应用领域产生了巨大的影响，与此同时却无法很清晰地解释神经网络的来龙去脉；
- 迄今为止，深度学习不够透明，神经网络整体看来仍然是一个黑箱。因此，人们一直致力于更透彻地去理解其中复杂的过程，从而达到进一步优化的目的；
- 由于人类对于世界的认知和感受主要来自于视觉，良好的可视化可以有效地帮助人们理解深度网络，并进行有效的优化和调节；



（引自 2018 AI predictions：8 insights to shape business strategy）

基于数据透出的多维度可视化分析

- 深度学习组件(TF-Tracer/TF-Profiler等)主要负责数据透出、实时监控和输出控制等，透出的数据主要是模型训练过程中的Raw Data(未经加工或计算统计)；
- 后端微服务(Tensorboard+/Notebook+/Facets)和前端WEB平台(高维可视化)负责相关数据的在线\离线\交互式等方面的可视化分析评估；
- 以模型训练任务的生命周期管理(Lifecycle)贯穿始终，从而形成多维度可视化分析的生态循环；



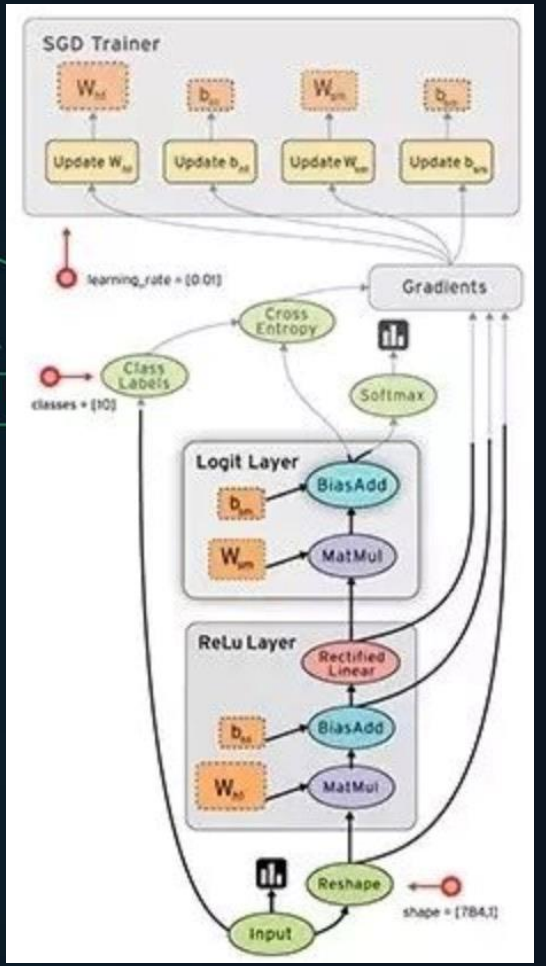
TF-Tracer：基于计算图的全面数据透出

- 基于原生的深度学习框架Tensoflow API(tf.train.SessionRunHook)编写的即插即用组件，无需用户开发额外的代码，只需用户通过配置文件增加相应配置信息，即可使用对应的组件；

- 配置信息由组件开关和组件配置参数信息两部分组成：
  - 打开相应组件开关后，组件功能不会对原有训练任务产生较大性能影响，从而保证线上训练效率；
  - 关闭相应组件开关后，对原有训练任务无任何功能或性能上的影响；

- 数据透出组件TF-Tracer是基于Tensorflow 计算图(tf.Graph)开发的，可以全面透出计算图中的所有变量(tf.Variable)；

- 基于图数据集(tf.GraphKeys)，通过正则表达式对变量集合进行匹配过滤，透出相应变量数据集合，同时也支持直接指定变量列表进行数据透出，支持NumPy/Bin两种数据格式输出；



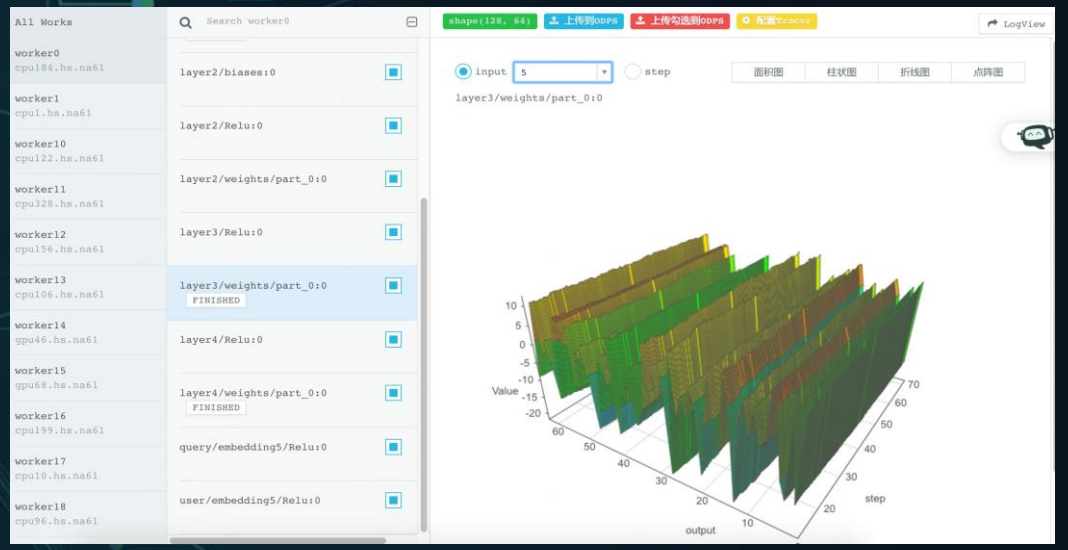
在线动态更新数据集实时透出

- 对于运行周期较长的模型训练，如Online Learning，在训练过程中发现异常的话，有时很难根据现有透出数据进行定位；而重启任务更新透出数据集的话，有些问题并不能复现（深度学习统计学特性）；

- TF-Tracer在不重启训练任务的前提下，支持在线动态更新透出数据集；

- 针对于cheif_only等于False的情况，即所有worker都透出数据时，支持指定worker动态更新，非指定worker透出数据不变；

- 右图：在线修改实时透出实例：变量从layer2/biases:0和layer2/weights/part_0:0到变量layer1/weights/part_1:0和layer4/weights/part_0:0



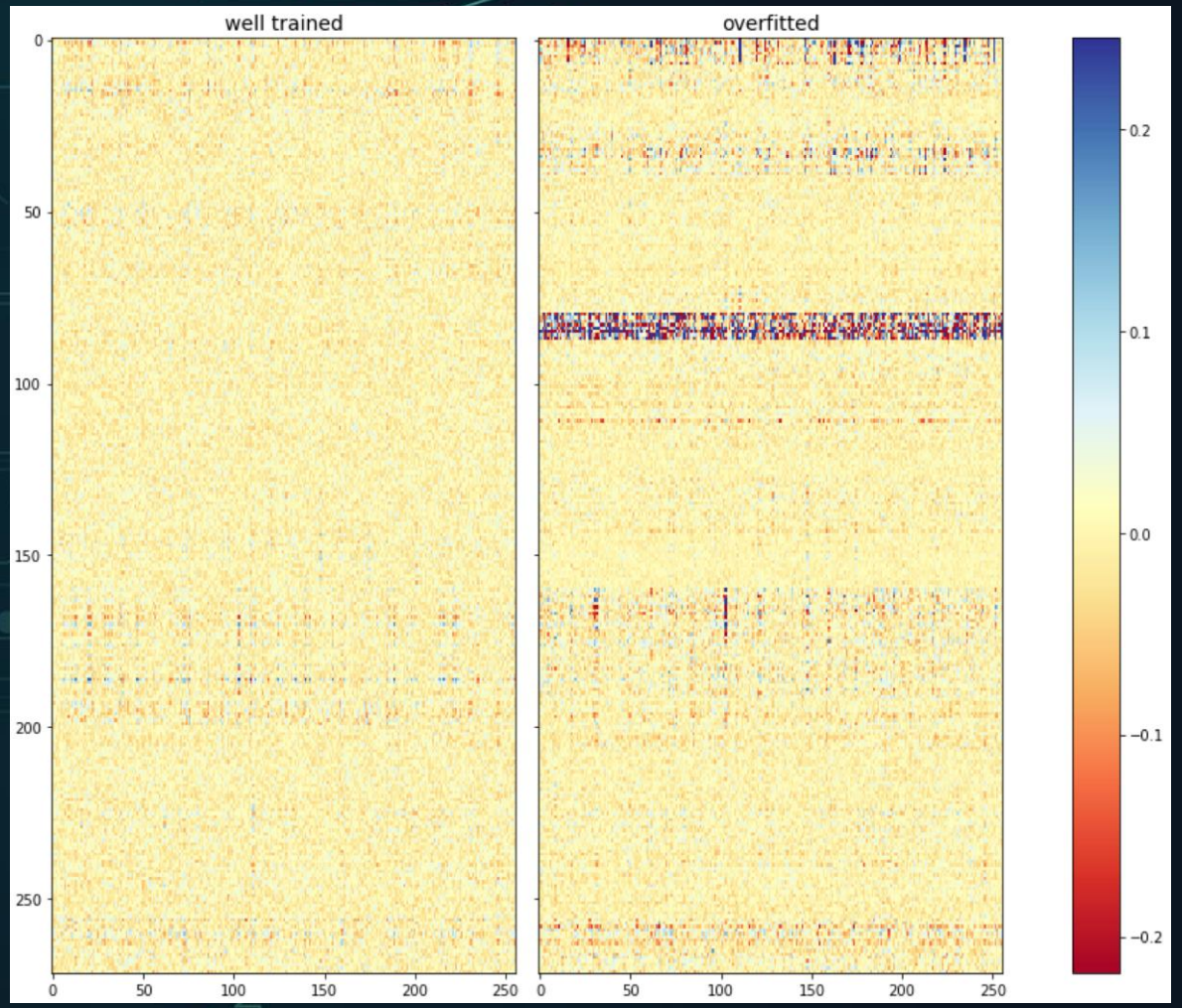

```
0.02270402]]
=========
[2018-03-01 14:51:45.788640] L[5400] G[118579] Qps[r:3869.06 l:2854.24 g:62676.46] Loss[0.17] Auc[0.6857]
[2018-03-01 14:51:45.818581] L[5401] G[118594] - trace data saved
tensor list:
--------
name:layer2/biases:0
dtype:float32 <type 'numpy.ndarray'>
shape:(128,)
value:
[ 0.01790209 -0.03666655  0.09270628  0.09151244 -0.04292365 -0.04259959
  0.08133119 -0.15516819  0.08246493  0.00069321  0.03735797  0.15530743
  0.01147886  0.05465052 -0.01667915  0.06843525 -0.0620437   0.12660104
 -0.13091807  0.04555457  0.00879552 -0.22227697 -0.00028289 -0.19268437
 -0.03175865  0.16655786 -0.18687147 -0.03417608 -0.08933324  0.07192472
  0.05357097 -0.04210736 -0.1145954  -0.03322025  0.0110665  -0.07485788
 -0.05734688  0.08138344  0.09226588  0.19552371  0.05838733  0.16474128
 -0.00281576  0.1499038  -0.10360274  0.02912908 -0.13552187 -0.08449893
 -0.14990972 -0.09576733 -0.0893248  -0.13229293  0.14485386 -0.00402579
 -0.05163569  0.07389592  0.01358198  0.09513265  0.09616991  0.00461894
 -0.07099531 -0.0559599  -0.11313137  0.12076628 -0.13767879  0.03393555
 -0.08958647  0.10032403  0.17202985  0.17546873  0.13662831  0.17085299
 -0.00429067  0.08333712  0.07270689  0.10470887 -0.00390323  0.07418979
  0.00823762  0.05768789  0.18099253 -0.00062893  0.09868123  0.08804181
 -0.03040639 -0.03628076 -0.17555773  0.12352357 -0.08225826 -0.07849138
```

章节 PART 03

模型评估

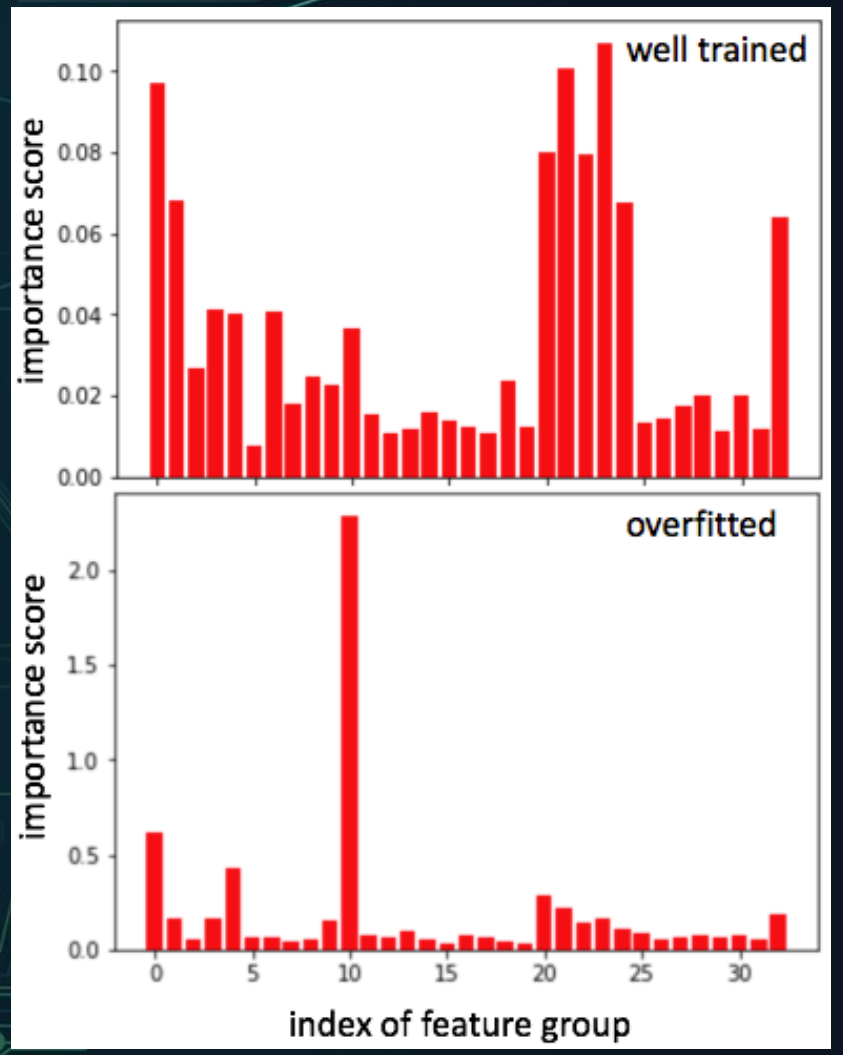
模型评估 -- 通过边权重了解模型训练状态

- 边的权重是神经网络模型的主要成分，因此分析边的权重可以对模型状态有一个最直接的了解；
- 模型权重分布异常是模型训练出现异常的一个重要表征，表明可能出现过拟合或者训练数据存在问题；
- 右图，过拟合的模型中，边权重的大小分布很不均匀，出现了大量权重（绝对值）极大的边，且集中在一条带状区域内；
- 这一条带状区域为某一组特征输入所连接的所有边，表明模型过度拟合了该组特征的信息。我们更进一步发现，无法通过正则或dropout来防止这种过拟合状况并提升模型效果，这表明问题出在该组特征输入上；



模型评估 -- 通过梯度了解特征输入的影响力

- 将输入层的某个神经元状态针对模型输出预估分求导，该梯度的强度表示了模型的预估输出对于该输入的微小变动的敏感度，从而可以反映出该输入对于模型的影响力；
- 梯度越强，表明该输入的对模型的影响越大；
- 右图，对比了两个不同状态的模型中（未过拟合 vs 过拟合，与上页展示的两个模型一致），由梯度求得的各个特征组的平均影响力；
- 可以清晰地看到两个状态的区别，过拟合的模型过度看中少量几组特征，尤其是编号为10的特征组（这正对应了上页图中的出现异常边权重的带状区域）；



章节 PART 04

特征分析与选择

特征分析 -- 面向高维ID化稀疏数据

特征效用分析（衡量有效性、相关性）

- 直接指标：iv/woe、缺失率、高频特征、特征离散度、分布变化率、取值/结构相关性等
- 间接指标：通过特征处理算子处理高复杂特征，完成效用分析

辅助特征设计（**预测特征效用**）

基于特征中心维护的三维拓扑的两种辅助手段：

- **无需提取数据**直接推测新特征效用等指标
- 相似相关特征给予用户灵感，高效快速设计新特征

高效特征选择（优化特征，提升模型性能）

- 基于特征长期质量得分过滤低质量特征
- 基于特征效用分析筛选与目标高相关的特征
- 结合DropRank/BenchMask/ScoreGap/新增特征效用等手段

特征关系挖掘（**推荐新特征**）

- 特征数值关系：皮尔逊系数等衡量特征间数值相关性
- 特征设计关系：组合特征间根据原料及组合方式衡量设计相关性
- 特征结构关系：比如基于Tree/Graph提取的特征，存在相邻/父子/兄弟等结构相关性

特征分析 – 特征溯源 / 风险预估

影响面分析:

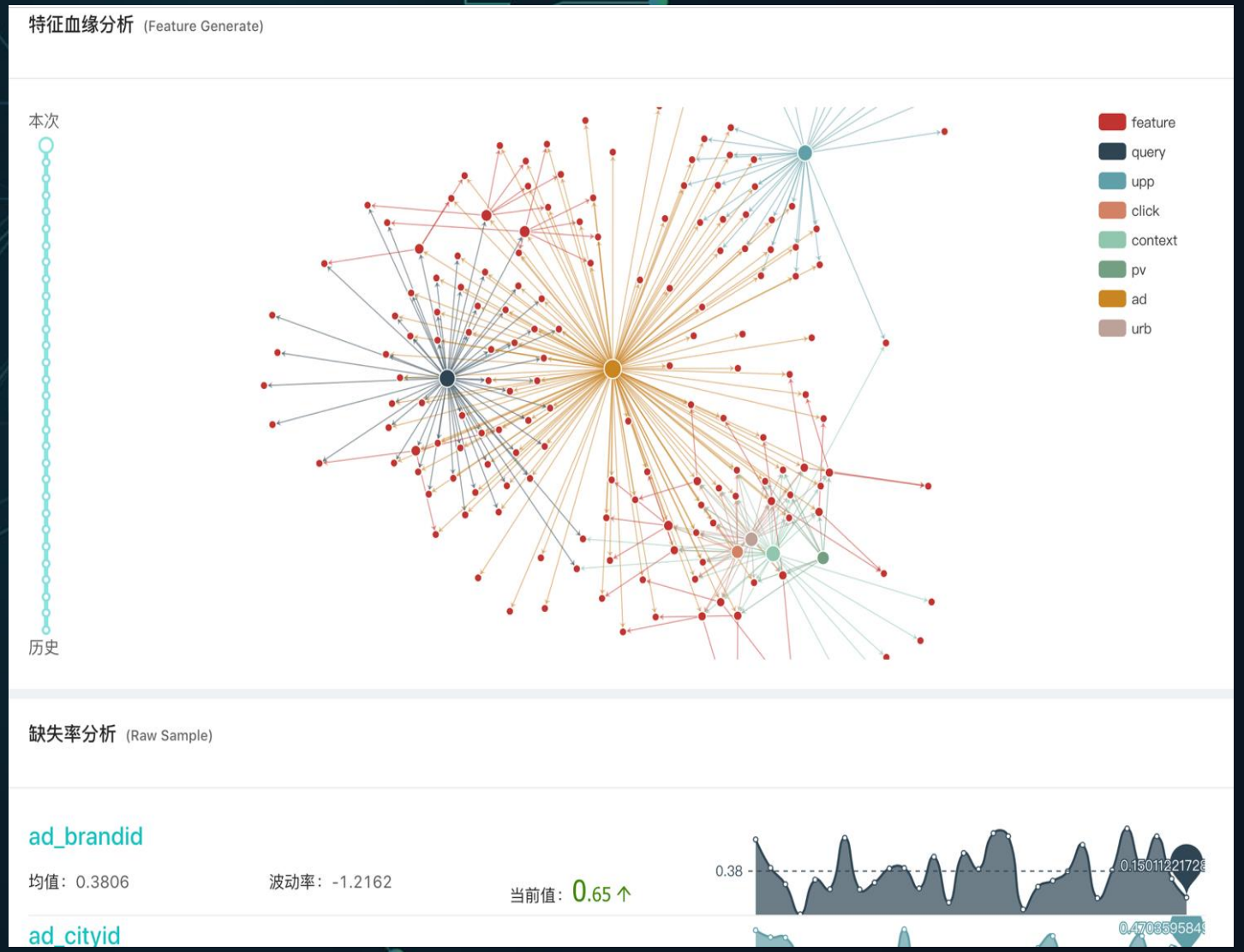
- 通过特征中心构建三维拓扑分析本次异常影响的特征 / 模型;
- 通过自身历史结构变化，预估风险;

影响路径获取:

- 通过有向图中的节点关系得到影响路径;

影响力分析:

- 基于拓扑图，结合辐射步长（红圈1步 / 橙圈2步 / ..）和特征相关性预估风险大小;



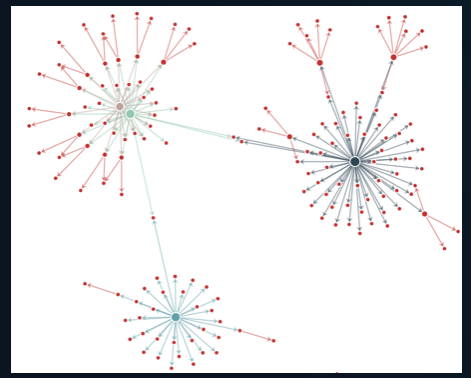
全量特征池 -- 建设健康特征生态圈

降低系统负担 特征有进有出

随着更多业务的接入，引入特征效用相关数据用于特征淘汰（效用持续过低、离散程度过高等），为特征中心维护健康生态提供更强大的数据支撑（传统无效特征淘汰通常基于使用率/缺失率/方差过低等统计值）。

全量特征池

特征关系可视化



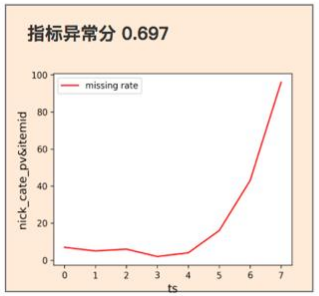
特征淘汰与预警

特征名称	特征效用分	特征使用率	操作
nick_cate_pv&itemid	0.001	0.000%	联系owner/删除
nick_shop_click	0.001	0.000%	联系owner/删除
nick_item_pv	0.207	5.330%	联系owner/删除
...	-	-	-
nick_cate_pv_14	0.875	66.67%	联系owner/删除

详细信息辅助决策

当前特征 nick_cate_pv&itemid

风险指标	指标异常分
缺失率	0.697
效用分	0.432
离散程度	0.501
...	-
分布变化率	0.275



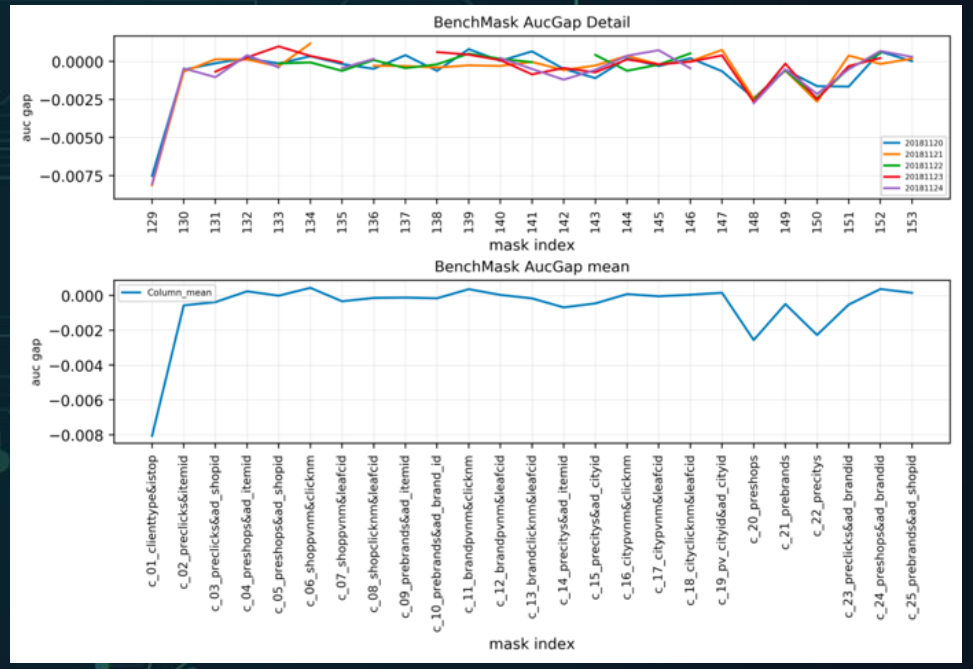
特征选择

- 改进了DropRank、BenchMask算法进行特征删减实验，新增特征效用分析、特征误导贡献，引用了Feature GAUC等方法，实现高效的特征优化实验，这些方法目前在大规模深度学习训练上完成了30+特征的删除；

改进BenchMask

- **一次预测任务实现多组特征评估缺失影响**
- 改进：采用评估任务替代训练任务，

结合随机mask并流程化，产出AUC_GAP、预估分的（严格一致率、STD、AVG等指标），结合三维拓扑的相关关系，进行特征选择；



特征选择

改进DropRank

- **所有特征的重要度变量直接参与训练**
- 改进：采用伴随式增量训练的方式代替直接介训练。在大规模数据、复杂模型的背景下，这一改变让方案具备可行性；

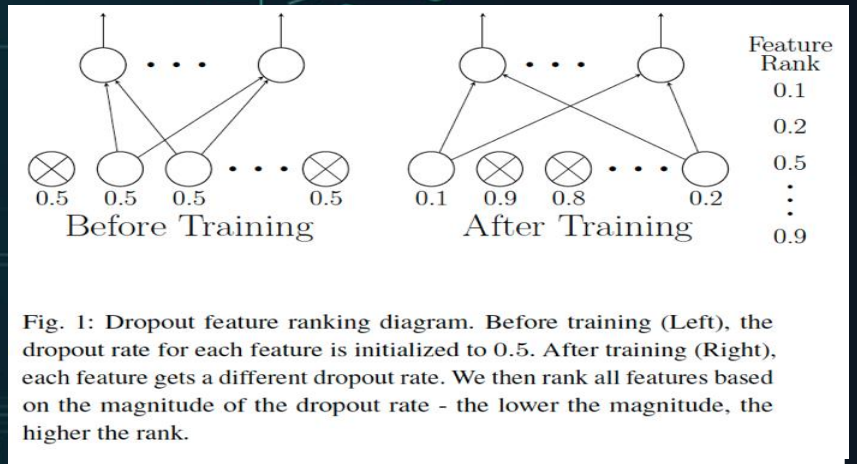


Fig. 1: Dropout feature ranking diagram. Before training (Left), the dropout rate for each feature is initialized to 0.5. After training (Right), each feature gets a different dropout rate. We then rank all features based on the magnitude of the dropout rate - the lower the magnitude, the higher the rank.

$$\tilde{z} = \text{sigmoid}(\frac{1}{t}(\log\theta - \log(1-\theta) + \log u - \log(1-u)))$$

提出特征误导贡献

- **挖掘对模型有负向作用的特征**
- 独创的用于二分类问题的误导评估。基于预估分排序与label是否一致得到新的label衡量误导，然后进行效用分析（IV值），挖掘具备分类误导倾向的特征。

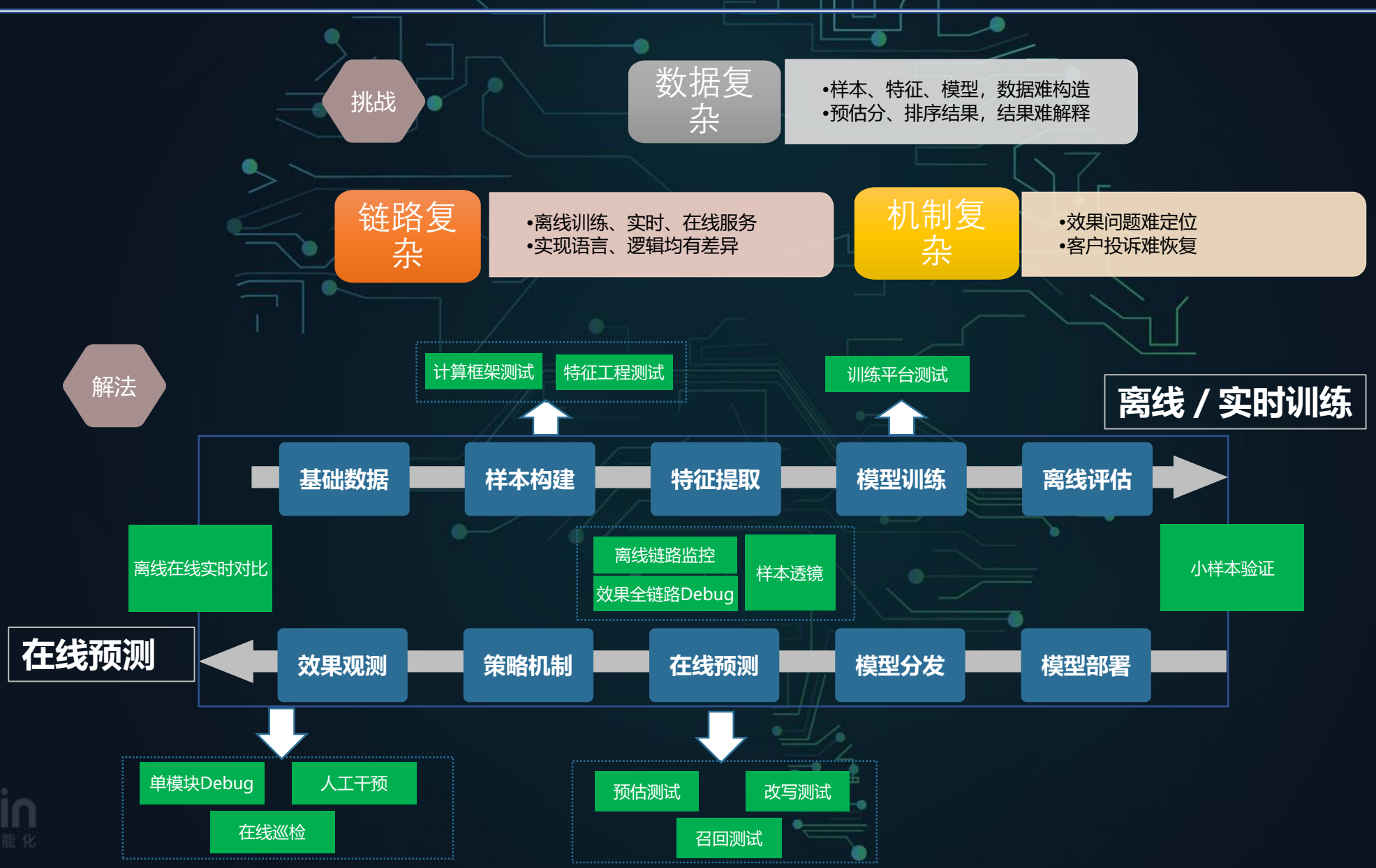
$$WOE_i = \ln(DistributionGood_i / DistributionBad_i)$$

$$IV = \sum (Population_i / Population_{total}) \cdot (DistributionGood_i - DistributionBad_i) \cdot WOE_i$$

章节 PART 05

算法系统的持续交付

持续交付 -- 工程质量



持续交付 -- 问题定义

如何构建算法系统持续交付闭环?如何打造15分钟级别的交付能力?

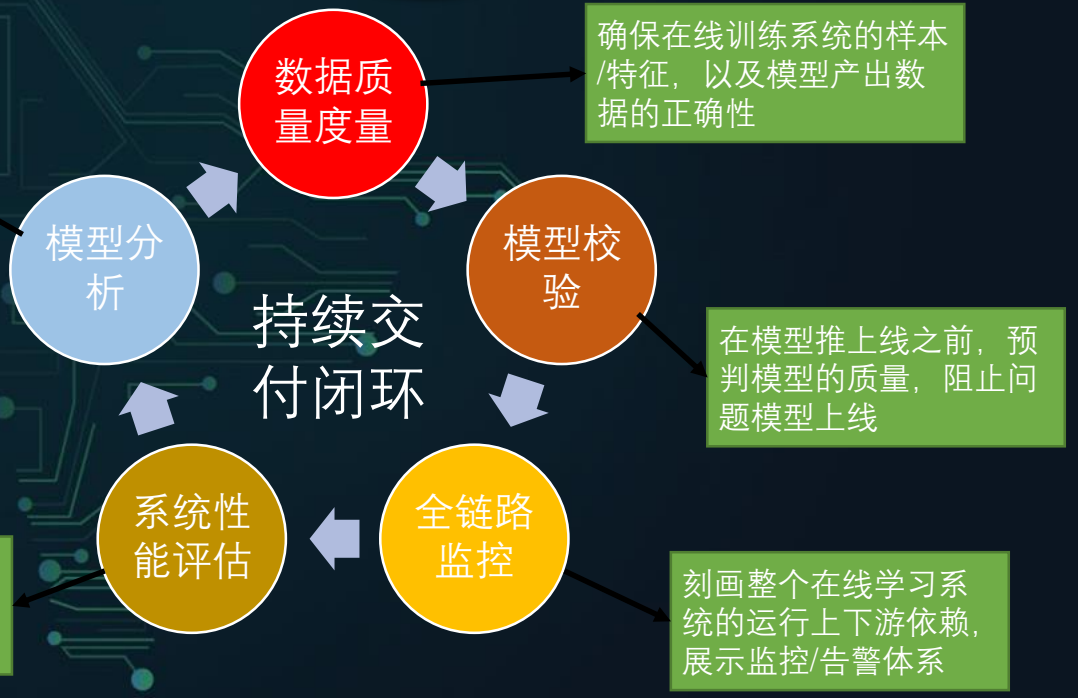
~~工程系统~~
持续交付链路

编译 → 单元测试 → 集成测试 → 灰度发布 → 正式发布 → 线上冒烟

算法系统交付核心：数据 + 模型

特点：高频长链路，线上/线下打通

- 关键词：
  - 检测速度快
  - 检测精度高
  - 全链路覆盖



确保在线训练系统的样本/特征，以及模型产出数据的正确性

在模型推上线之前，预判模型的质量，阻止问题模型上线

刻画整个在线学习系统的运行上下游依赖，展示监控/告警体系

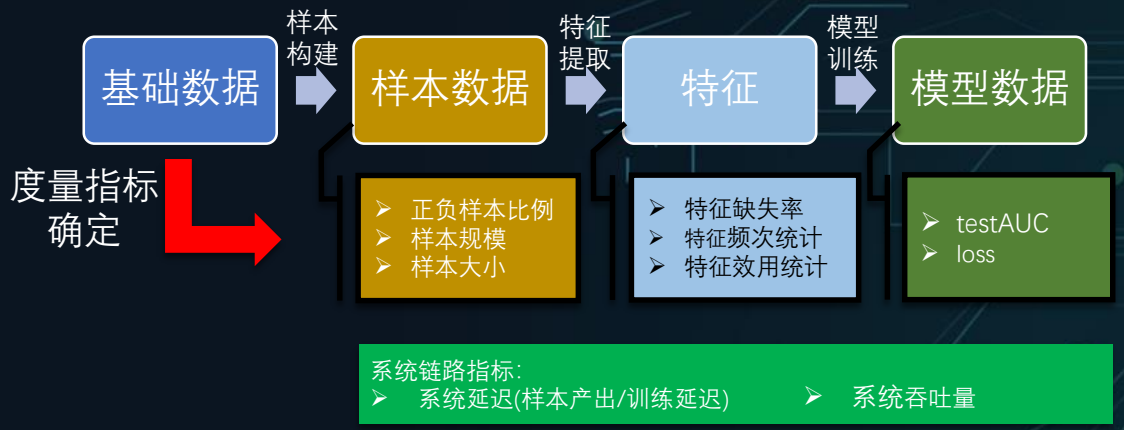
全链路系统容量摸底，系统承载能力预判

系统具备数据透视，数据分析/评估的能力

持续交付 -- 数据质量度量

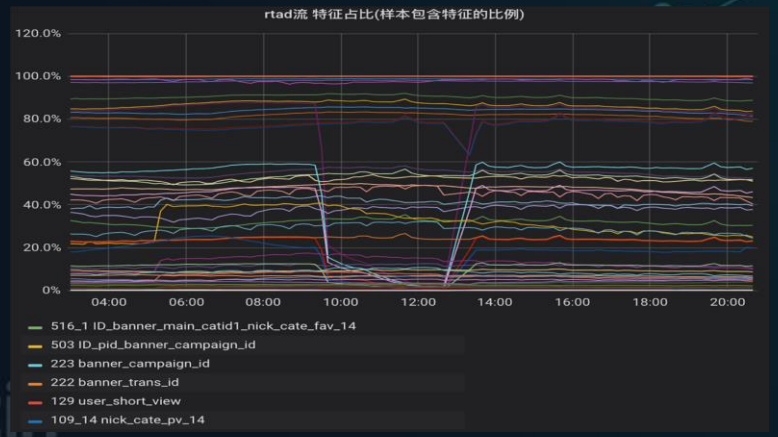
价值剖析:

- 数据是算法系统的核心资产
- 构建算法持续交付的基础



系统链路指标:

- 系统延迟(样本产出/训练延迟)
- 系统吞吐量



构建以模型为核心的时空压缩

持续交付 -- 全链路监控

➢系统挑战：

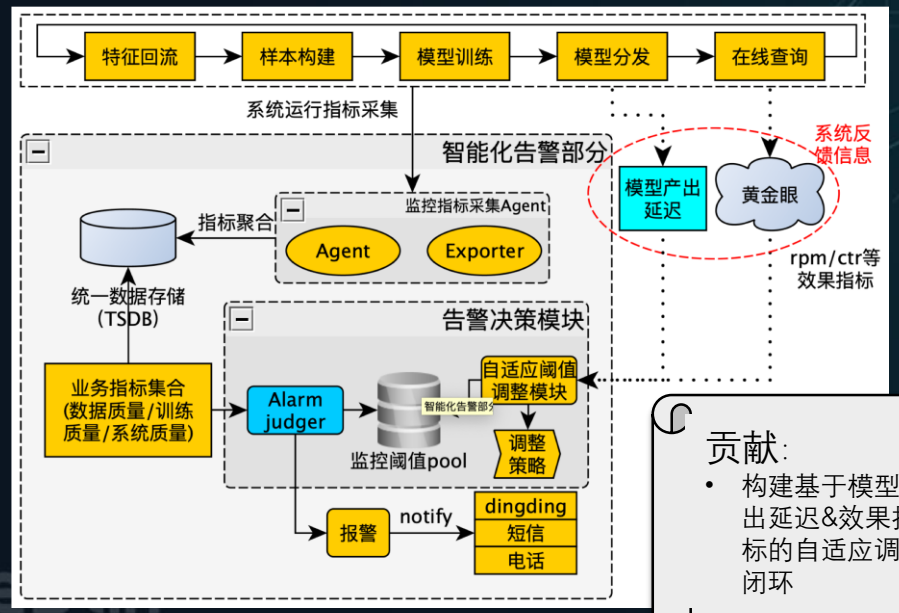
- 实时环境变化剧烈，异常判定规则随时失效
- 系统异常情况下解决告警信息冗余问题

➢解法：

- 智能化告警：构建基于效果反馈的链路，实现告警阈值的自适应调整
- 构建系统拓扑：告警时间片打平，构建告警依赖

智能化告警

- 解决在线学习环境剧烈变化造成的告警误报的问题
- 解决新接入系统的告警阈值冷启动问题

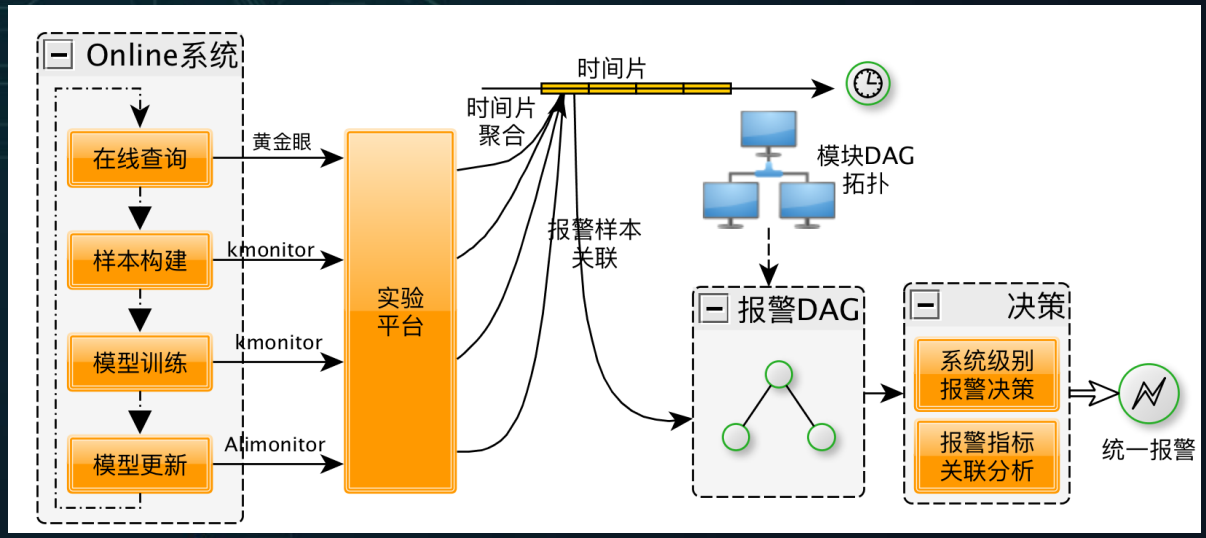


贡献：
- 构建基于模型产出延迟&效果指标的自适应调整闭环

系统级别告警抑制

- 解决多模块依赖，系统级别告警信息冗余的问题

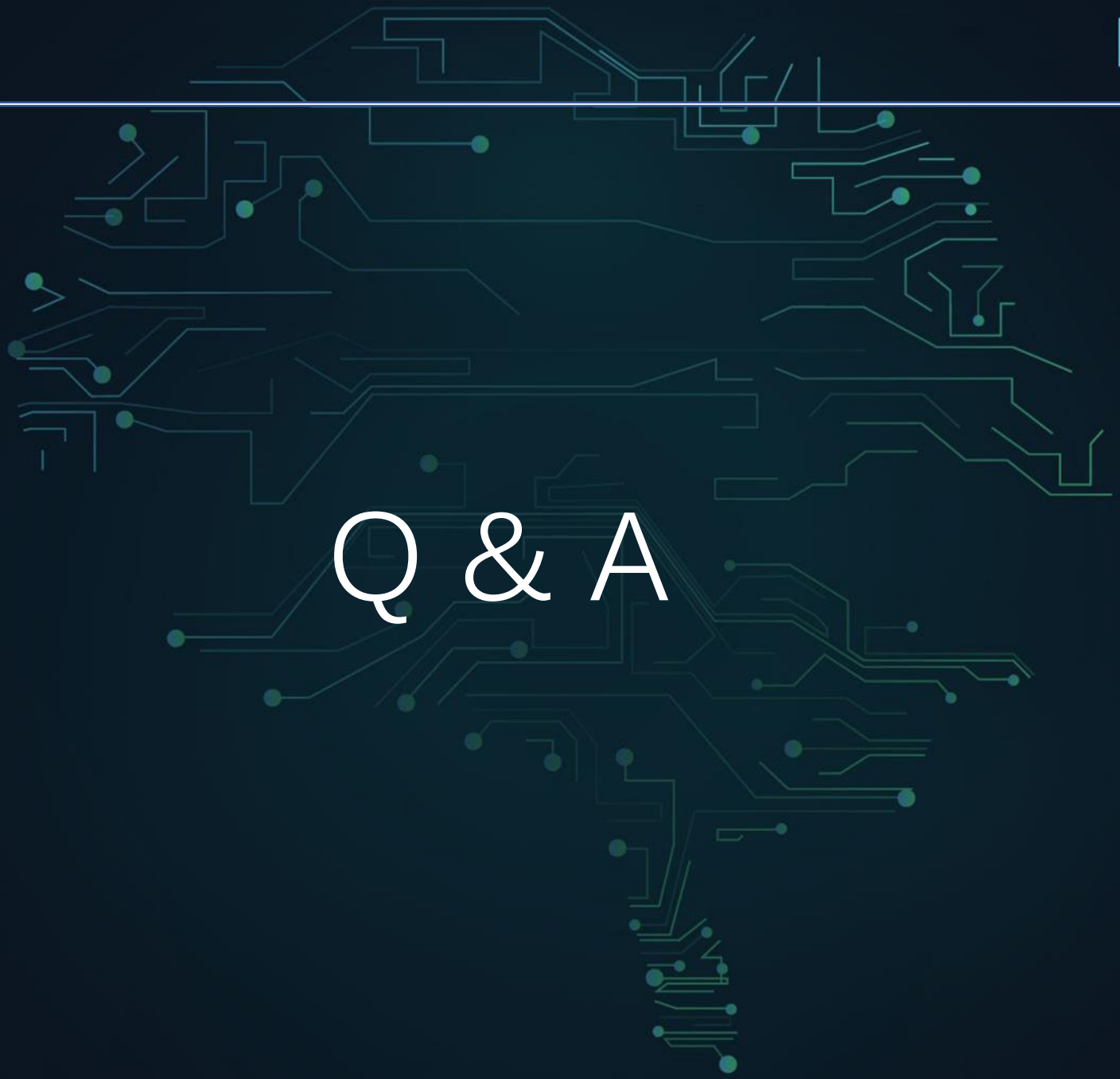
贡献：
- 基于系统拓扑构建告警拓扑
- 告警时间的时间片聚合



团队论文：

- 2018 ACM SIGIR Workshop on eCommerce
  - *Visualizing and Understanding Deep Neural Networks in CTR Prediction*
  - 可视化理解深度神经网络CTR预估模型；
- 2019 KDD Workshop on Deep Learning Practice for High-Dimensional Sparse Data
  - *An Adaptive Approach for Anomaly Detector Selection and Fine-Tuning in Time Series*
  - 基于时间序列异常检测的检测器与运行参数自适应选择；
- 2019 KDD Workshop on Deep Learning Practice for High-Dimensional Sparse Data
  - *AMAD: Adversarial Multiscale Anomaly Detection on High-Dimensional and Time-Evolving Categorical Data*
  - AMAD：基于对抗网络与多层次表征学习的大规模稀疏数据时间演化数据异常检测；
- 2019 EENMF
  - *An End-to-End Neural Matching Framework for E-Commerce Sponsored Search*
  - 一种商业搜索的端到端匹配框架

欢迎关注
阿里巴巴经济体
技术质量公众号



Q & A

THANK YOU

感 谢 聆 听

Testin
助力产业智能化

NCTS 第二届 中国云测试行业峰会