



大数据导论

Introduction to Big Data



大数据推荐系统基础

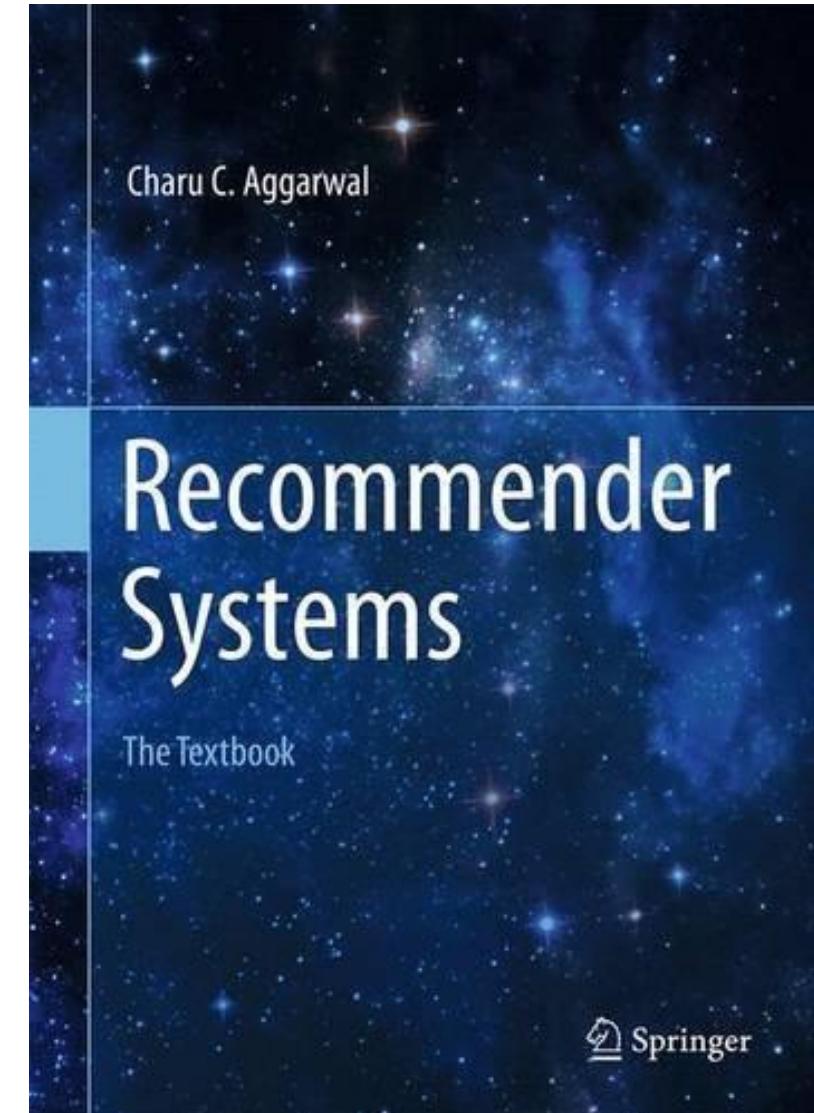
叶允明

计算机科学与技术学院

哈尔滨工业大学（深圳）

目 录

- 推荐系统的基本概念
- 协同过滤算法
 - 基于邻居的协同过滤算法 (Sec. 2.3)
 - 基于矩阵分解的协同过滤算法 (Sec. 3.6)



推荐系统的基本概念

推荐系统的典型应用

● 根据浏览记录推荐商品

The diagram illustrates how a recommendation system uses a user's browsing history to suggest similar products. It shows two examples of product cards with yellow outlines, followed by arrows pointing to a '为你推荐' (Recommended for You) section.

Top Example: A user has viewed an Apple iPhone 8 Plus (64GB, 5.5 inches). The price is ￥5699.00. The card includes a video thumbnail, a rating of 98%好评, and a link to '查看同款拍拍二手'.

Bottom Example: A user has viewed an Honor 9i (4GB+64GB, 幻夜黑, 移动联通双卡双待). The price is ￥1399.00. The card includes a photo of a woman holding the phone, a rating of 99%好评, and a link to '查看同款拍拍二手'.

Recommendations:

- For iPhone 8 Plus:** Black Shark Gaming Phone (8GB+128GB), Apple iPhone X/iPhone 8/8 Plus, and Apple iPhone 7 Plus.
- For Honor 9i:** LeTV Z5 (6GB+128GB), Honor 9 Youth Edition (全网通), and Huawei畅享8.

排行榜 (Ranking): The same three phones listed above are also shown in a separate ranking section.

推荐系统的典型应用

● 根据听歌历史推荐歌单

The image shows a mobile application interface for a music service. At the top, there's a navigation bar with tabs: '乐库' (Library), '推荐' (Recommendation), '趴间' (Podcast), and '看点' (Highlight). Below the navigation bar is a promotional banner for a new album 'WATCH ME' by 陈慧琳 (Celine Tam) with the text '首发' (First Release). A yellow arrow points from the circled '每日30首' (30 songs daily) icon in the main content area down to the recommended song list.

推荐 (Recommendation) tab is selected.

每日30首 (30 songs daily) icon is circled in red.

必听歌单 (Must-Hear Playlists) section:

- [舒缓] 思念是一种病 (161.4万 plays)
- [林忆莲] 每个人心中都有... (189.7万 plays)
- [经典] 国语金曲【绝对好...】 (648.9万 plays)
- [王菲] 红豆 (425.2万 plays)
- [蔡健雅] 带我走 (6410 plays)
- [周杰伦] 明月几时有 (81823 plays)

底部导航栏:

- 发现 (Discover) - 红色图标
- 我的音乐 (My Music)
- 动态 (Activity)
- 个人中心 (Personal Center)

右侧歌单推荐:

- 根据你的品味,为你每天推荐30首
- ▶ 全部播放 30首
- 歌单封面: Love Me Like You Do - Ellie Goulding (MV)
- 歌单封面: Radioactive - Imagine Dragons (MV)
- 歌单封面: 遇见 - 孙燕姿 (MV)
- 歌单封面: Cold Water - Major Lazer; Justin Bieber, ... (HQ)
- 歌单封面: Me and My Broken Heart - Rixton (HQ)
- 歌单封面: Love Me Like You Do - Ellie Goulding (MV)

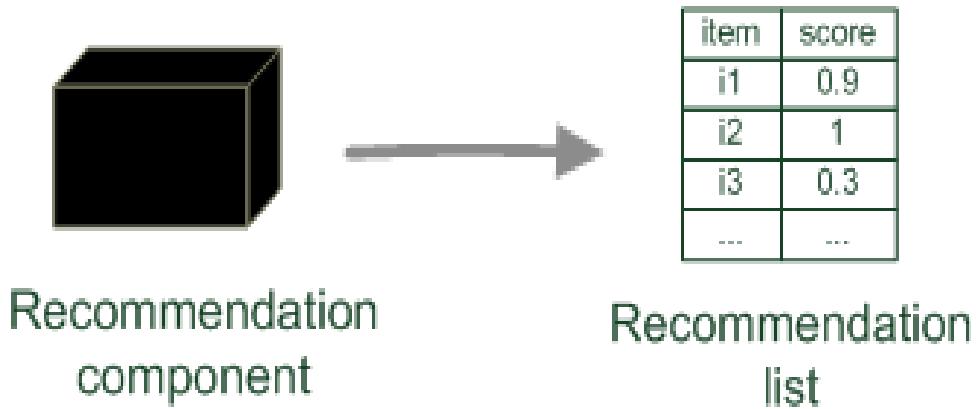
推荐系统的典型应用

● 根据浏览历史推荐新闻

The screenshot shows the Today's Headlines (今日头条) mobile application interface. On the left, there is a sidebar with various categories: 推荐 (Recommendation), 阳光宽频 (Broadband), 热点 (Hot Topics), 图片 (Images), 科技 (Technology), 娱乐 (Entertainment), 游戏 (Games), 体育 (Sports), 汽车 (Automobiles), 财经 (Finance), 搞笑 (Funny), and 更多 (More). A red button labeled '推荐' (Recommendation) is highlighted. The main content area displays a large image of a harbor with many fishing boats under a cloudy sky, with the caption '浙江舟山：千艘渔船归港备战台风“安比”'. Below this, a news item is highlighted with a yellow border: '习近平：欢迎塞内加尔成为第一个同中国签署“一带一路”合作文件的西非国家' (President Xi: Welcome Senegal as the first African country to sign the 'Belt and Road' cooperation document). This item is from 国际 (International) on 人民网 (People's Network) with 26 comments. To the right of the sidebar is a vertical navigation bar with categories: 要闻 (Headline News), 社会 (Society), 娱乐 (Entertainment), 体育 (Sports), 军事 (Military), and 明星 (Star). A blue banner at the top of the main content area says '为您推荐了10篇文章' (Recommended 10 articles for you). Below the banner, several news cards are listed:

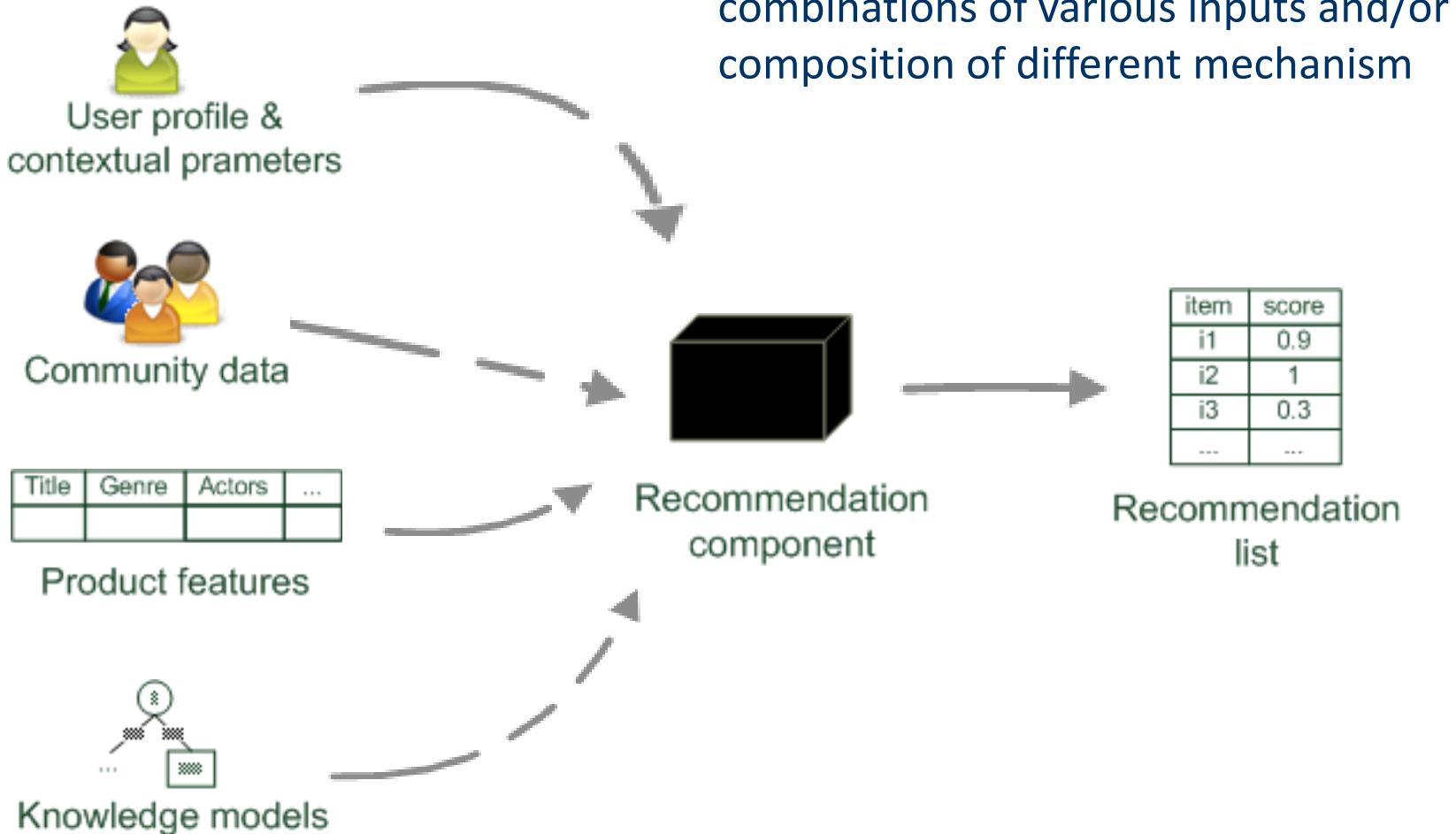
- 习近平：欢迎塞内加尔成为第一个同中国签署“一带一路”合作文件的西非国家 (International, 人民网, 26评论, 刚刚)
- 习近平在南非媒体发表署名文章 (International, 新华网, 1评论, 刚刚)
- 中国陆军首度军长大考，释放出什么信号？ (Military, 上观新闻, 20评论, 刚刚)
- 事业单位合并后，有职称的人员应该如何安置？ (Social, 悟空问答, 刚刚)
- 面对美国颠倒黑白，华春莹的这些回应太精彩！ (International, 海外网, 25评论, 刚刚)

A recommender system is a subclass of information filtering system that seeks to predict the "rating" or "preference" a user would give to an item. —Wiki

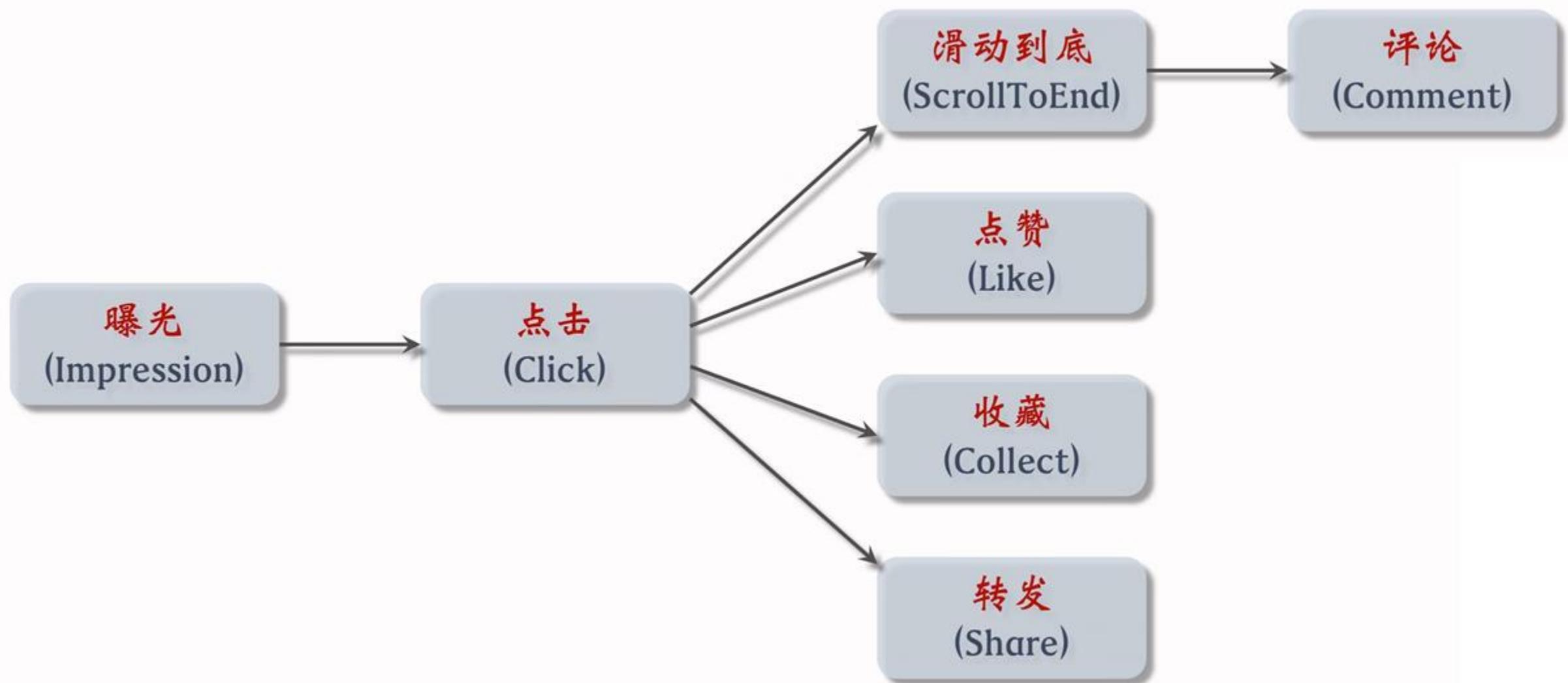


推荐系统：通过相关性估计减少信息过载！

推荐系统的多源数据



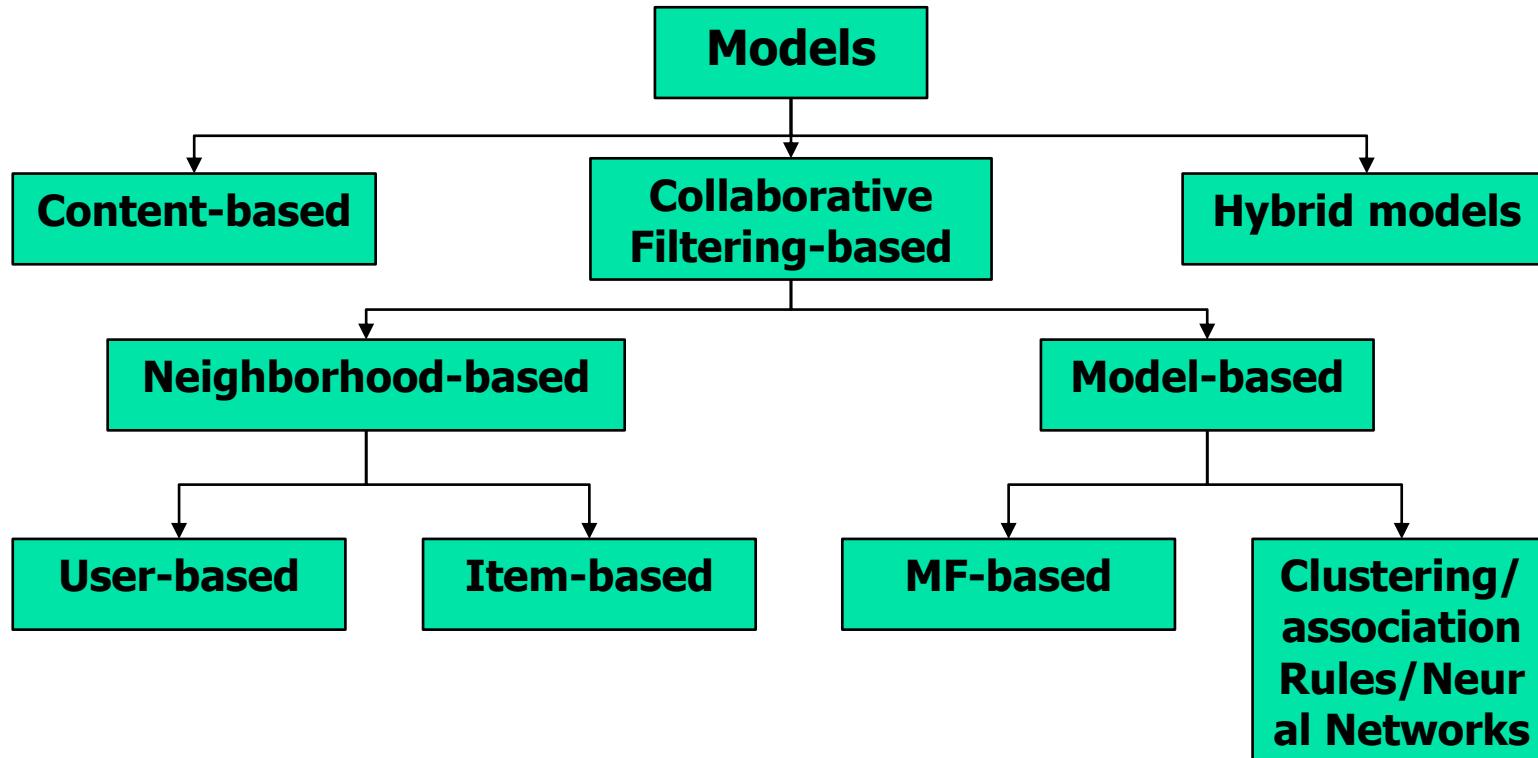
常见的推荐系统转化过程



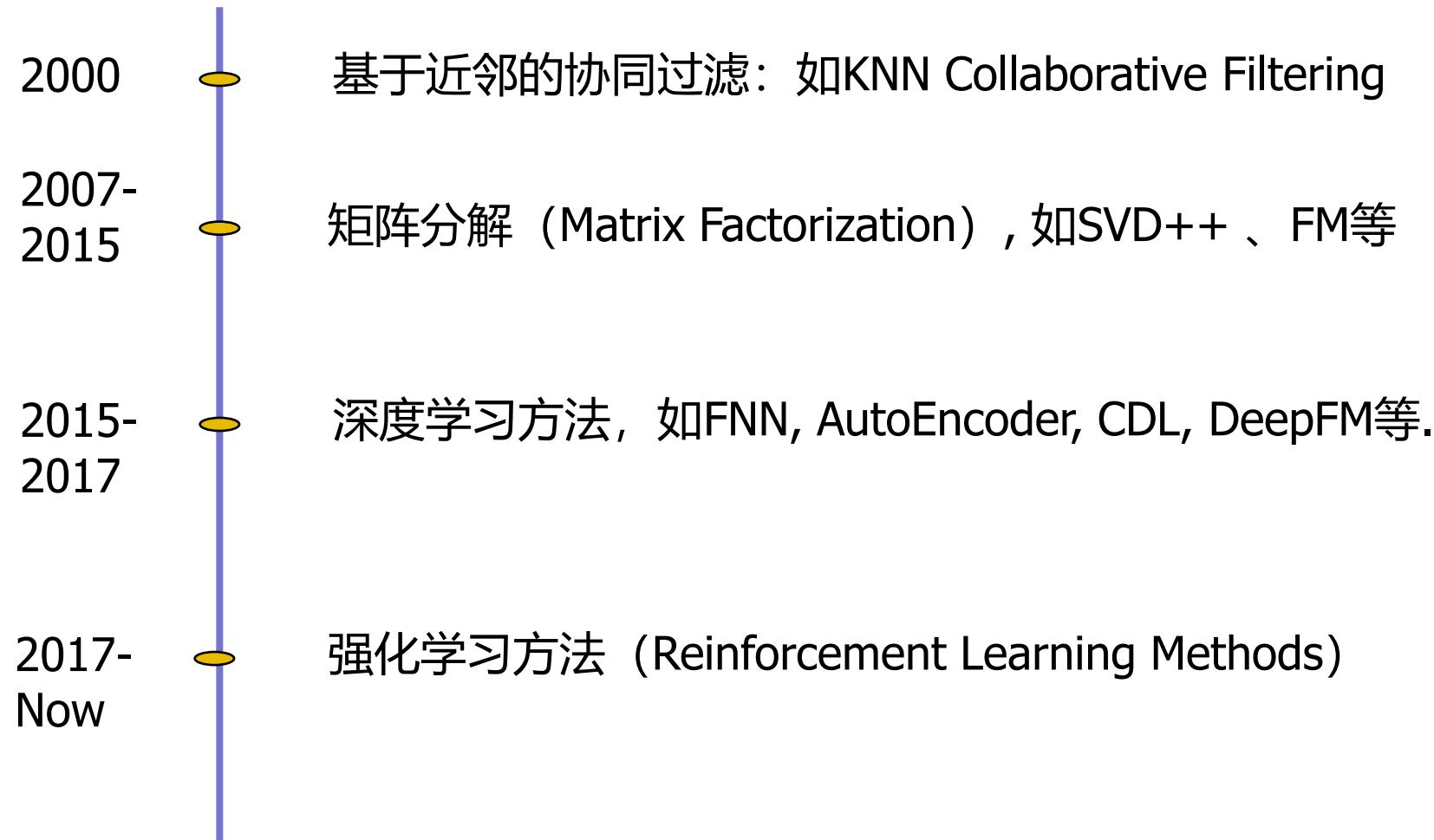
常见的推荐系统评价指标

- 点击率 = 点击次数 / 曝光次数
- 点赞率 = 点赞次数 / 点击次数
- 收藏率 = 收藏次数 / 点击次数
- 转发率 = 转发次数 / 点击次数
- 用户规模：
 - 日活用户数 (DAU) 、月活用户数 (MAU)

传统推荐模型的主要类别



推荐系统的发展历程



基于邻居的协同过滤

什么是协同过滤（Collaborative Filtering, CF）？

	Book1	Book2	Book3	Book4	Book5	Book6
User1							
User2							
User3							
User4							
User5							
User6	?	?		?	?	?	?

基于邻居的协同过滤算法

- 具有广泛应用领域的有效推荐方法

- 在网络书店、电影推荐等领域广泛应用
 - 简单有效、且具有良好的可解释性

- 基本假设与思想：

- 用户对物品进行显式或隐式的评分 (rating)
 - 假设：用户未来的偏好与过去的偏好相似
 - 基于“群体智慧 (wisdom of the crowd) ”来推荐商品

- 可分为两种算法：

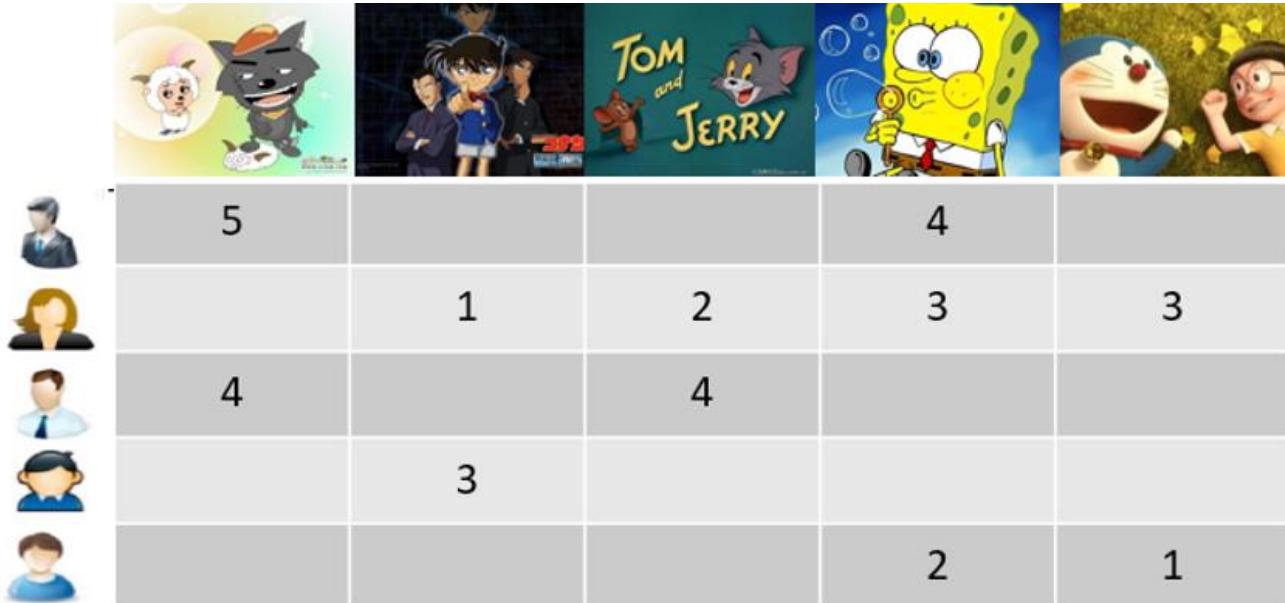
- 基于用户的协同过滤 (User-based collaborative filtering)
 - 基于物品的协同过滤 (Item-based collaborative filtering)

评分矩阵 (Rating matrix)

User	Item	Rating
1	1	5
1	4	4
...
u	j	r
...
...



评分矩阵: $R = [r_{uj}]$



基于用户的协同过滤算法

Alice	5	3	4	4	?
User1	3	1	2	3	3
User2	4	3	4	3	5
User3	3	3	1	5	4
User4	1	5	5	2	1

基于用户的协同过滤算法

■ 需解决三个问题：

- 如何度量相似性？
- 需要选多少个“邻居”？
- 如何基于邻居的评分做出预测？

Alice	5	3	4	4	?
User1	3	1	2	3	3
User2	4	3	4	3	5
User3	3	3	1	5	4
User4	1	5	5	2	1

用户相似性度量方法

■ 常用的相似性度量方法：皮尔逊相关系数（Pearson correlation）

u, v : 用户

r_{uk} : 用户 u 对物品 k 的评分

I_u : 被用户 u 评分过的物品对应的索引号集合

μ_u : 用户 u 的平均评分 (基于其历史评分计算)

- 第一步是计算 μ_u :

$$\mu_u = \frac{\sum_{k \in I_u} r_{uk}}{|I_u|} \quad \forall u \in \{1 \dots m\}$$

- 第二步是计算用户 u 和 v 的皮尔逊相关系数:

$$\text{Sim}(u, v) = \text{Pearson}(u, v) = \frac{\sum_{k \in I_u \cap I_v} (r_{uk} - \mu_u) \cdot (r_{vk} - \mu_v)}{\sqrt{\sum_{k \in I_u \cap I_v} (r_{uk} - \mu_u)^2} \cdot \sqrt{\sum_{k \in I_u \cap I_v} (r_{vk} - \mu_v)^2}}$$

用户相似性度量示例

Alice	5	3	4	4	?
User1	3	1	2	3	3
User2	4	3	4	3	5
User3	3	3	1	5	4
User4	1	5	5	2	1

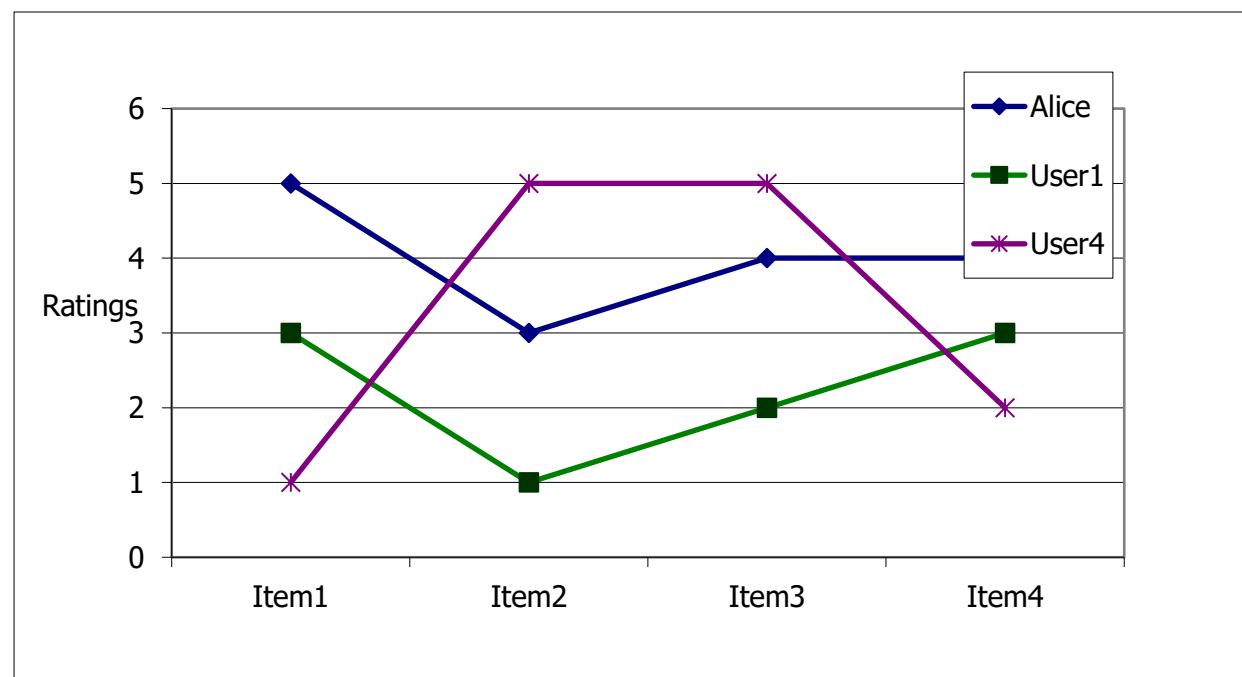


$\text{sim} = 0.85$

$\text{sim} = 0.00$

$\text{sim} = 0.70$

$\text{sim} = -0.79$



预测过程

- 第一步：选出 k 个具有最高皮尔逊相关系数的用户作为“邻居”集合
- 第二步：每个邻居的评分值需进行去均值化（为什么？）：

$$s_{uj} = r_{uj} - \mu_u \quad \forall u \in \{1 \dots m\}$$

- 第三步：计算预测值

$$\hat{r}_{uj} = \mu_u + \frac{\sum_{v \in P_u(j)} \text{Sim}(u, v) \cdot s_{vj}}{\sum_{v \in P_u(j)} |\text{Sim}(u, v)|} = \mu_u + \frac{\sum_{v \in P_u(j)} \text{Sim}(u, v) \cdot (r_{vj} - \mu_v)}{\sum_{v \in P_u(j)} |\text{Sim}(u, v)|}$$

➤ $P_u(j)$ 表示与用户 u 最相似的、且在物品 j 上有过评分的 k 个用户（邻居）集合

预测过程示例

$$\hat{r}_{uj} = \mu_u + \frac{\sum_{v \in P_u(j)} \text{Sim}(u, v) \cdot s_{vj}}{\sum_{v \in P_u(j)} |\text{Sim}(u, v)|} = \mu_u + \frac{\sum_{v \in P_u(j)} \text{Sim}(u, v) \cdot (r_{vj} - \mu_v)}{\sum_{v \in P_u(j)} |\text{Sim}(u, v)|}$$

Alice	5	3	4	4	?	
User1	3	1	2	3	3	sim = 0.85
User2	4	3	4	3	5	sim = 0.00
User3	3	3	1	5	4	sim = 0.70
User4	1	5	5	2	1	sim = -0.79

$$pred(Alice, \text{Cartoon5}) = 4 + \frac{0.85 * (3 - 2.4) + 0.7 * (4 - 3.2)}{0.85 + 0.7} = 4.69$$

基于物品的协同过滤算法

- 利用物品之间的相似性来做预测

Alice	5	3	4	4	?
User1	3	1	2	3	3
User2	4	3	4	3	5
User3	3	3	1	5	4
User4	1	5	5	2	1

物品相似性度量方法

- 物品 i 和 j 的相似性基于在这两个物品上都有评分的用户集合

- U_i : 在物品 i 上有评分的用户集合

$$\text{AdjustedCosine}(i, j) = \frac{\sum_{u \in U_i \cap U_j} s_{ui} \cdot s_{uj}}{\sqrt{\sum_{u \in U_i \cap U_j} s_{ui}^2} \cdot \sqrt{\sum_{u \in U_i \cap U_j} s_{uj}^2}}$$

➤ $s_{ui} = r_{ui} - \mu_u$

预测过程

- 选出 k 个具有最高相关系数的物品作为物品 t 的“邻居”集合
- 计算：

$$\hat{r}_{ut} = \frac{\sum_{j \in Q_t(u)} \text{AdjustedCosine}(j, t) \cdot r_{uj}}{\sum_{j \in Q_t(u)} |\text{AdjustedCosine}(j, t)|}$$

- $Q_t(u)$ 是与商品 t 最相关的 top- k 个商品（且用户 u 有过评分）

r_{uj} 无需再做均值化，为什么？

两种方法的特点对比

■ 基于物品的推荐算法

- 物品的相似性更稳定
- 推荐的物品更相关, 具有更好的准确度
- 可能会推荐一些很显然、很普通的物品 (如推荐牛奶)

■ 基于用户的推荐算法

- 稳定性不如前者
- 推荐的多样性更好, 可能会推荐一些有新意、眼前一亮的商品

基于邻居的推荐模型：优缺点分析

- 优点

- 简单，容易实现和调优
- 可解释性好
- 对增量数据具有比较好的稳定性

缺点

- 推荐阶段需要更多的计算时间（类似于KNN）
- 容易受数据稀疏性问题（data sparsity problems）的影响

数据稀疏性问题

- **冷启动问题 (Cold start problem)**

- 新建设系统中缺少数据积累的问题
 - 新物品、新用户该如何推荐？

- **简单处理方法**

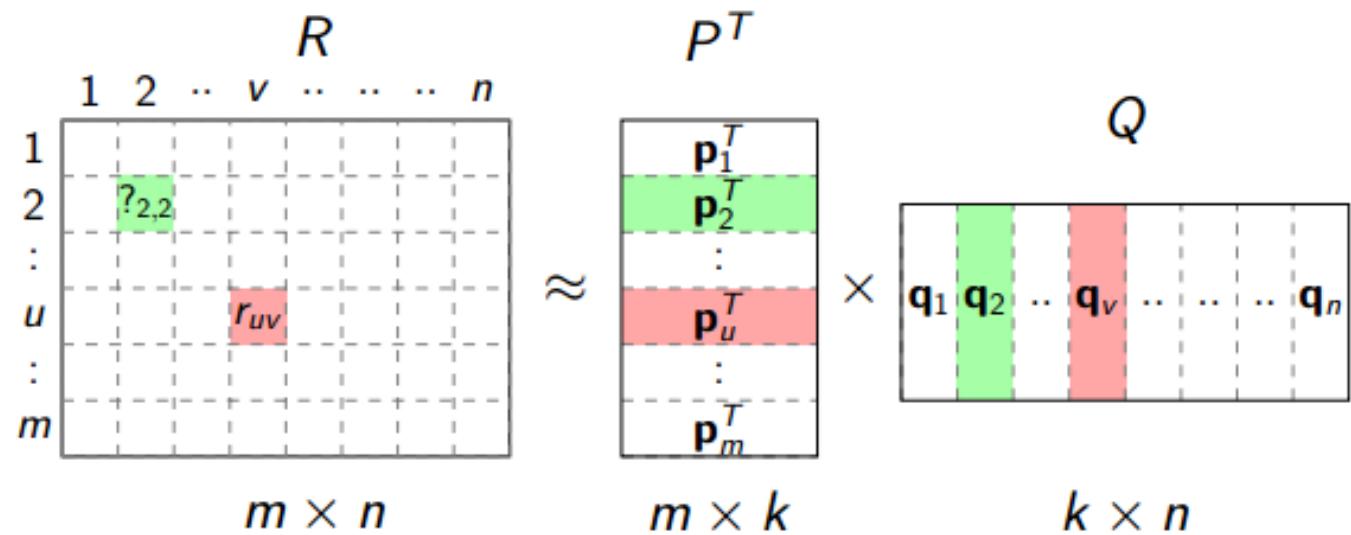
- 请求或要求用户对一些物品评分
 - 在推荐系统建设的初始阶段采用基于内容的推荐模型
 - 更好的方法.....

基于矩阵分解的协同过滤

基本思想

- **潜在因子模型 (Latent factor models)**
 - 商品和用户的评分都受到一些共同的“潜在因子”的影响
 - 例如：用户-电影的评分矩阵的潜在因子：
 - ✓ 电影可以分为：喜剧片、战争片、烂漫爱情片等
 - ✓ 用户对电影的偏好也可以上面的类别因子来表示
 - ✓ 则以上分类就是电影和用户偏好的共同潜在因子
- **主要方法：基于矩阵分解挖掘潜在因子**
 - 矩阵分解推荐模型曾获得巨大成功：Netflix Prize and KDD Cup 2011

矩阵分解



- k : 隐藏因子的子空间维度
- $r_{u,v} = \mathbf{p}_u^T \mathbf{q}_v$
- $?_{2,2} = \mathbf{p}_2^T \mathbf{q}_2$

基于矩阵分解的推荐模型

A	5	3	5.49	1	
B	4	3	4.84	1	
C	1	1	5.19	5	
D	1	0.70	4	4	
E	1.59	1	5	4	

假设有两个
潜在因子

p_A	2.38	0.40
p_B	2.04	0.41
p_C	0.32	2.19
p_D	0.27	1.72
p_E	0.62	1.78

q_1	1.99	0.21
q_2	1.31	0.20
q_3	1.96	2.08
q_4	0.03	2.27

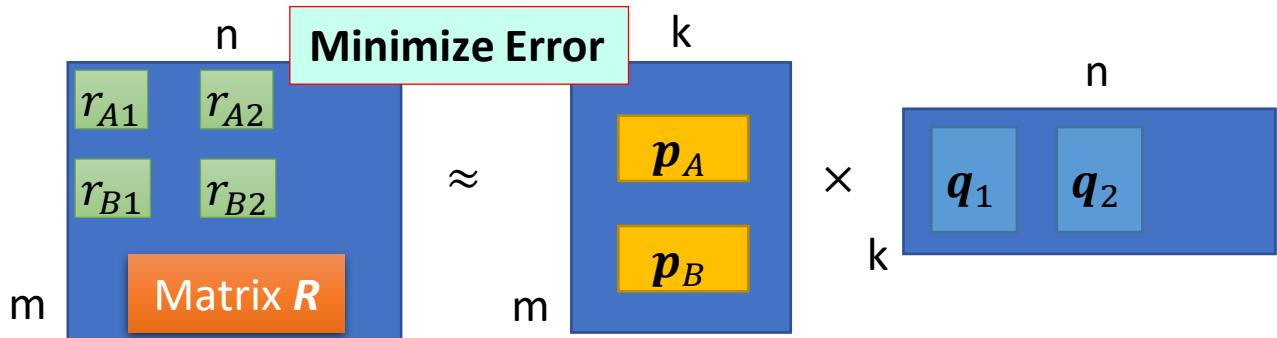
基于矩阵分解的推荐模型

No. of User = m

A	5	3	0	1
B	4	3	0	1
C	1	Matrix R		5
D	1	0	4	4
E	0	1	5	4

No. of Cartoon = n

No. of latent factor = k



$$\begin{aligned} L &= \frac{1}{2} \|R - P^T Q\|^2 \\ &= \frac{1}{2} \sum_{(u,v) \in R} (r_{uv} - p_u^T \cdot q_v)^2 \end{aligned}$$

用SVD(Singular value decomposition)求解?

基于随机梯度下降的学习算法

- 目标函数：

$$\begin{aligned} \mathbf{P}^*, \mathbf{Q}^* &= \arg \min_{\mathbf{P}, \mathbf{Q}} L(\mathbf{P}, \mathbf{Q}) = \arg \min_{\mathbf{P}, \mathbf{Q}} \frac{1}{2} \|\mathbf{R} - \mathbf{P}^T \mathbf{Q}\|^2 \\ &= \arg \min_{\mathbf{P}, \mathbf{Q}} \frac{1}{2} \sum_{(u,v) \in S} (r_{uv} - \mathbf{p}_u^T \cdot \mathbf{q}_v)^2 \end{aligned}$$

- 加入正则化：

$$\mathbf{P}^*, \mathbf{Q}^* = \arg \min_{\mathbf{P}, \mathbf{Q}} \frac{1}{2} \sum_{(u,v) \in S} ((r_{uv} - \mathbf{p}_u^T \cdot \mathbf{q}_v)^2 + \lambda_p \|\mathbf{p}_u\|^2 + \lambda_q \|\mathbf{q}_v\|^2)$$

$$\arg \min_{P,Q} \frac{1}{2} \sum_{(u,v) \in S} ((r_{uv} - \mathbf{p}_u^T \cdot \mathbf{q}_v)^2 + \lambda_p \|\mathbf{p}_u\|^2 + \lambda_q \|\mathbf{q}_v\|^2)$$

- 学习算法步骤

- 随机初始化 P, Q
- do
 - ✓ For 每个训练样本 $(r_{uv}, (u, v) \in S)$ do:

$$① \quad e_{uv} = r_{uv} - \mathbf{p}_u^T \cdot \mathbf{q}_v$$

$$② \quad \mathbf{q}_v \leftarrow \mathbf{q}_v + \eta \cdot (e_{uv} \cdot \mathbf{p}_u^T - \lambda_q \cdot \mathbf{q}_v)$$

$$③ \quad \mathbf{p}_u \leftarrow \mathbf{p}_u + \eta \cdot (e_{uv} \cdot \mathbf{q}_v^T - \lambda_p \cdot \mathbf{p}_u)$$

- Until stopping criteria satisfied

点击率预测 (Click-Through Rate) 问题

Normal App



Ranked by CTR

CTR
Click Through Rate

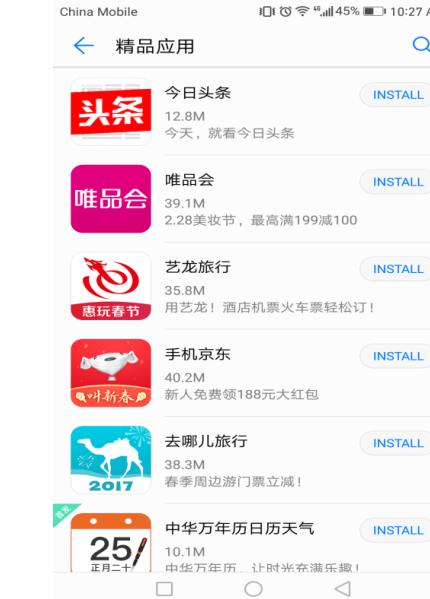
Game App



Ranked by CTR × LTV

LTV
Life Time Value

Advertise App



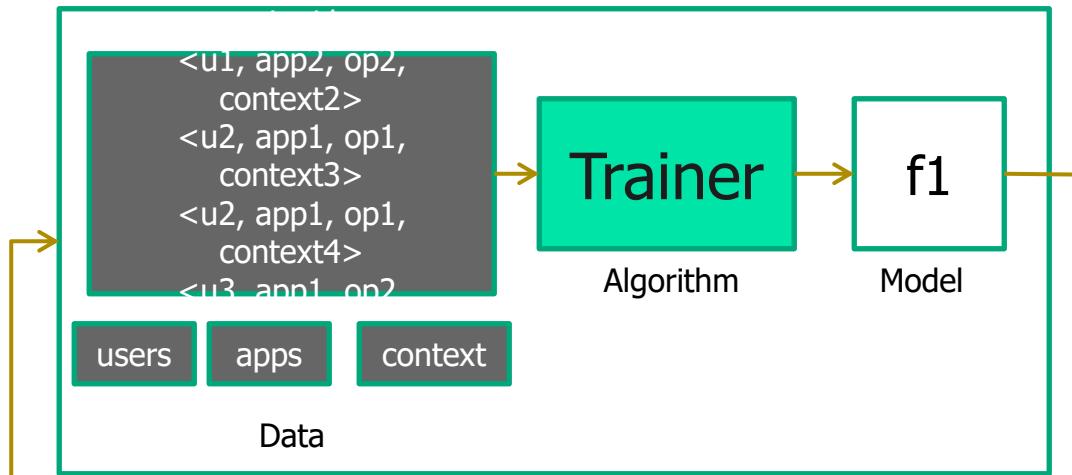
Ranked by CTR × CPC

CPC
Cost Per Click

CTR is critic in recommender system, not only affect the user experience, but also determine the benefits.

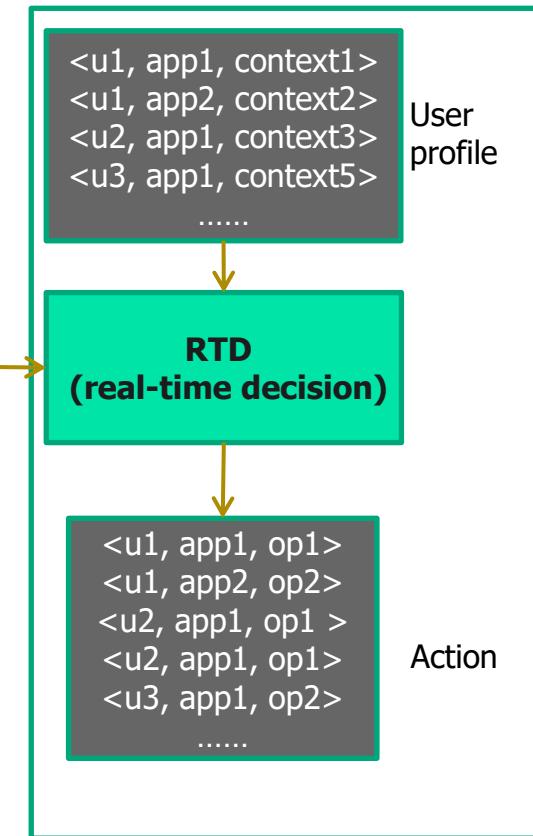
点击率预测 (Click-Through Rate) 问题

✓ Data, Algorithm, Model



Modeling Process

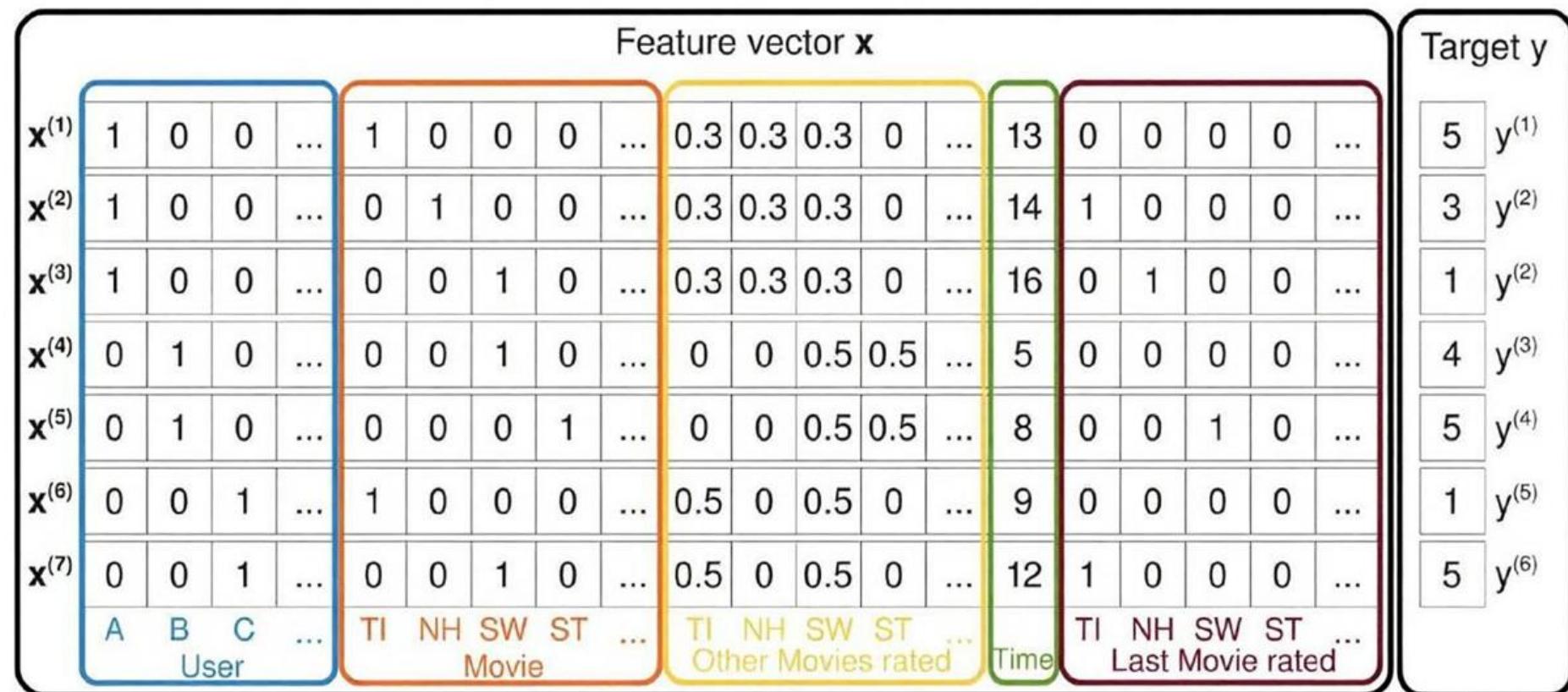
Recommendation Process



从用户-物品矩阵到通用多特征矩阵的挑战

	5	3	
			4
	2		3

User-Item Matrix



基于线性回归的CTR模型

$$y = w_0 + w_1x_1 + w_2x_2 + \cdots + w_nx_n$$

$$y = w_0 + \sum_{i=1}^n w_i x_i$$

无法挖掘特征之间的非线性关系！

二阶多项式回归模型

	x_1	x_2	...	x_{n-1}	x_n
x_1	x_1x_1	x_1x_2	...	x_1x_{n-1}	x_1x_n
x_2	x_2x_1	x_2x_2	...	x_2x_{n-1}	x_2x_n
:	:	:	:	:	:
x_{n-1}	$x_{n-1}x_1$	$x_{n-1}x_2$...	$x_{n-1}x_{n-1}$	$x_{n-1}x_n$
x_n	x_nx_1	x_nx_2	...	x_nx_{n-1}	x_nx_n

挖掘特征之间的二阶非线性关系



$$y = w_0 + \sum_{i=1}^n w_i x_i + \sum_{i=1}^n \sum_{j \geq i}^n w_{ij} x_i x_j$$

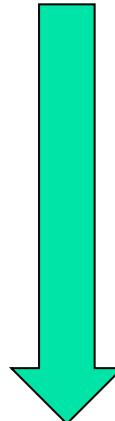
存在问题：

样本中如果没有出现 (x_i 与 x_j) 交互的特征组合，则无法对相应参数 (w_{ij}) 进行估计

因子分解机(Factorization Machines, FM)模型

$$y = w_0 + \sum_{i=1}^n w_i x_i + \sum_{i=1}^n \sum_{j \geq i} w_{ij} x_i x_j$$

二项式参数 w_{ij} 可以组成一个矩阵 W



根据 Cholesky 分解，则可以分解成

$$W = VV^T$$

V 的第 j 列便是第 j 维特征的**隐向量**。即每个参数

$$w_{ij} = \langle v_i, v_j \rangle$$

$$y = w_0 + \sum_{i=1}^n w_i x_i + \sum_{i=1}^n \sum_{j=i+1}^n \langle v_i, v_j \rangle x_i x_j$$

因子分解机(Factorization Machines, FM)模型

$$\hat{y}(\mathbf{x}) := w_0 + \sum_{i=1}^n w_i x_i + \sum_{i=1}^n \sum_{j=i+1}^n \langle \mathbf{v}_i, \mathbf{v}_j \rangle x_i x_j \quad (1)$$

where the model parameters that have to be estimated are:

$$w_0 \in \mathbb{R}, \quad \mathbf{w} \in \mathbb{R}^n, \quad \mathbf{V} \in \mathbb{R}^{n \times k} \quad (2)$$

And $\langle \cdot, \cdot \rangle$ is the dot product of two vectors of size k :

$$\langle \mathbf{v}_i, \mathbf{v}_j \rangle := \sum_{f=1}^k v_{i,f} \cdot v_{j,f} \quad (3)$$

因子分解机(Factorization Machines, FM)模型

$$\hat{y}(\mathbf{x}) := w_0 + \sum_{i=1}^n w_i x_i + \sum_{i=1}^n \sum_{j=i+1}^n \langle \mathbf{v}_i, \mathbf{v}_j \rangle x_i x_j$$

$$\begin{aligned} \sum_{i=1}^n \sum_{j=i+1}^n \langle \mathbf{v}_i, \mathbf{v}_j \rangle x_i x_j &= \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n \langle \mathbf{v}_i, \mathbf{v}_j \rangle x_i x_j - \frac{1}{2} \sum_{i=1}^n \langle \mathbf{v}_i, \mathbf{v}_i \rangle x_i x_i \end{aligned}$$

$$= \frac{1}{2} \left(\sum_{i=1}^n \sum_{j=1}^n \sum_{f=1}^k v_{i,f} v_{j,f} x_i x_j - \sum_{i=1}^n \sum_{f=1}^k v_{i,f} v_{i,f} x_i x_i \right)$$

$$= \frac{1}{2} \sum_{f=1}^k \left(\left(\sum_{i=1}^n v_{i,f} x_i \right) \left(\sum_{j=1}^n v_{j,f} x_j \right) - \sum_{i=1}^n v_{i,f}^2 x_i^2 \right)$$

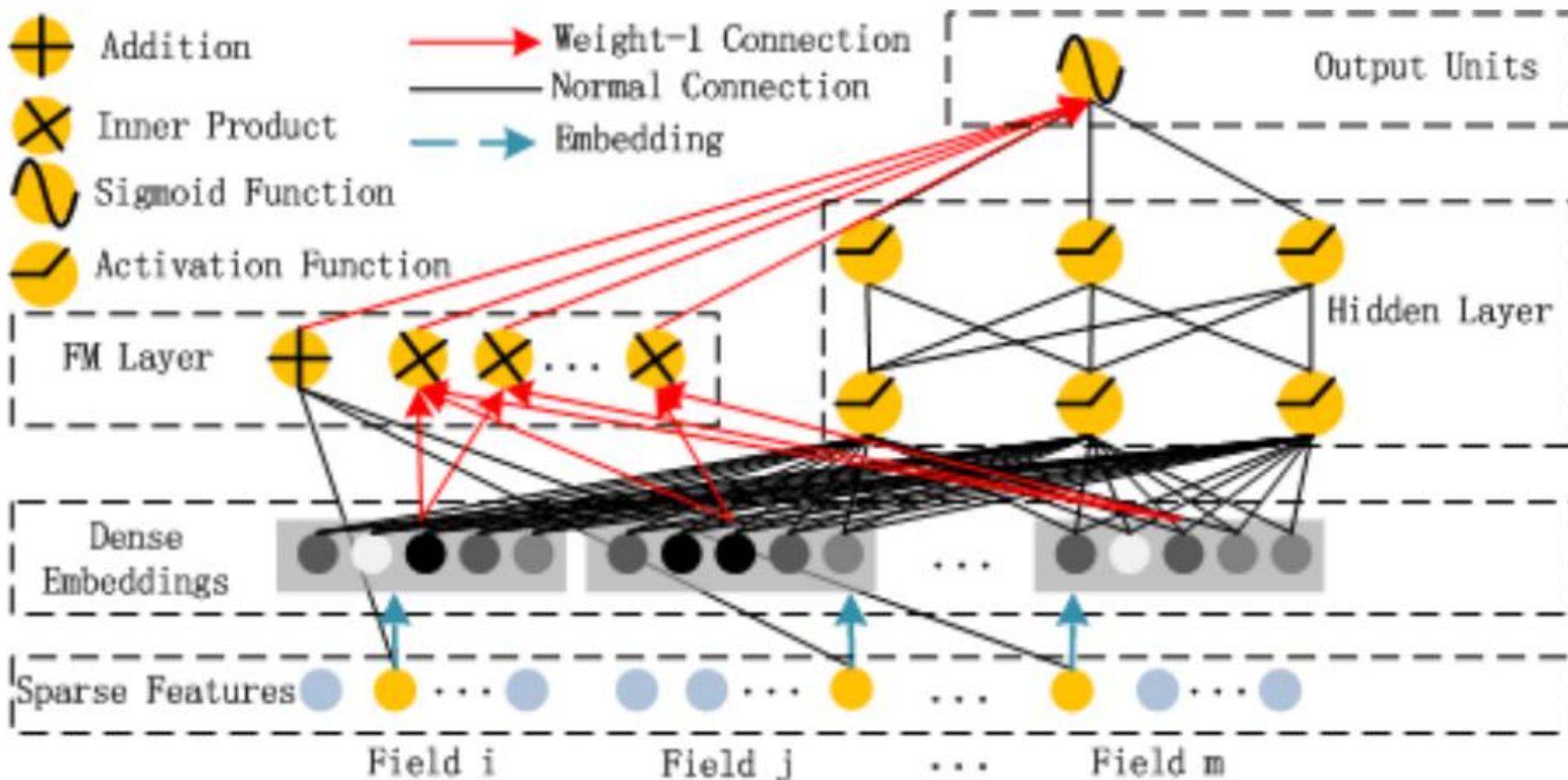
$$= \frac{1}{2} \sum_{f=1}^k \left(\left(\sum_{i=1}^n v_{i,f} x_i \right)^2 - \sum_{i=1}^n v_{i,f}^2 x_i^2 \right)$$

因子分解机(Factorization Machines, FM)模型

$$\hat{y}(\mathbf{x}) := w_0 + \sum_{i=1}^n w_i x_i + \frac{1}{2} \sum_{f=1}^k \left(\left(\sum_{i=1}^n v_{i,f} x_i \right)^2 - \sum_{i=1}^n v_{i,f}^2 x_i^2 \right)$$

$$\frac{\partial}{\partial \theta} \hat{y}(\mathbf{x}) = \begin{cases} 1, & \text{if } \theta \text{ is } w_0 \\ x_i, & \text{if } \theta \text{ is } w_i \\ x_i \sum_{j=1}^n v_{j,f} x_j - v_{i,f} x_i^2, & \text{if } \theta \text{ is } v_{i,f} \end{cases}$$

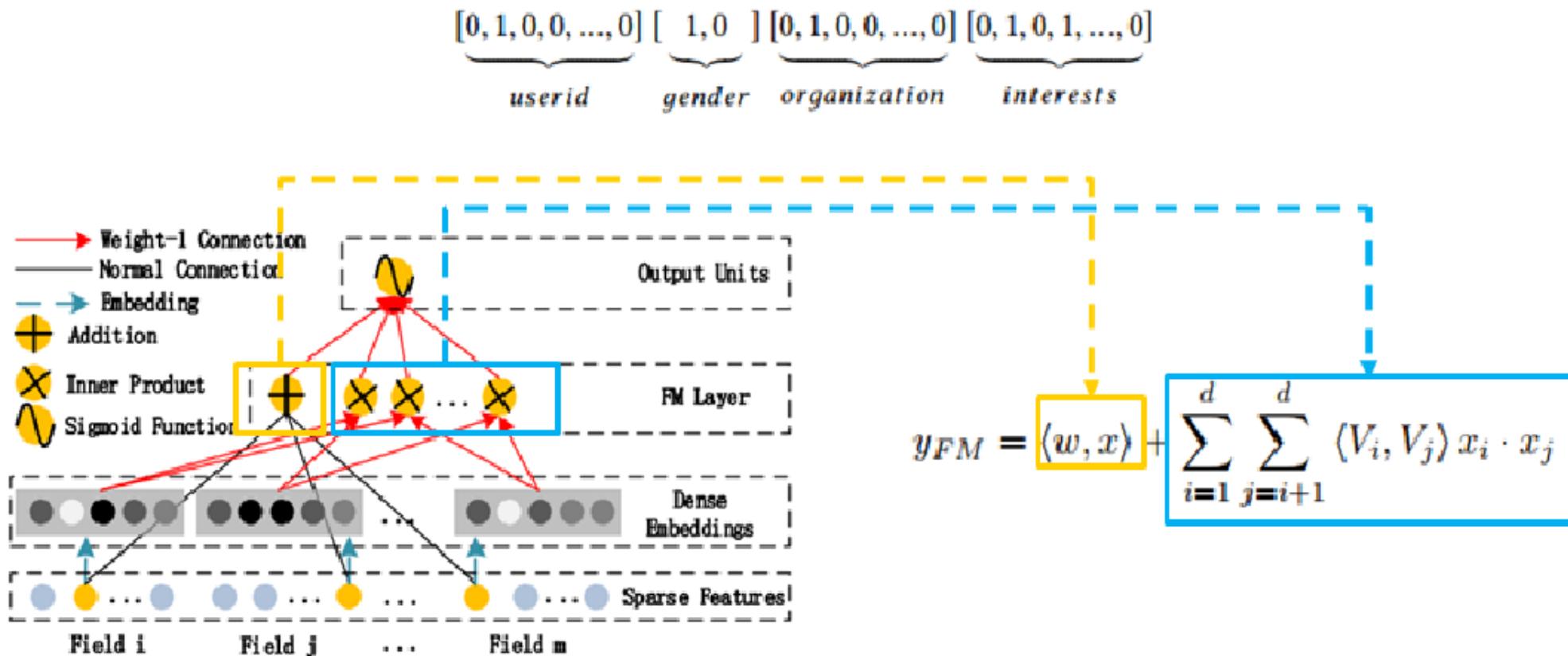
DeepFM推荐模型



DeepFM推荐模型

■ FM-Component

- where $w \in R^d$ and $V_i \in R^k$ (k is dimension of latent vector which is given) . The Addition unit ($\langle w, x \rangle$) reflects the importance of order-1 features, and the Inner Product units represent the impact of order-2 feature interactions.



Acknowledgements

- Some text, figures and formulations are from WWW. Thanks for their sharing. If you have copyright claim please contact with me at yym@hit.edu.cn.
- This lecture is distributed for nonprofit purpose.

Thank You for Your Attention

Contact me at: yym@hit.edu.cn

Tel: 26033008, 13760196623

Address: Rm.1402, H# Building