

展现下单率模型

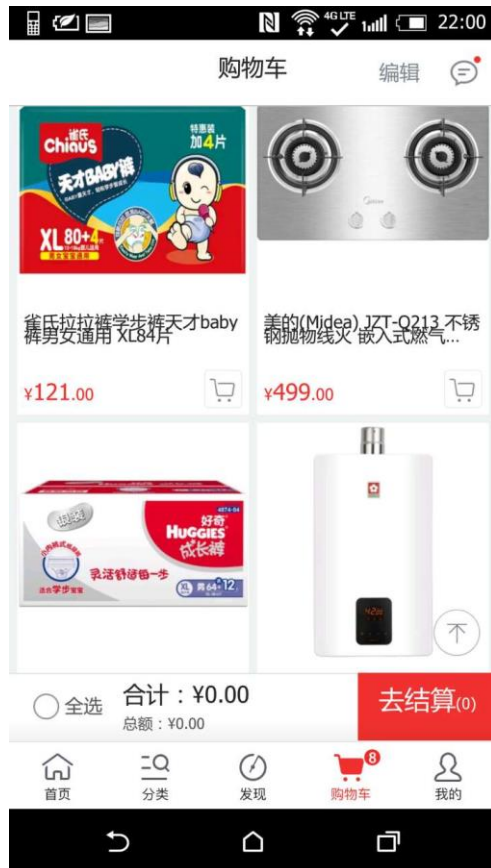
ads-recommend@jd.com

郭文涛

内容

- 应用场景
- 问题建模
- 算法实现
- 算法效果
- 后续工作

应用场景



应用场景

推荐配件

优惠套装

最佳组合



金士顿(Kingston)骇客神条 Fury系列 DDR3 1600

¥279.00



安钛克 (Antec) 额定450W VP 450P 电源 (主

¥259.00



九州风神 (DEEPCOOL) 玄冰400 CPU散热器

¥99.90



技嘉 (GIGABYTE) B85M-D3H主板 (Intel B85/

¥499.00



英特尔 (Intel) 酷睿四核 i5-4590 1150接口 盒

¥1389.00



西部数据(WD)蓝盘 1TB SATA6Gb/s 7200转64M

¥329.00

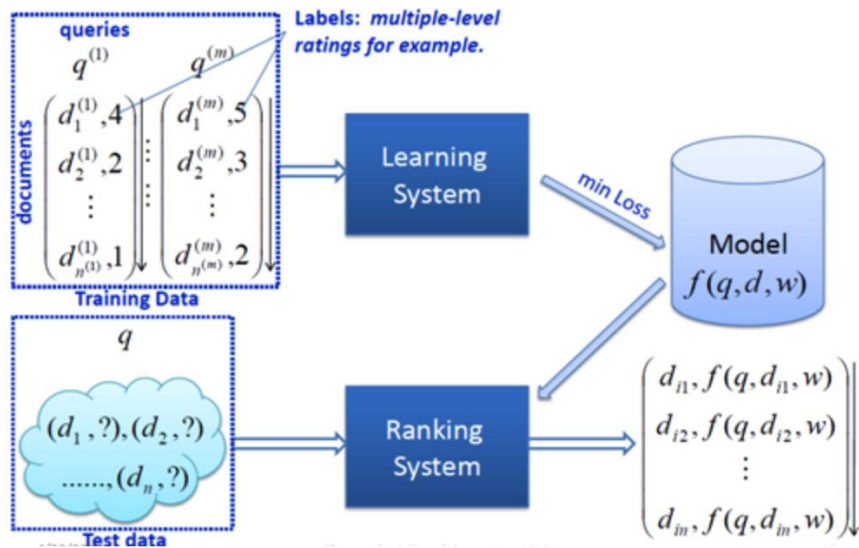
- 共约20+个推荐位
- 推荐场景复杂，至少四种以上
- 每天的流量约？亿，点击量约有1500w

问题建模-传统方案

- 广告系统： $pctr * bid$
 - 建模点击率绝对值
 - 采用point-wise训练
 - 评估auc
 - 线上评估：收入，点击量等
- 推荐系统/搜索引擎：相关性排序
 - 建模相对序关系learning to rank
 - 评估dcg，ndcg
 - 训练样本多采用pair-wise
 - 电商中线上评估：gmV，订单量

问题建模-传统方案

- Learning to rank
 - 框架 https://en.wikipedia.org/wiki/Learning_to_rank#Pairwise_approach
 - 评估指标 DCG , NDCG https://en.wikipedia.org/wiki/Discounted_cumulative_gain



问题建模-原方案

- 排序公式
 - 自然结果 $P_{ctr} * P_{gmv}$,
 - 广告坑位制插入, 广告内部 $pctr * bid$
 - Gmv 和收入不能很好平衡
- 建模方式
 - P_{ctr} 展现 \rightarrow 点击
 - P_{gmv} 点击 \rightarrow 下单金额, 样本稀疏
 - 训练样本: point-wise 样本

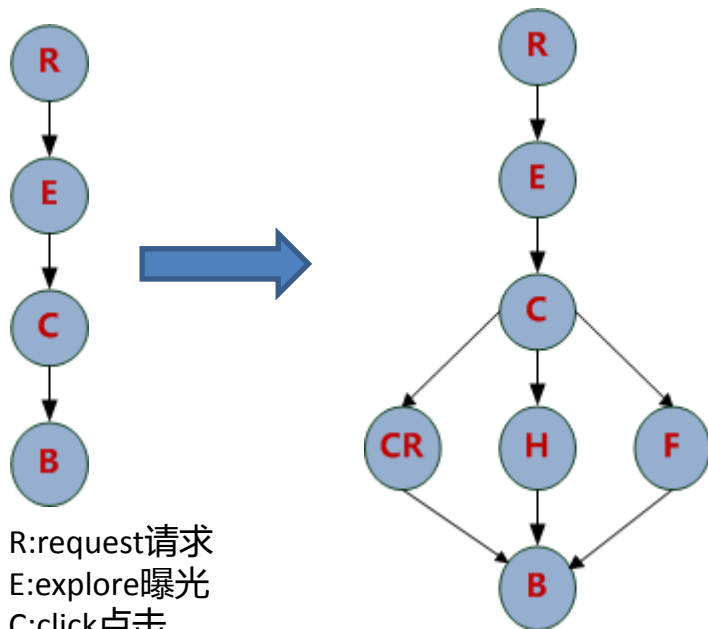


问题建模

- 业务目标：Gmv不降，订单行不降低20%，增加广告收入
- 建模分析
 - 广告系统+推荐系统：既有收入目标，又有转化指标约束
 - gmv，回归模型
 - 订单数，二分类
 - 点击，二分类
 - 转化相关指标不降
 - Point-wise预估绝对值
 - Pair-wise预估相互序关系

问题建模-多指标统一

- 购买路径-概率图模型解释



- R:request请求
- E:explore曝光
- C:click点击
- B:buy 购买
- H:hold 停留
- CR : cart购物车
- F : favorite关注

$$p(B|R) = p(B|C) * p(C|E) * p(E|R)$$

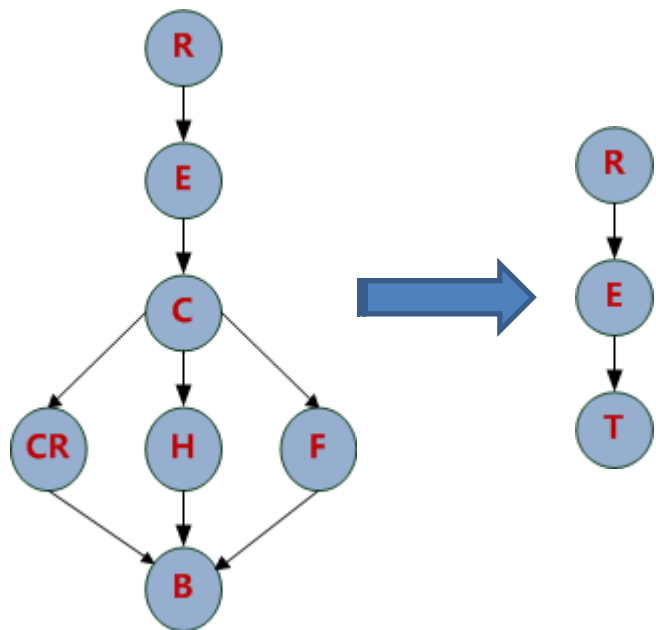
$$p(B|R) = (p(B|F)p(F|C)+p(B|H)p(H|C)+p(B|CR)p(CR|C))*$$

$$p(C|E) * p(E|R)$$

- 建模更逼近真实场景
- 可以刻画点击未购买的样本之间的差异

问题建模-多指标统一

- 购买路径-概率图模型解释



$$p(B|R) = (p(B|F)p(F|C) + p(B|H)p(H|C) + p(B|CR)p(CR|C)) * p(C|E) * p(E|R)$$

$$p(C|E) * p(E|R)$$

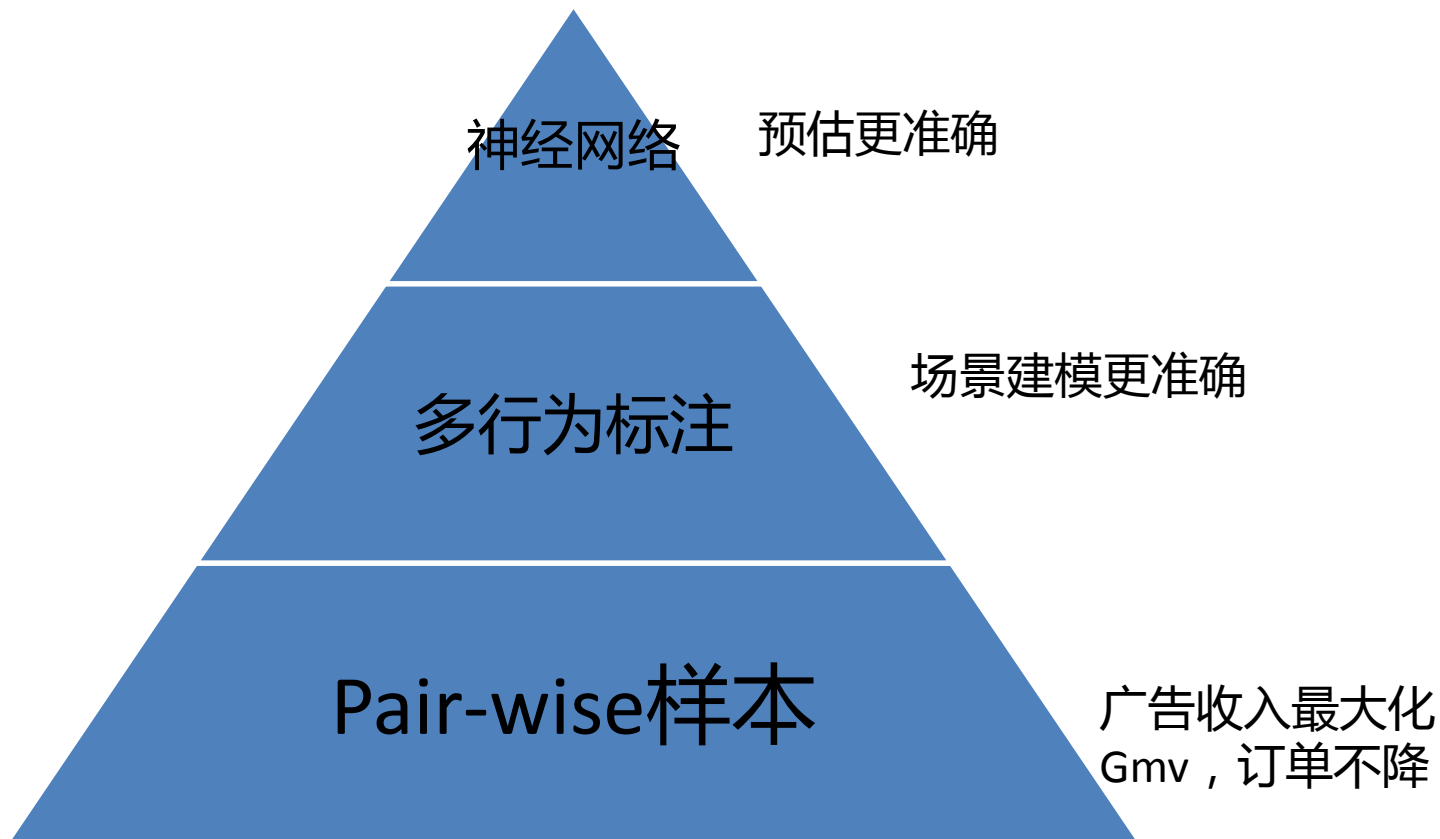


$$= P(T|E) * p(E|R)$$

$$p(T|E) = f(C, H, B, CR, \dots)$$

- 解决多种标注问题
- 扩展更多若标注数据，建模充分
- 解决购买数据稀疏问题，整体优化

问题建模-最终方案



算法实现-损失函数

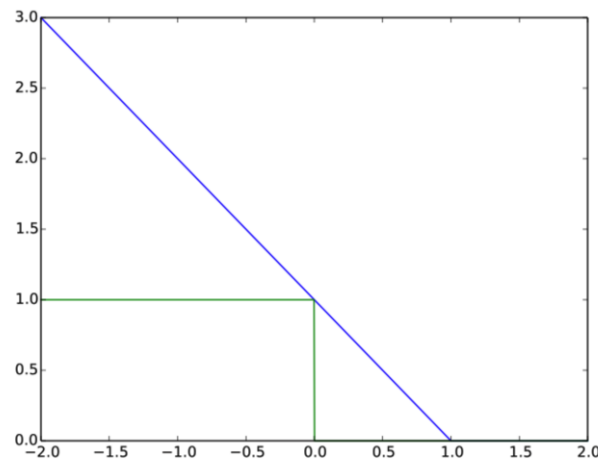
- Pair-wise : 优化序关系
 - Hinge Loss for soft margin (refer svmrank)
 - Regularization : L1 , L2

- Hinge loss For svm

$$f = \max(0, 1 - ty)$$

- Hinge loss For “rank”

$$f_{hinge} = \sum_{i=0}^N \max(0, m - (t_{1i} - t_{2i}) * (y_{1i} - y_{2i}))$$



算法实现-损失函数

- mean squared error : 优化最小误差
 - mean absolute error区别
- MSE Loss For point-wise

$$f = \sum_{i=0}^N (y_i - t_i)^2$$

- MSE LOSS For “pair-wise”

$$f_{mse} = \sum_{i=0}^N (t_{1i} - y_{1i})^2 + (t_{2i} - y_{2i})^2$$

算法实现-损失函数

- Loss function

优化gmv、点击

$$f = \sum_{i=0}^N (t_{1i} - y_{1i})^2 + (t_{2i} - y_{2i})^2 +$$

优化序关系

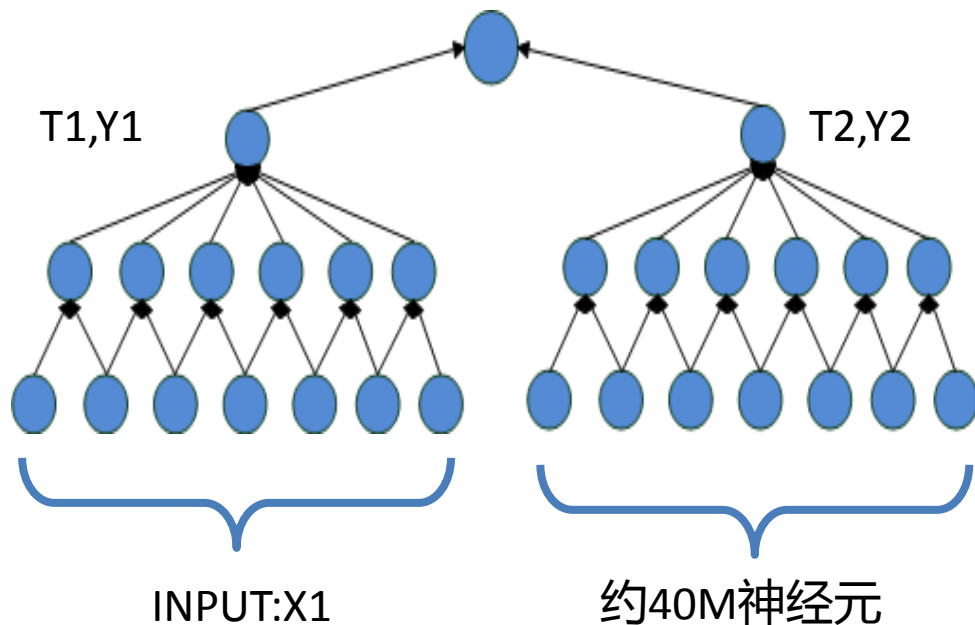
$$\lambda \sum_{i=0}^N \max(0, m - (t_{1i} - t_{2i}) * (y_{1i} - y_{2i}))$$

- 拟合目标

$$t_i = f(click, gmv, action, \dots)$$

算法实现-模型

- 模型结构



- 结构说明

- 左右结构完全相同
- 左右参数共享
- 第一层参数40M
- 激活函数leakly relu
- Sgd训练

- 效果

- 相比使用cnn结构，逆序比降低5%，ndcg提高3%。
- 训练速度是cnn100+倍（dnn多线程，cnn单线程gpu）。

算法实现-样本选择

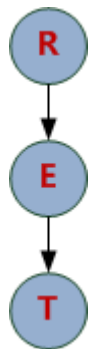
- pair构造
 - Click > Skip above?
 - Last click > Skip above?
 - Click > click earlier?
 - Last click > skip Previous?
 - Click > No-Click Next?
 - Buy > Click >
- 无点击数据利用
 - 直接使用, 无hinge loss只有mse loss
 - $Ad_{nth} > Ad_{(n+m)th}$

算法效果

- Abtest/小流量效果
 - 点击增加 $2.74\% \pm 0.56\%$
 - App gmv 提升 $12.62\% \pm 7.37\%$
 - App 订单提升 $12.13\% \pm 2.43\%$
- 全流量效果
 - 整体效果：appgmv和订单重新建立优势。
 - 减低系统复杂性，单模型代替多模型，节省predictor机器约70%
- 其他
 - 更方便引入更多优化目标
 - 优化目标统一，在线简单

后续工作

- 数据优化



$$p(B|R) = P(E|R) * f(C, B, H, CR, F)$$

真实曝光 更多行为引入

- 算法升级

- 使用统一模型
- 各种维度sku embedding
- Loss functions 优化
- Pair样本构造

Thank You