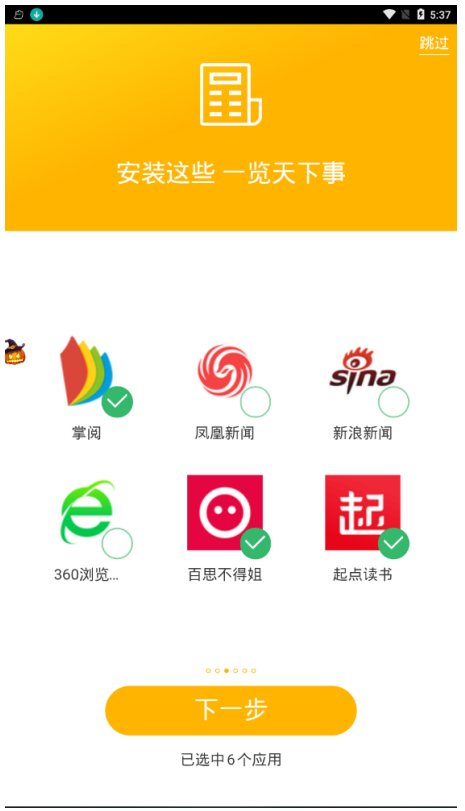
Recommendation System

陈浩

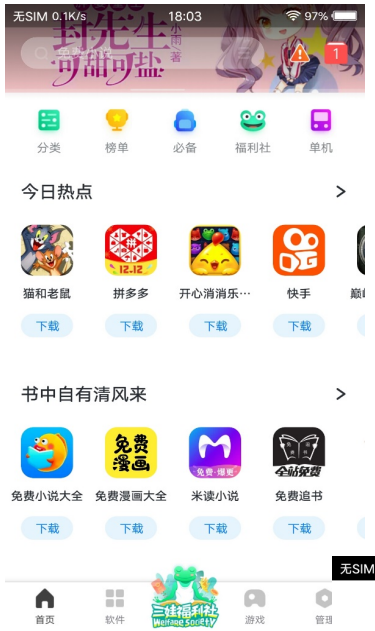
2020-12-11

Example



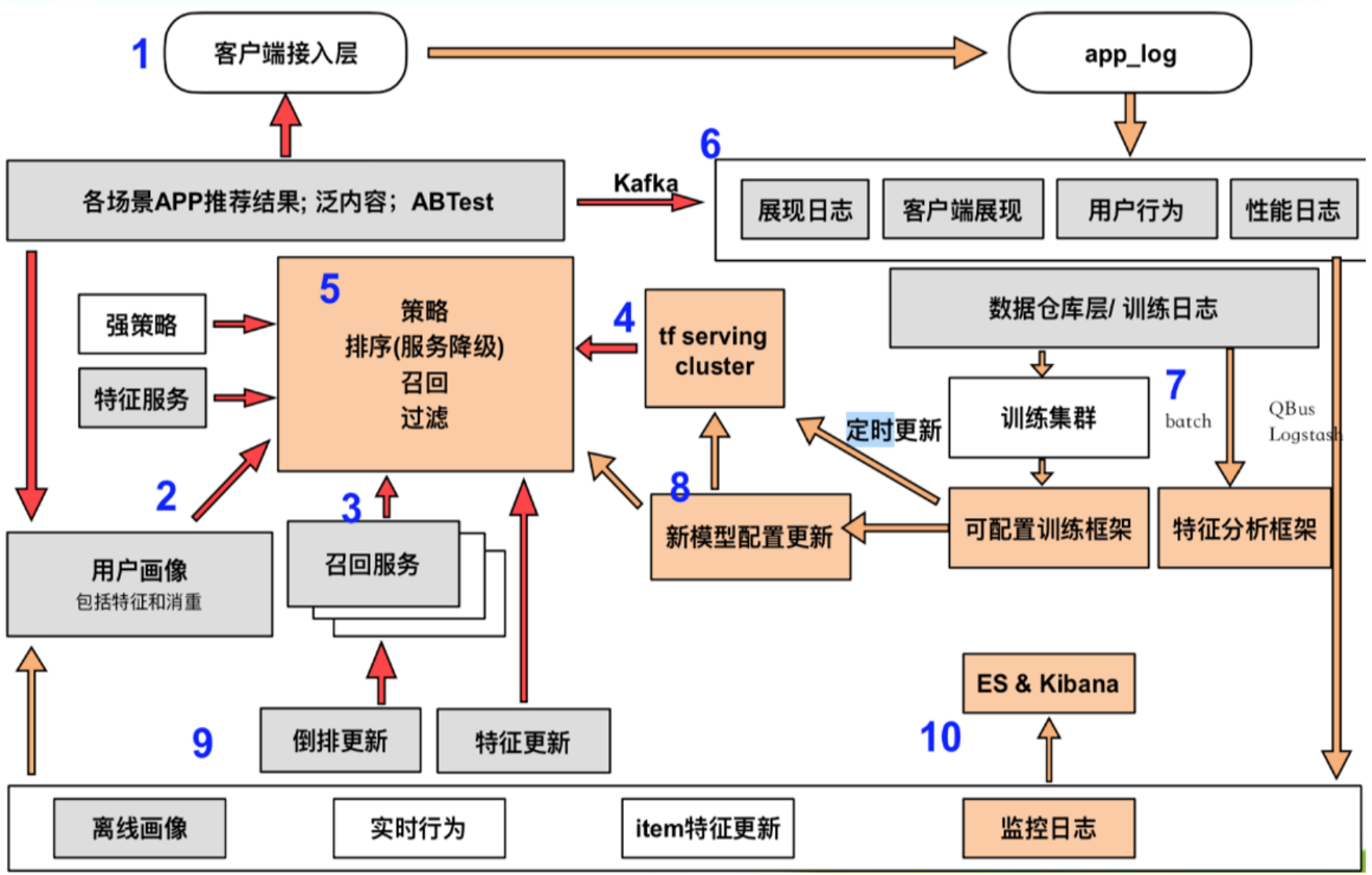
基于用户选择年龄、性别和地域信息推荐购物、旅游、新闻、工具、游戏四类二级类别应用
User 特征：用户的年龄、性别、地域、网络环境、渠道、手机型号品牌，
Item 特征：应用的名称、标签、简介、评论以及历史点击、下载、搜索、安装等信息

Example

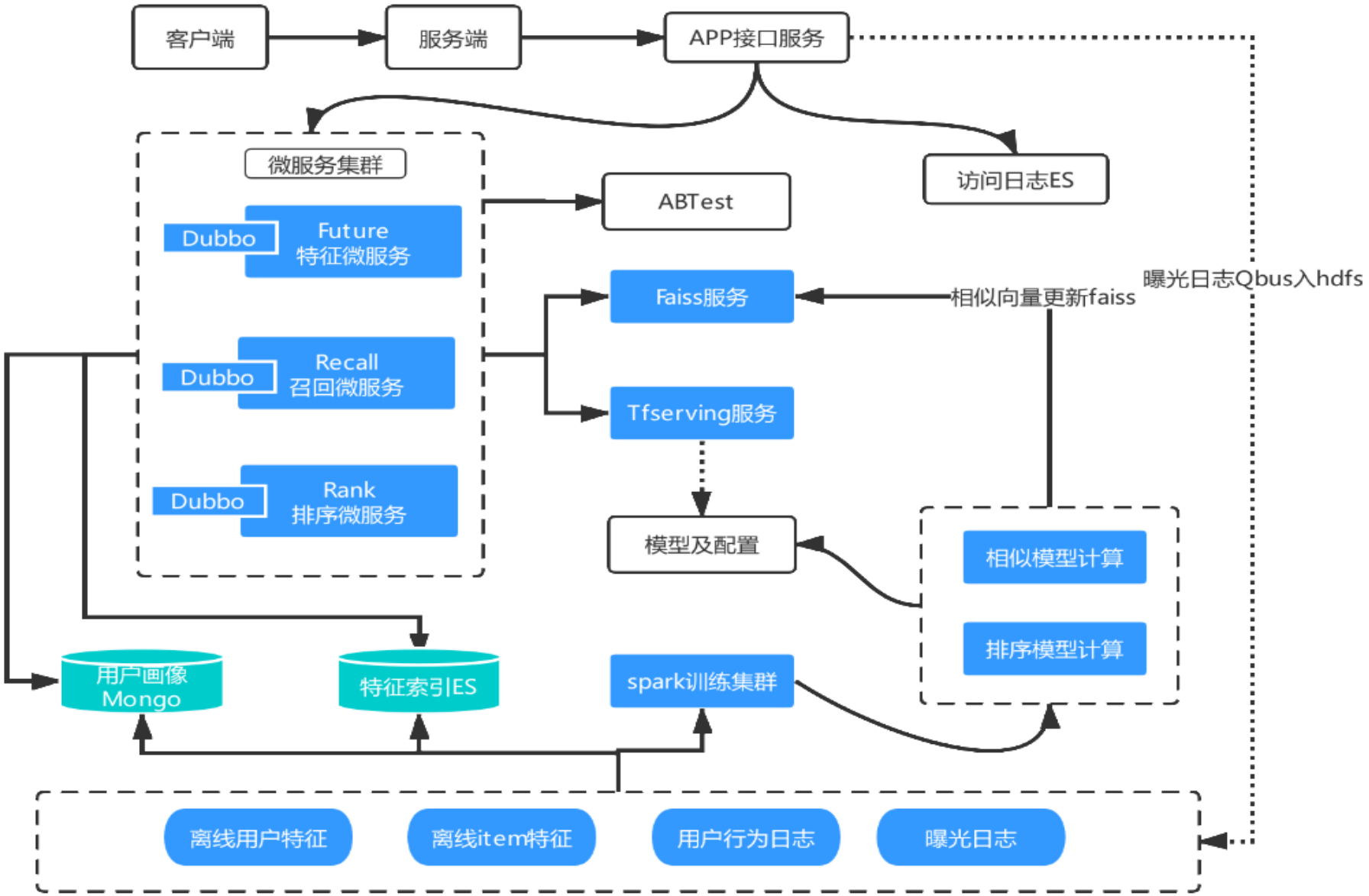


今日热点
弹窗
详情页
安装完成
搜索词
Topic List
开屏
搜索结果
猜你喜欢
下了还会下
……

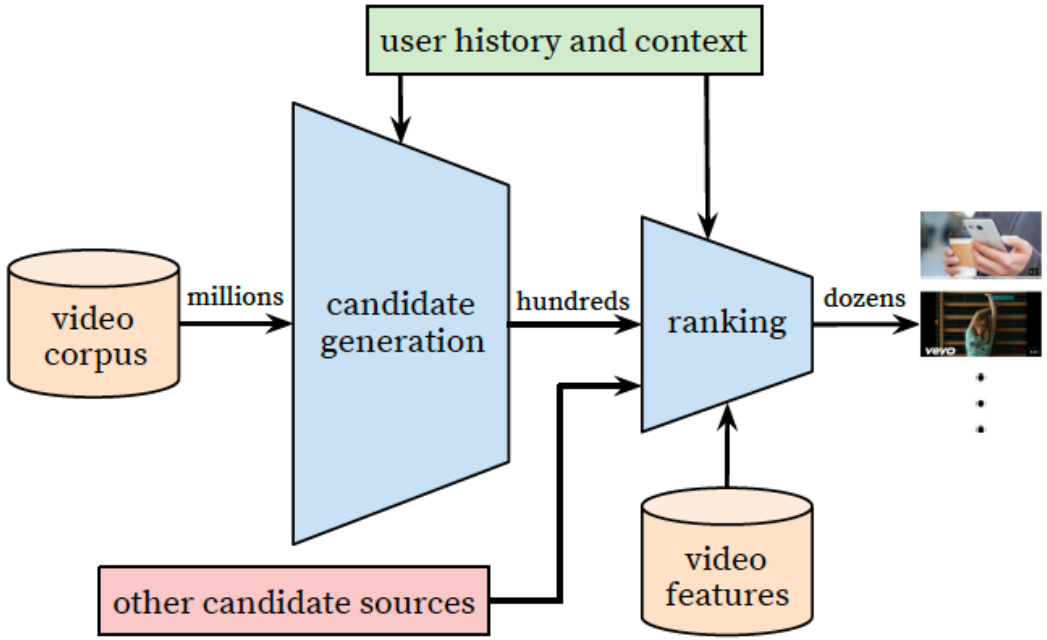
Engine Framework



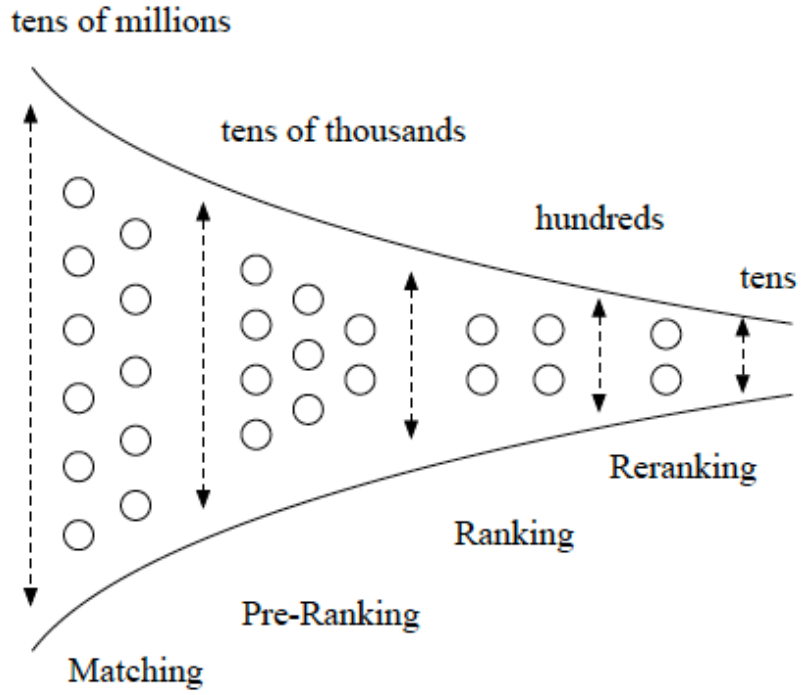
Algorithm Framework



Recommendation System Stage



Google 2016



Ali 2019

召回 → 排序 → 策略

Match - TGI

Target Group Index（目标群体指数）(冷启动)

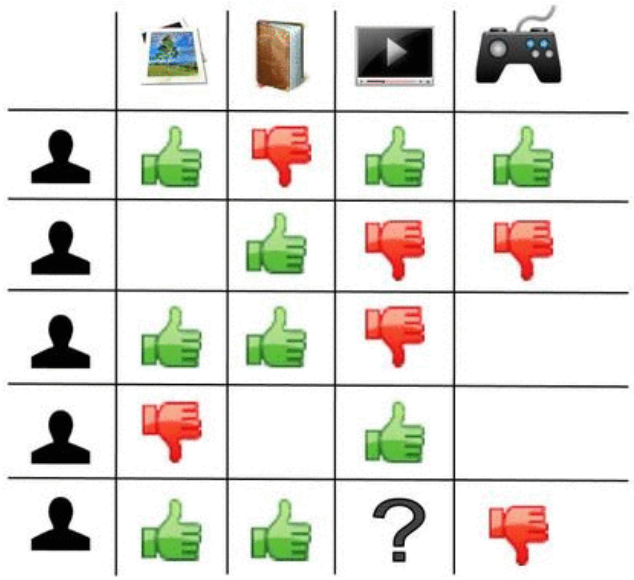
$$TGI=\frac{\text{目标群体中具有某一特征的群体所占比例}}{\text{总体中具有相同特征的群体所占比例}} * \text{标准数}100$$

人数比例	男性	女性
总人数	60%	40%
王者荣耀	90%	10%
美图秀秀	20%	80%

TGI	男性	女性
王者荣耀	150	25
美图秀秀	33.33	200

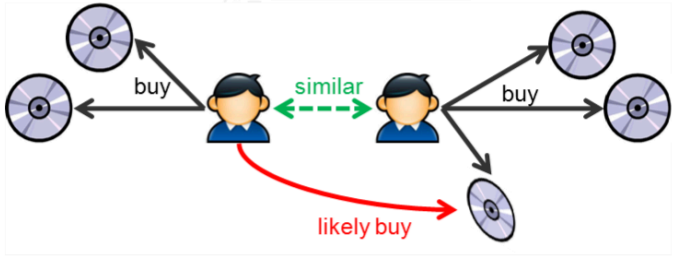
$$[\text{王者荣耀,男性}]\ TGI=\frac{90\%}{60\%} * 100 = 150$$

Match - Collaborative Filtering

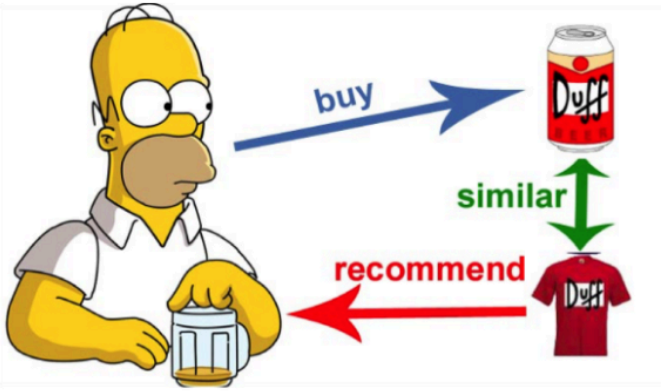


$$\mathrm{Sim}(u_i, u_k) = \frac{\sum_j (r_{ij} - r_i)(r_{kj} - r_k)}{\sqrt{\sum_j (r_{ij} - r_i)^2 \sum_j (r_{kj} - r_k)^2}}$$

$$r_{ij} = \frac{\sum_k \mathrm{Sim}(u_i, u_k) r_{kj}}{\text{number of ratings}}$$



User-based CF

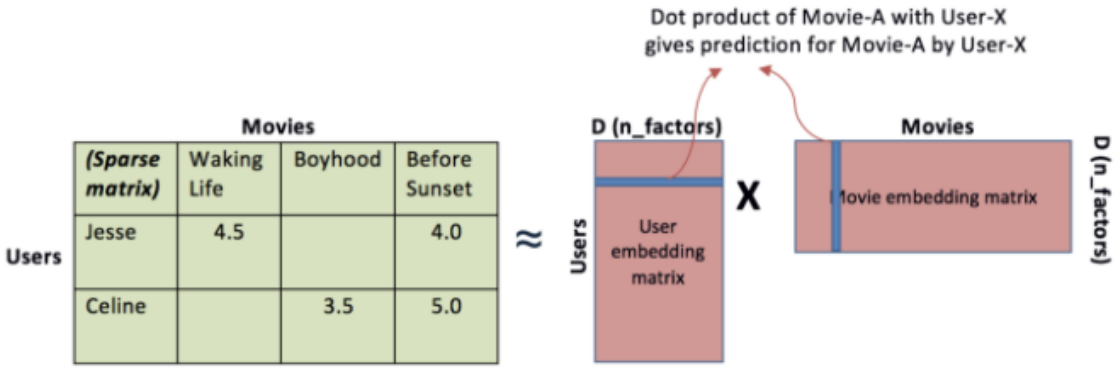


Item-based CF

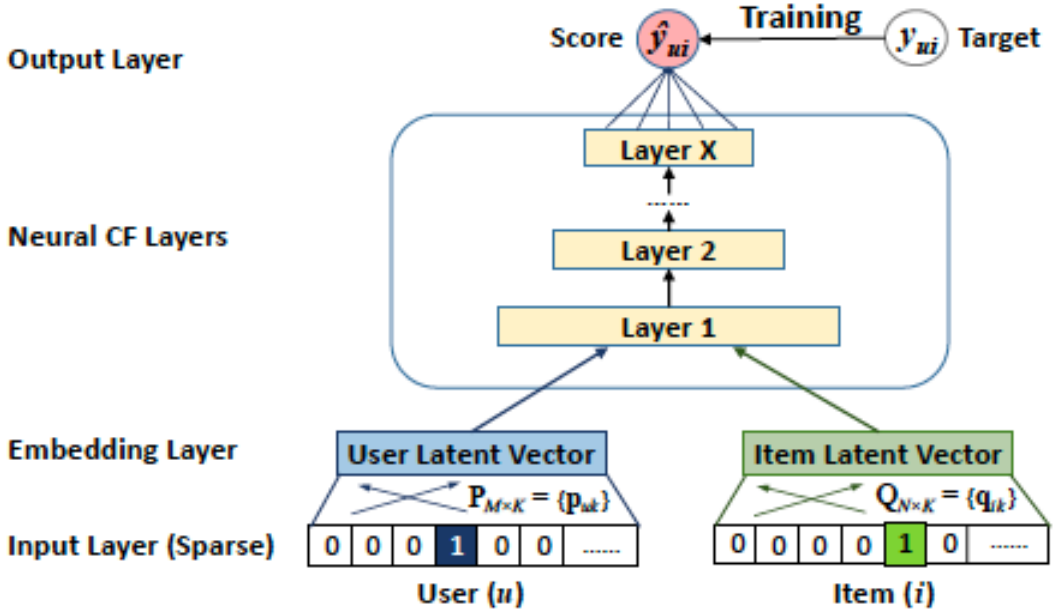
优点：新颖性、相似推荐

缺点：冷启动、稀疏性、扩展性

Match - MF vs NCF

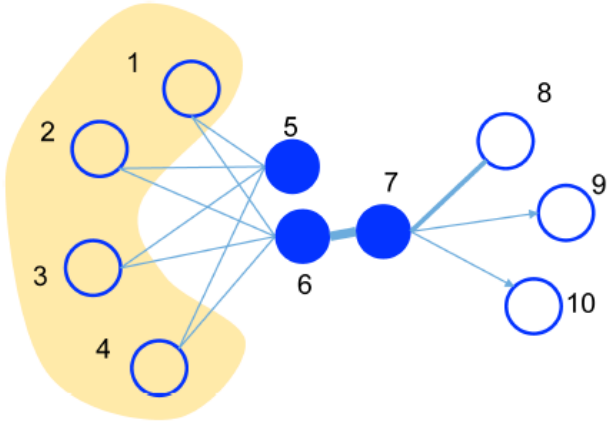


$$\min_{p,q,b_u,b_i} \sum \left(r_{ui} - \left(p_u^T q_i + \mu + b_u + b_i\right)\right)^2$$

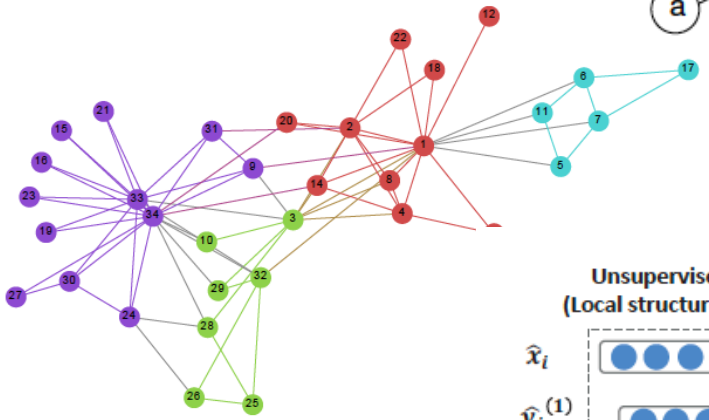


$$\begin{aligned}
\mathbf{z}_1 &= \phi_1\left(\mathbf{p}_u, \mathbf{q}_i\right) = \begin{bmatrix} \mathbf{p}_u \\ \mathbf{q}_i \end{bmatrix} \\
\phi_2\left(\mathbf{z}_1\right) &= a_2\left(\mathbf{W}_2^T \mathbf{z}_1 + \mathbf{b}_2\right) \\
&\cdots\cdots \\
\phi_L\left(\mathbf{z}_{L-1}\right) &= a_L\left(\mathbf{W}_L^T \mathbf{z}_{L-1} + \mathbf{b}_L\right) \\
\hat{y}_{ui} &= \sigma\left(\mathbf{h}^T \phi_L\left(\mathbf{z}_{L-1}\right)\right)
\end{aligned}$$

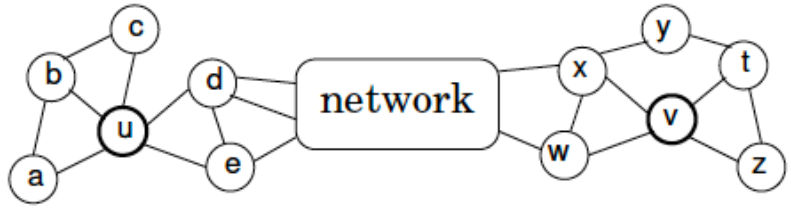
Match - Graph Embedding



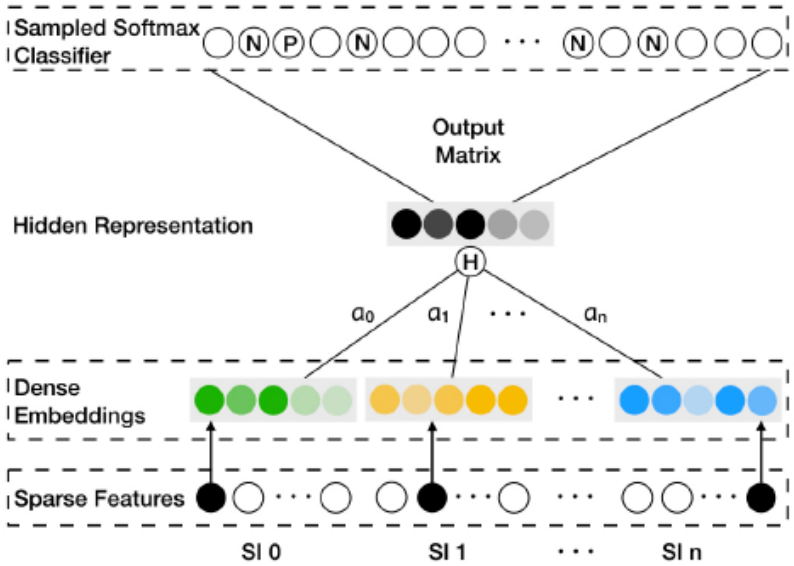
LINE



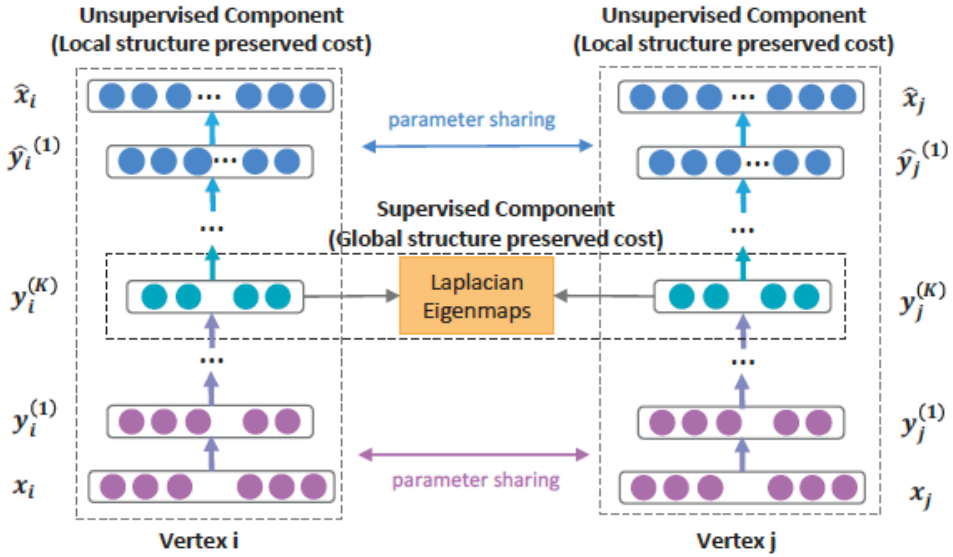
Deepwalk



Struct2vec



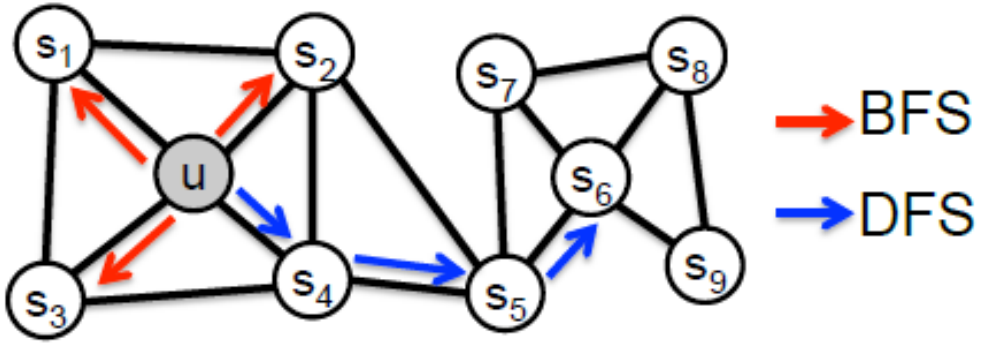
EGES



SDNE

Match - Node2vec

兼顾了深度优先和广度优先的策略



Data：有偏的随机游走策略，将空间数据转化为序列数据

给定节点 v ,访问下一个节点 x 的概率为：

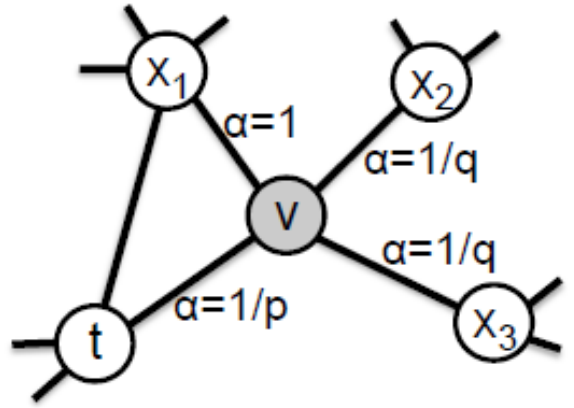
$$p(c_t = x \mid c_{t-1} = v) = \begin{cases} \frac{\pi_{vx}}{Z} & (v,x) \in E \\ 0 & otherwise \end{cases}$$

Z 为归一化常数

node2vec利用两个超参数 p和 q 来控制随机游走的策略，假设随机游走经过边(t,v)到达节点 x 则 $\pi_{vx} = \alpha_{pq}(t,x) \cdot w_{vx}$

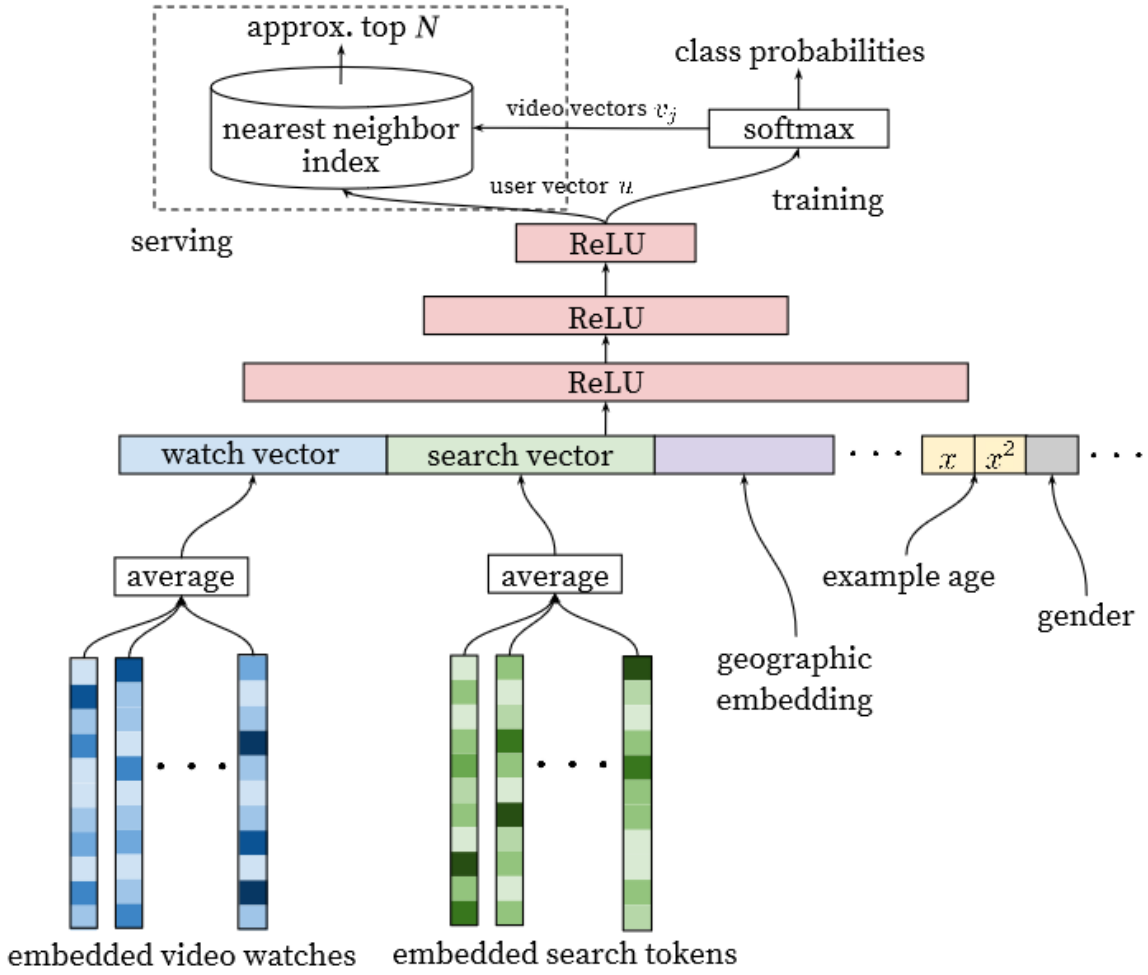
$$\alpha_{pq} = \begin{cases} \frac{1}{p} & if \quad d_{tx} = 0 \\ 1 & if \quad d_{tx} = 1 \\ \frac{1}{q} & if \quad d_{tx} = 2 \end{cases}$$

d_{tx} 为节点 t 到节点 x 之间的最短路径距离

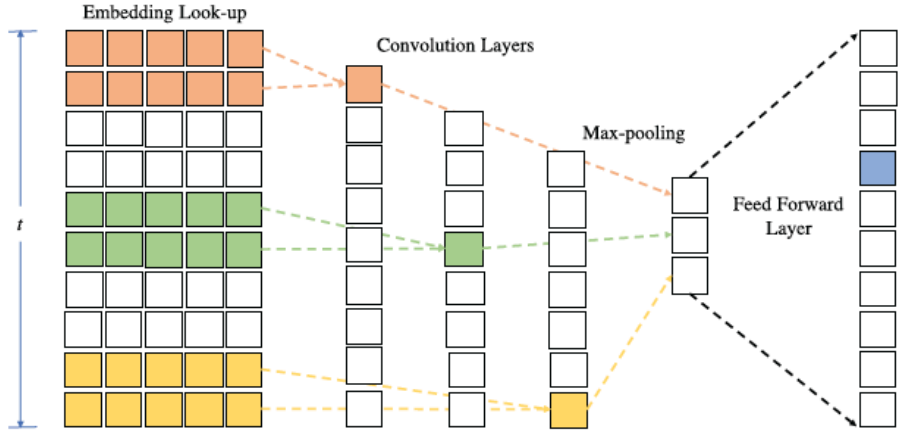


p 控制重复访问刚刚访问过的节点的概率
q 控制偏向BFS或者偏向DFS的概率

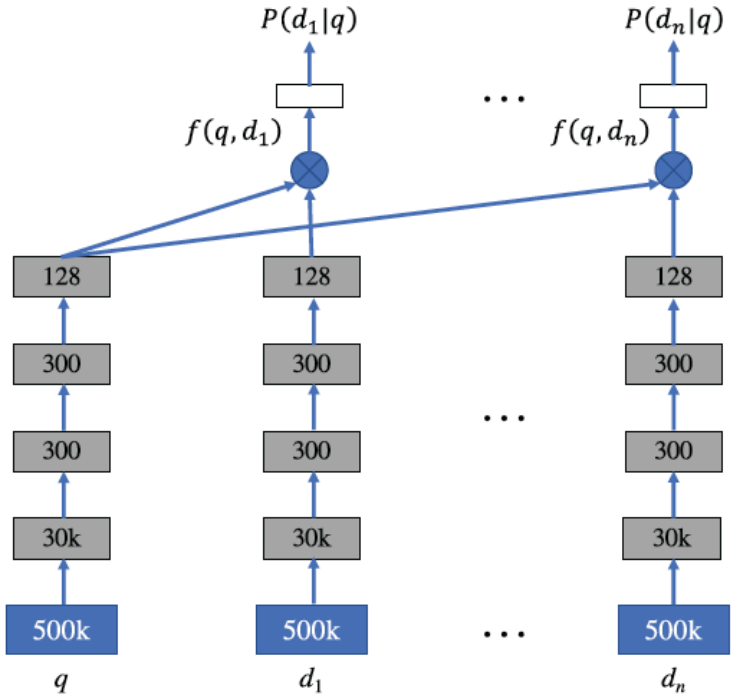
Match - DNN



Google 2016



Caser

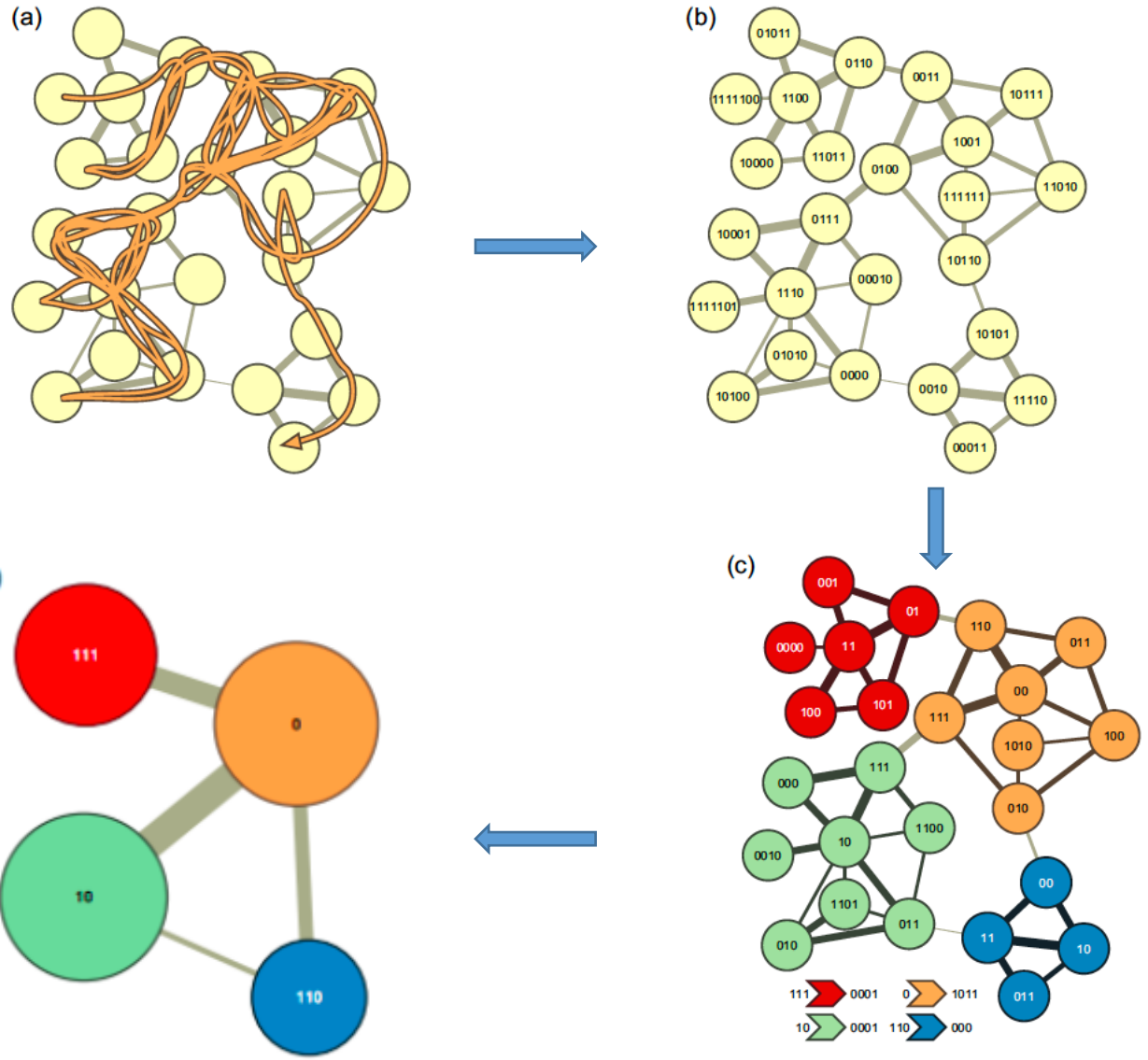


DSSM

Match - Others

- 热门
- 广告
- 聚类（InfoMap）
- 关联规则挖掘
- 序列召回
- 最近邻
- 树模型

…



InfoMap

Rank - FM

$$x = [x_{user}; x_{item}] = [x_1, x_2, \ldots x_n]$$

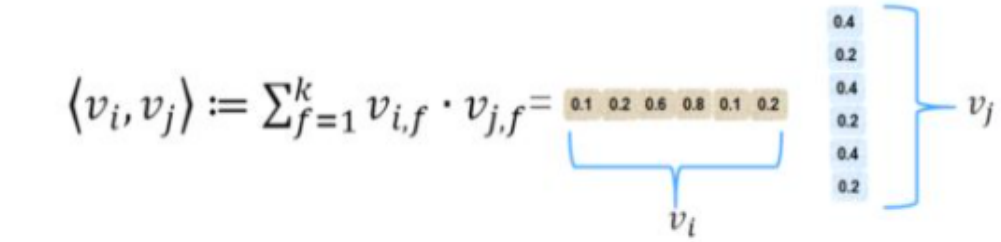
$$y \in [0, 1]$$

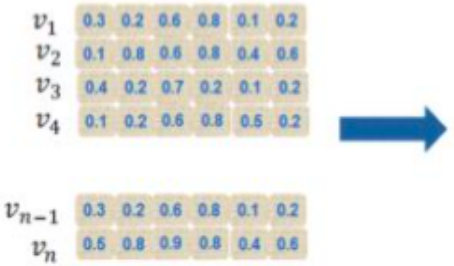
$$Linear : \hat{y}(x) = w_0 + \sum_{i=1}^{n} w_i x_i$$

$$LR : \hat{y}(x) = \frac{1}{1+w_0 \exp(-w^T x)}$$

$$\mathrm{FM} : \hat{y}(x) = w_0 + \sum_{i=1}^{n} w_i x_i + \sum_{i=1}^{n} \sum_{j=i+1}^{n} \langle v_i, v_j \rangle x_i x_j$$

FM模型





$w_{i,j} = \langle v_i, v_j \rangle \neq 0$

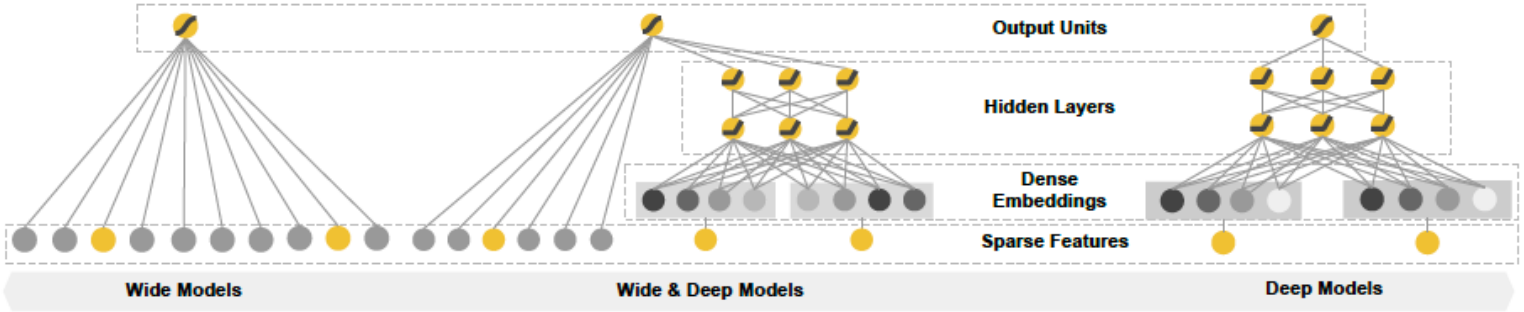
even if 训练数据中$x_i x_j = 0$

only if 在训练数据中存在k使得$x_i x_k \neq 0$

FM模型泛化能力强

Rank - Wide&Deep

$$w_{wide} = \frac{1}{1+w_0 \exp(-w^T x)}$$

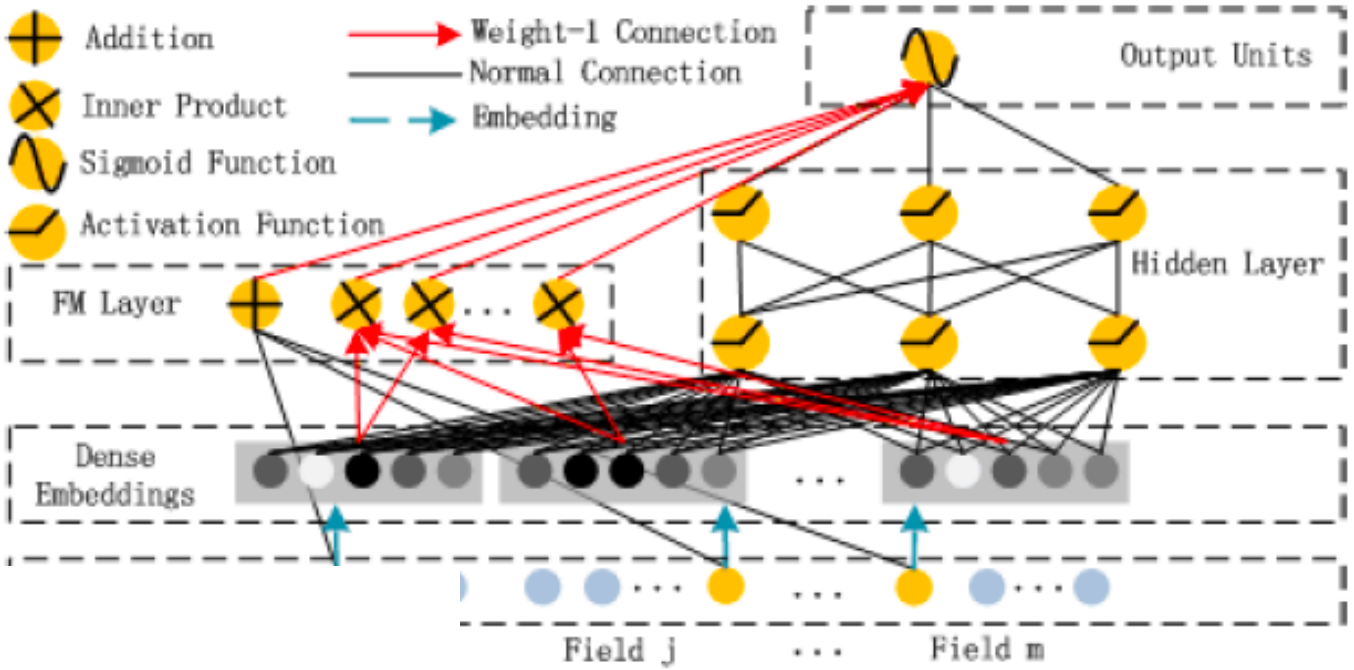


$$\hat{y}(x) = \sigma\left(\mathrm{w}_{wide}^T[\mathrm{x}, \phi(\mathrm{x})] + \mathrm{w}_{\mathrm{deep}}^T a^{(l_f)} + b\right)$$

- Wide: Memorization, 基于人工先验知识，特征交叉，喂入Wide侧，让Wide侧能够记住这些规则
- Deep: Generalization，普通DNN，embedding categorical feature 让DNN学习到这些特征之间的深层交叉，以增强扩展能力
- 人工特征交叉
  样本X中的特征gender=female和特征language=en同时为1
  新的组合(gender=female, language=en)才为1
- Optimizer: Follow- the-regularized-leader (FTRL) Algorithm

Rank - DeepFM

- 自动的二阶特征交叉
- FM和Deep部分共享embedding
- 参数量虽多，效果优于wide&deep



$$\hat{y}(x) = \sigma(y_{FM} + y_{DNN})$$

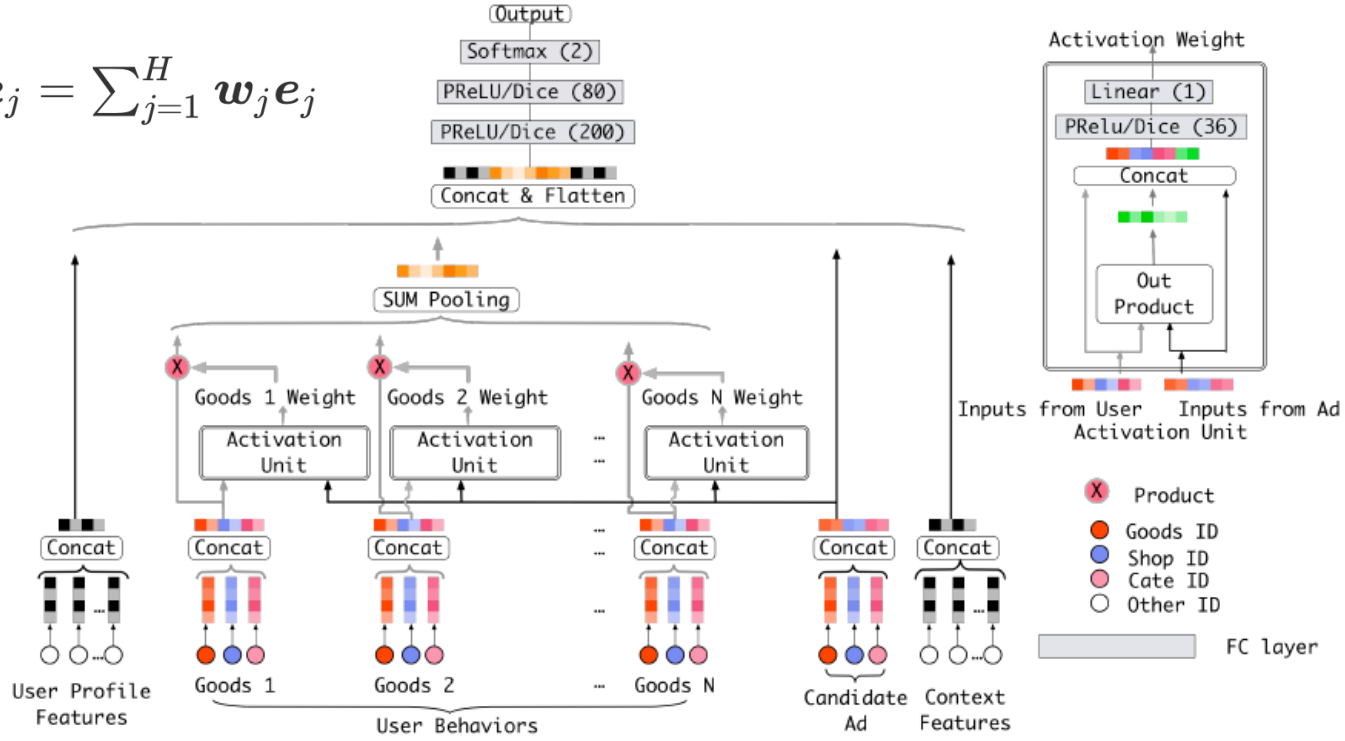
$$y_{FM} = < w, x > + \sum_{j_1=1}^{d} \sum_{j_2=j_1+1}^{d} < V_i, V_j > x_{j_1} . x_{j_2}$$

Rank - DIN

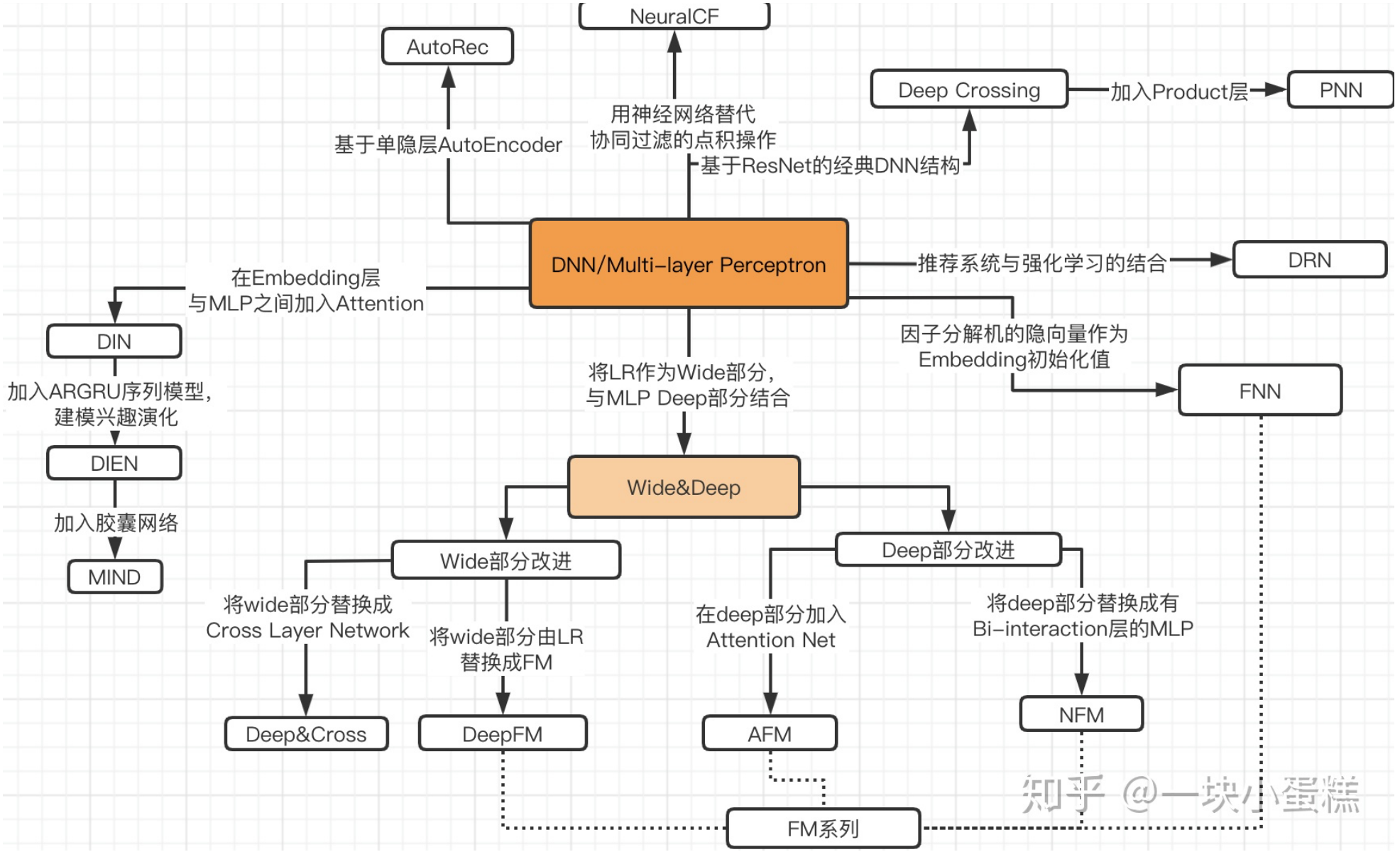
The use of fixed-length vector will be a bottleneck, which brings difficulty for Embedding&MLP methods to capture users diverse interests effectively from rich historical behaviors.

$$\boldsymbol{v}_U(A) = f(\boldsymbol{v}_A, \boldsymbol{e}_1, \boldsymbol{e}_2, \ldots, \boldsymbol{e}_H) = \sum_{j=1}^{H} a(\boldsymbol{e}_j, \boldsymbol{v}_A)\boldsymbol{e}_j = \sum_{j=1}^{H} \boldsymbol{w}_j \boldsymbol{e}_j$$

- 适合电商场景
- 考虑到用户的历史行为特征
- 考虑到用户的兴趣的变化

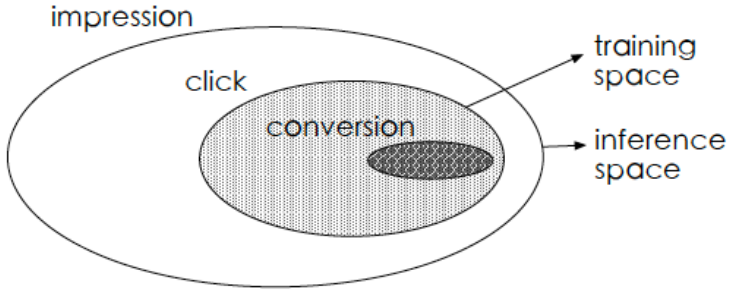


Rank - Others


NeuralCF
AutoRec
Deep Crossing
加入Product层
PNN
用神经网络替代
协同过滤的点积操作
基于单隐层AutoEncoder
基于ResNet的经典DNN结构
DNN/Multi-layer Perceptron
推荐系统与强化学习的结合
DRN
在Embedding层
与MLP之间加入Attention
因子分解机的隐向量作为
Embedding初始化值
DIN
FNN
加入ARGRU序列模型，
建模兴趣演化
将LR作为Wide部分，
与MLP Deep部分结合
DIEN
Wide&Deep
加入胶囊网络
MIND
Wide部分改进
Deep部分改进
将wide部分替换成
Cross Layer Network
将wide部分由LR
替换成FM
在deep部分加入
Attention Net
将deep部分替换成有
Bi-interaction层的MLP
Deep&Cross
DeepFM
AFM
NFM
FM系列
知乎 @一块小蛋糕

Multi Task Learning- ESMM

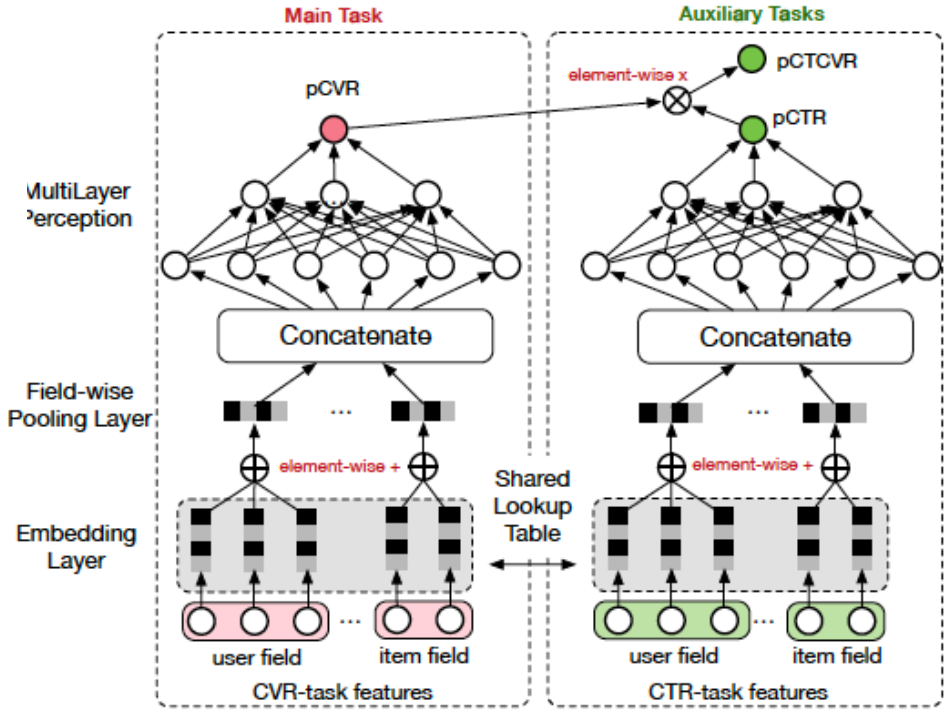
CTR : 从曝光到点击
CVR : 从点击到购买的转化



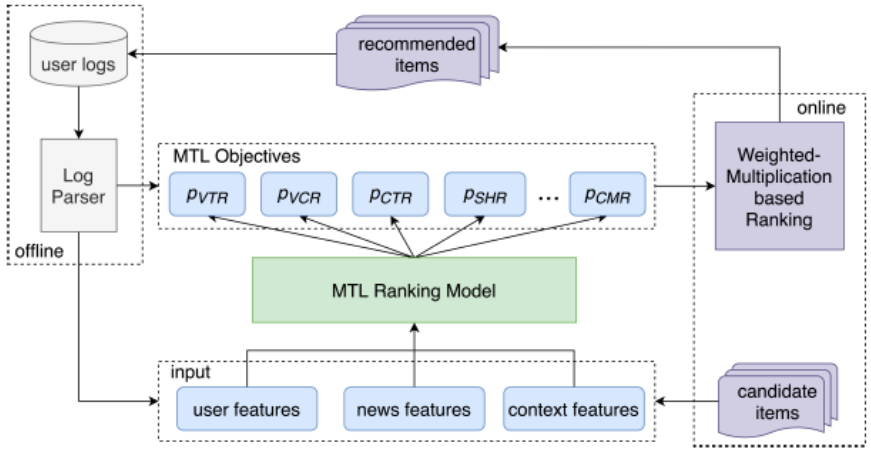
$$\underbrace{p(y=1,z=1\mid x)}_{pCTCVR}=\underbrace{p(y=1\mid x)}_{pCTR}\times\underbrace{p(z=1\mid y=1,x)}_{pCVR}$$

$$p(z=1\mid y=1,x)=\frac{p(y=1,z=1|x)}{p(y=1|x)}$$

$$L\left(\theta_{cvr},\theta_{ctr}\right)=\sum_{i=1}^{N}l\left(y_i,f\left(x_i;\theta_{ctr}\right)\right)$$
$$+\sum_{i=1}^{N}l\left(y_i\&z_i,f\left(x_i;\theta_{ctr}\right)\times f\left(x_i;\theta_{cvr}\right)\right)$$



Multi Task Learning- PLE

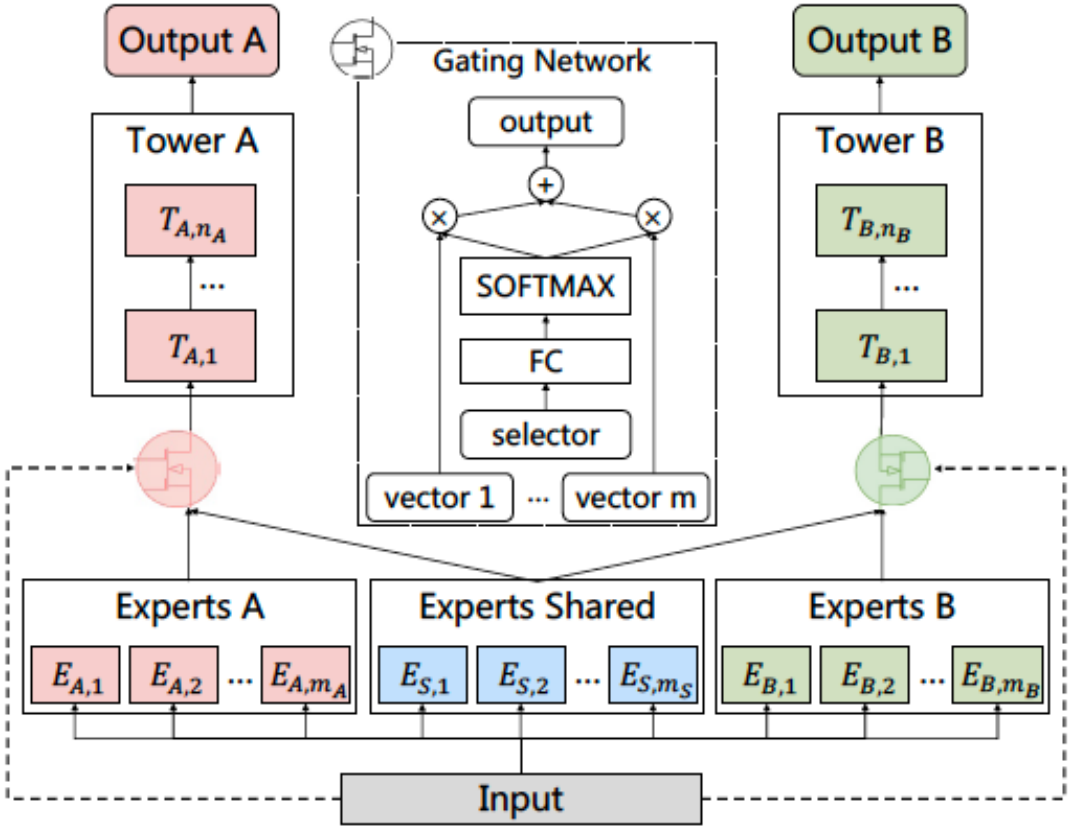


we have observed an interesting seesaw phenomenon that performance of one task is often improved by hurting the performance of some other tasks.

$$S^k(x) = \left[E^T_{(k,1)}, E^T_{(k,2)}, \ldots, E^T_{(k,m_k)}, E^T_{(s,1)}, E^T_{(s,2)}, \ldots, E^T_{(s,m_s)}\right]^T$$

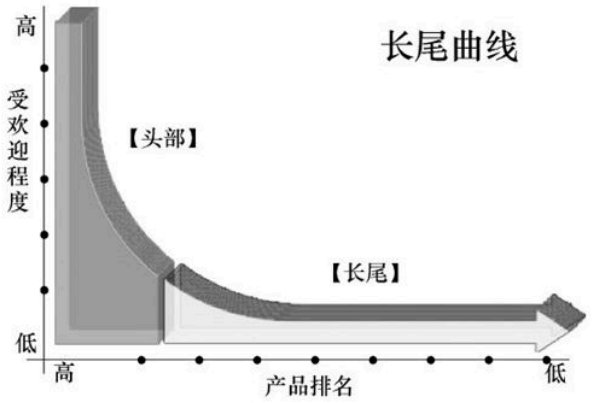
$$w^k(x) = \text{Softmax}\left(W^k_g x\right)$$

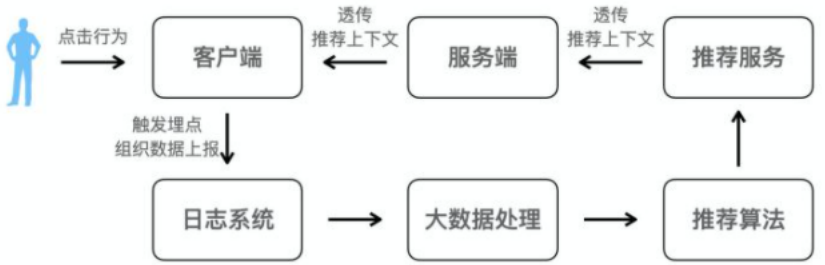
$$g^k(x) = w^k(x) S^k(x)$$

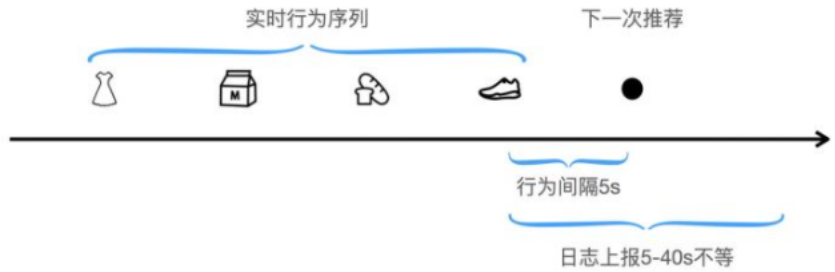


Problem

- 冷启动问题
- 评价指标不一
- Item 长尾问题
- E&E, exploration & exploitation 兴趣探索
- Data：
  - 数据量不足和数据稀疏
  - 数据清洗不干净或者过于干净
  - 数据分布的不一致
- Model：
  - 线下AUC涨，线上CTR跌
  - 特征/数据出现穿越
  - 实时性问题
- 短期被人高估，长期被人低估







谢　谢！