



数据挖掘导论

Introduction to Data Mining

第六章 推荐系统

刘 淇

Email: qiliuql@ustc.edu.cn

课程主页:

<http://staff.ustc.edu.cn/~qiliuql/DM2017YZ.html>



主要内容

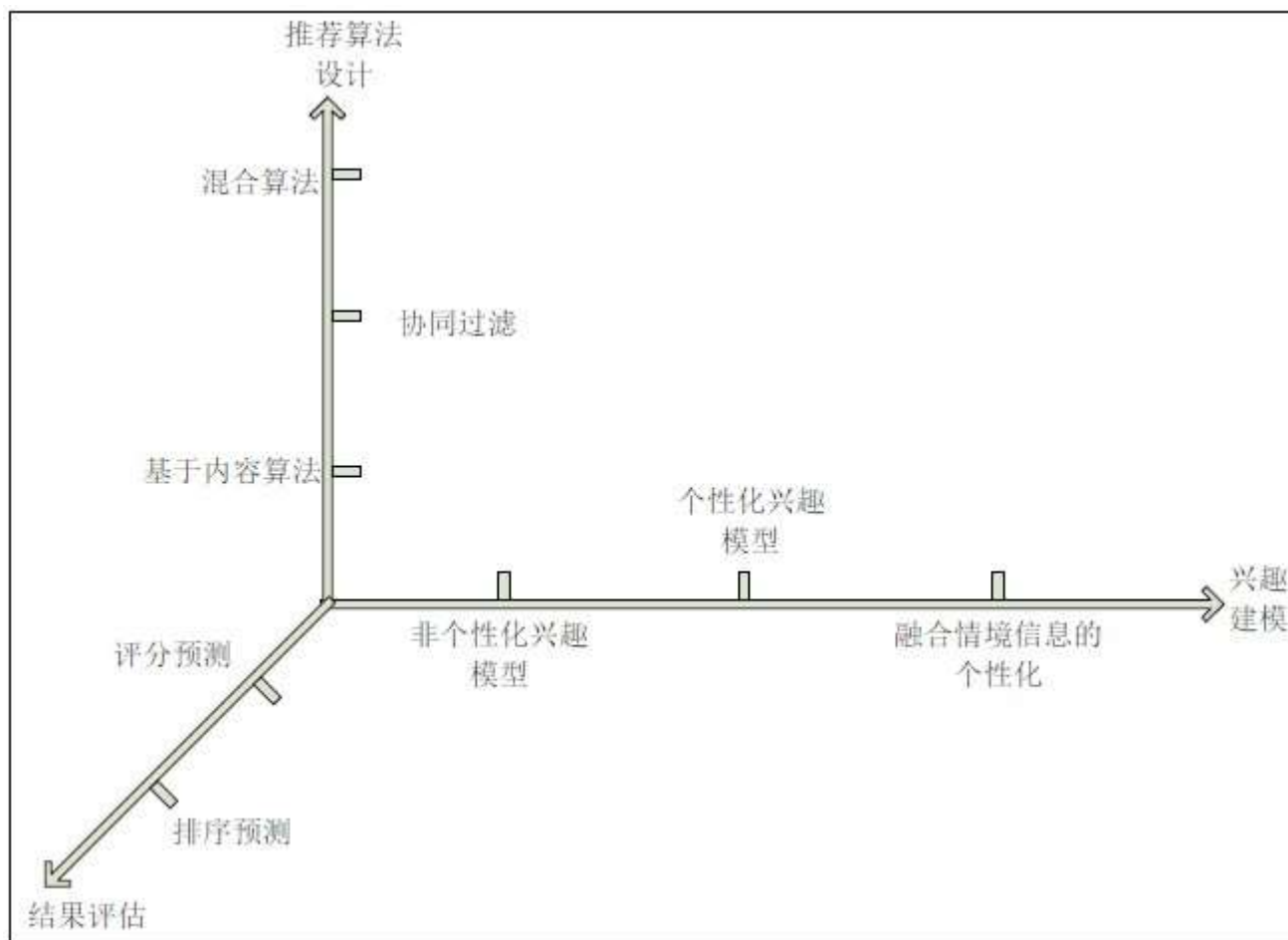
2

- 什么是推荐系统
 - 背景、定义、应用场景
- 推荐方法概述
 - 兴趣建模
 - 推荐算法设计
 - 推荐结果的评估
- 案例学习
 - 基于用户兴趣扩展的个性化推荐方法
 - 面向推荐系统的纠结心理挖掘
- 小结及未来的路
- 资料推荐



推荐算法设计

3

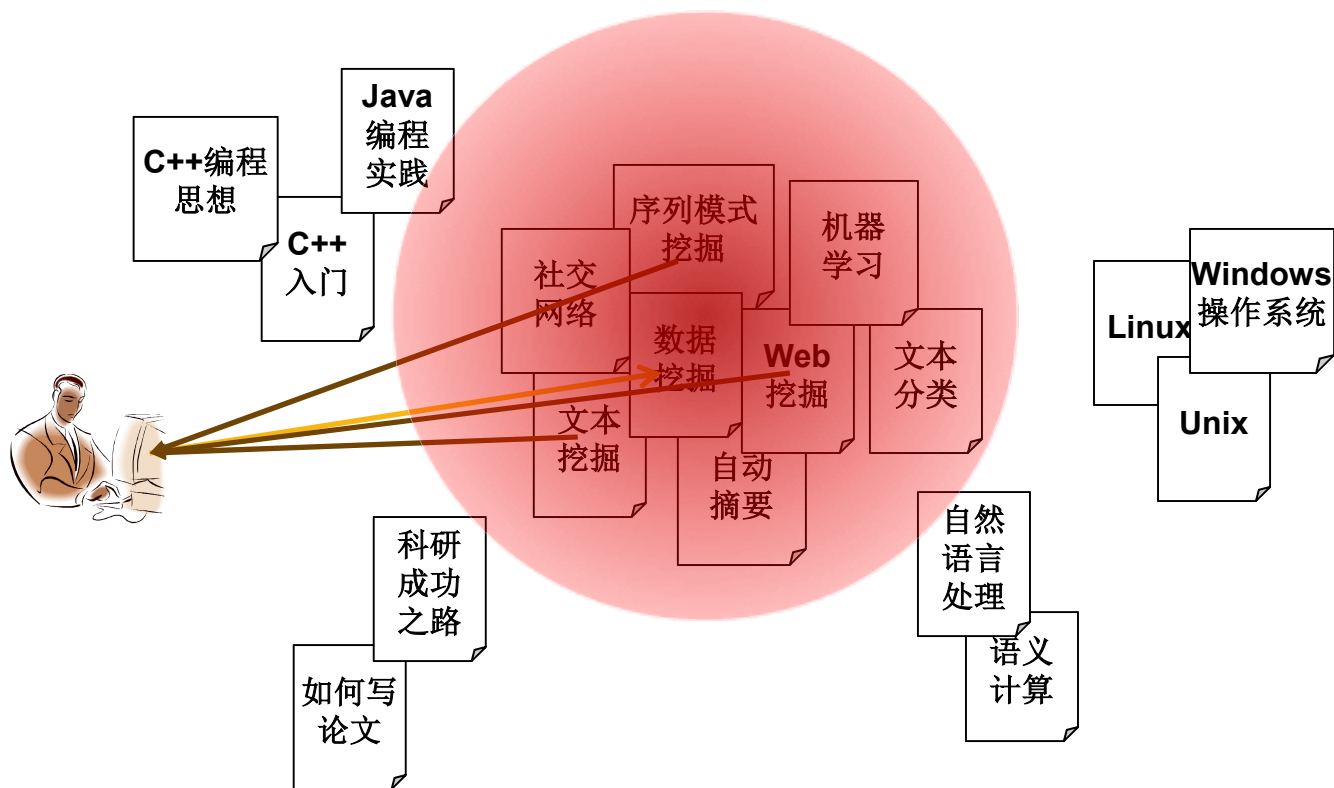




推荐算法设计

4

□ 基于内容推荐算法





推荐算法设计

5

□ 基于内容推荐算法

□ 算法基本步骤:

- 分别对用户和项目建立配置文件;
- 通过分析已经购买（或浏览）过的内容，建立或更新用户的配置文件;
- 比较用户与项配置文件的相似度，并直接向用户推荐与其配置文件最相似的项目。

□ 基于内容的推荐算法根本在于信息获取和信息过滤！



推荐算法设计

6

□ 基于内容推荐算法

□ 令Content(s)为项目s的配置文件，可用向量表示：

$$\mathbf{w}_s = (w_{s1}, w_{s2}, \dots, w_{sk})$$

其中每个分量 w_{si} 表示关键词 k_i 对于项s的重要程度。

□ 令UserProfile(c)为用户c的配置文件，可用向量表示：

$$\mathbf{w}_c = (w_{c1}, w_{c2}, \dots, w_{ck})$$

表示，其中每个分量 w_{ci} 表示关键词 k_i 对于用户c的重要性。



推荐算法设计

7

□ 基于内容推荐算法

□ 配置文件常用特征

■ 网页

- words, hyperlinks, images, tags, comments, titles, URL, topic

■ 音乐

- genre, rhythm, melody, harmony, lyrics, meta data, artists, bands, press releases, expert reviews, loudness, energy, time, spectrum, duration, frequency, pitch, key, mode, mood, style, tempo

■ 用户

- age, sex, job, location, time, income, education, language, family status, hobbies, general interests, Web usage, computer usage, fan club membership, opinion, comments, tags, mobile usage

■ 情境

- time, location, mobility, activity, socializing, emotion



推荐算法设计

8

- 基于内容推荐算法
 - 信息获取
 - TF-IDF
 - 聚类分析
 - Bayes分类
 - 决策树
 - 人工神经网络
 - 信息过滤
 - 设置合理阈值
 - 信息更新
 - 人机交互，在线反馈



推荐算法设计

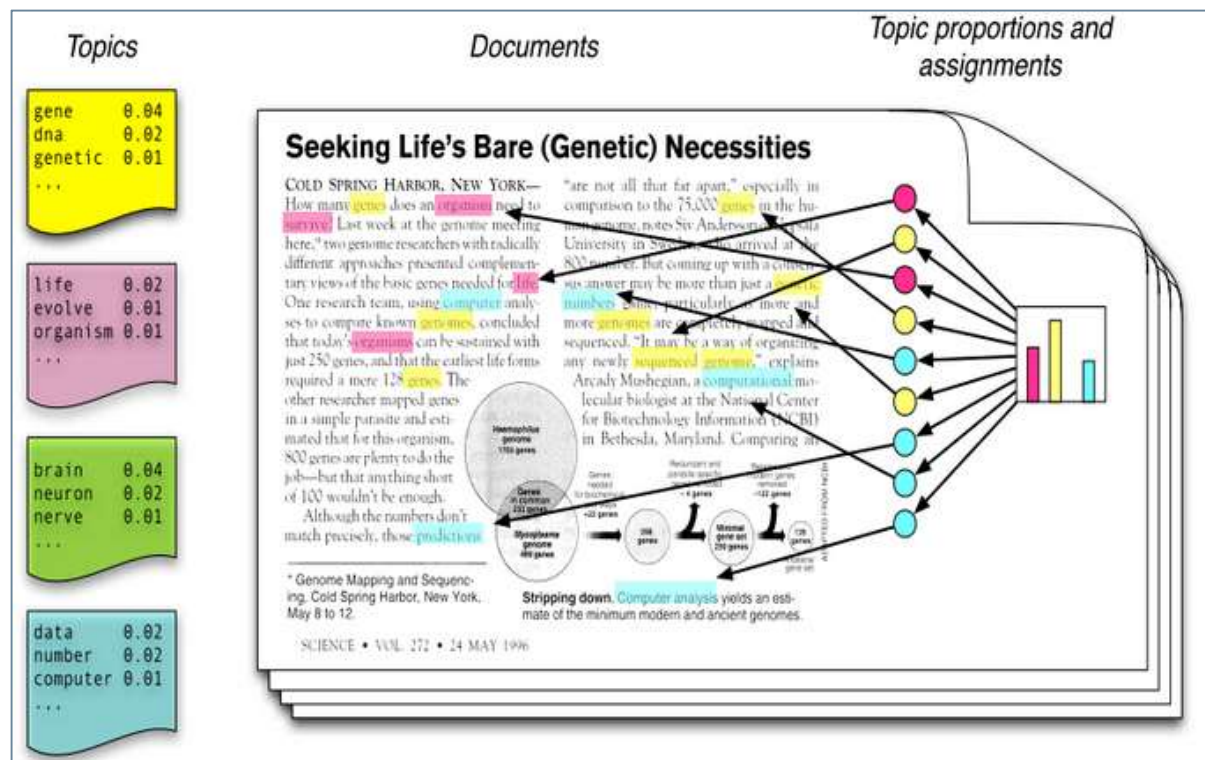
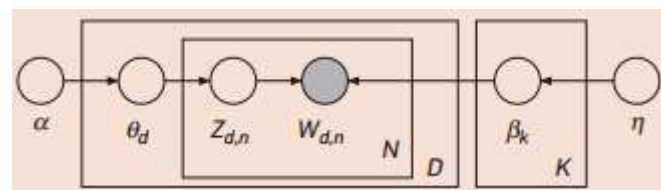
9

基于内容推荐算法

信息获取---降维

概率主题模型：PLSI、LDA

每个文档存在一个主题分布，每个主题上有单词分布





推荐算法设计

10

□ 基于内容推荐算法

□ 用户c对项目s的喜欢程度可以预测为:

$$r_{c,s} = \text{score}(\text{User Profile}(c), \text{Content}(s))$$

□ 可以使用向量 w_c 和 w_s 表示成一个值，例如夹角余弦方法:

$$r_{c,s} = \cos(w_c, w_s) = \frac{w_c \cdot w_s}{\|w_c\|_2 \times \|w_s\|_2}$$



推荐算法设计

11

□ 基于内容推荐算法

□ 优点：

- 简单易懂；
- 可以处理新用户和新项问题（冷启动）；
- 可以不受打分稀疏性问题的约束；
- 通过列出推荐项目的内容特征，可以解释为什么推荐这些项

□ 缺点：

- 受到信息获取技术的约束，难于提取多媒体数据（图形、视频流、声音流等）的内容特征；
- 难以发现用户潜在的但自己尚未觉察的兴趣偏好；
- 对用户不希望得到相似内容的推荐系统（如新闻推荐系统），该方法并不适用。



推荐算法设计

12

□ 协同过滤推荐算法

□ 基于用户(User-based)算法

- 利用历史评分信息计算用户之间的相似性;
- 根据相似度得到邻居用户集, 利用邻居用户在目标项上的评分信息来预测目标用户对目标项的喜好程度;
- 根据计算所得的喜好程度对目标用户进行推荐。

□ 需要数据:

- 用户评分矩阵user-item

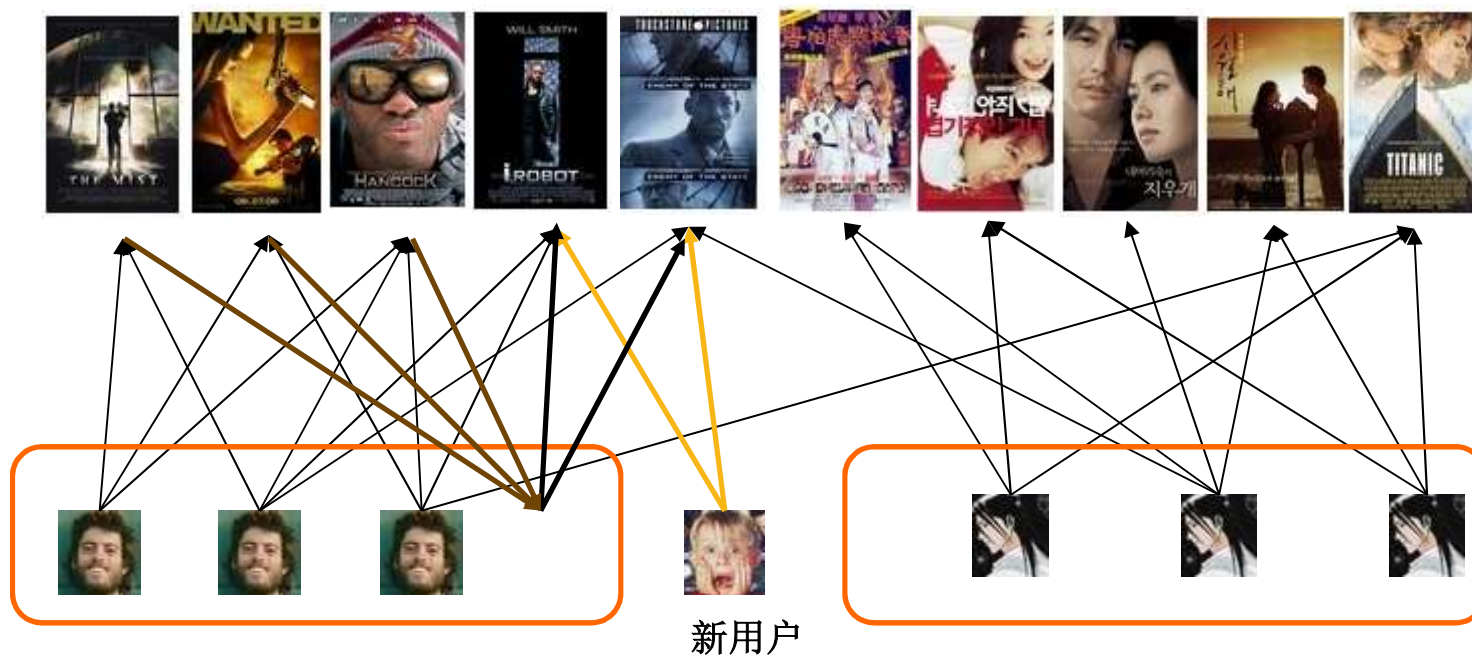
	Item-1	Item-2	Item-3	Item-4	Item-5	Item-6
User-1	4	*	2	5	*	*
User-2	3	2	1	*	*	3
User-3	*	2	*	3	*	4
User-4	*	3	3	5	4	*
User-5	5	*	3	4	*	*



推荐算法设计

13

- 协同过滤推荐算法
 - 基于用户(User-based)算法





推荐算法设计

14

□ 基于用户的协同过滤算法

□ 用户集合: $C = \{c_1, c_2, \dots, c_N\}$; 项集合: $S = \{s_1, s_2, \dots, s_M\}$

□ 用户c对项s的喜好预测值为:

$$r_{c,s} = \bar{r}_c + k \sum_{\hat{c} \in \hat{C}} \text{sim}(c, \hat{c}) \cdot (r_{\hat{c},s} - \bar{r}_{\hat{c}})$$

其中 k 为一个标准化因子, 通常 $k = 1 / \sum_{\hat{c} \in \hat{C}} |\text{sim}(c, \hat{c})|$

\hat{C} 为与用户c相似度较高且已经对项s进行评价了的用户集合,
 $\text{sim}(i, j)$ 表示用户 i 和 j 之间的相似性,

用户c的平均打分 \bar{r}_c 定义为 $\bar{r}_c = \sum_{s \in S_c} r_{c,s} / |S_c|$, 其中 $S_c = \{s \in S \mid r_{c,s} \neq 0\}$



推荐算法设计

15

□ 基于用户的协同过滤算法--KNN

□ Pearson相似性:

$$sim(x, y) = \frac{\sum_{s \in S_{xy}} (r_{x,s} - \bar{r}_x)(r_{y,s} - \bar{r}_y)}{\sqrt{\sum_{s \in S_{xy}} (r_{x,s} - \bar{r}_x)^2 (r_{y,s} - \bar{r}_y)^2}}$$

□ 夹角余弦相似性:

$$sim(x, y) = \cos(x, y) = \frac{\sum_{s \in S_{xy}} r_{x,s} r_{y,s}}{\sqrt{\sum_{s \in S_{xy}} r_{x,s}^2 r_{y,s}^2}}$$

□ 其中, $S_{xy} = S_x \cap S_y$ 定义为用户x和y共同打过分的项集合。



推荐算法设计

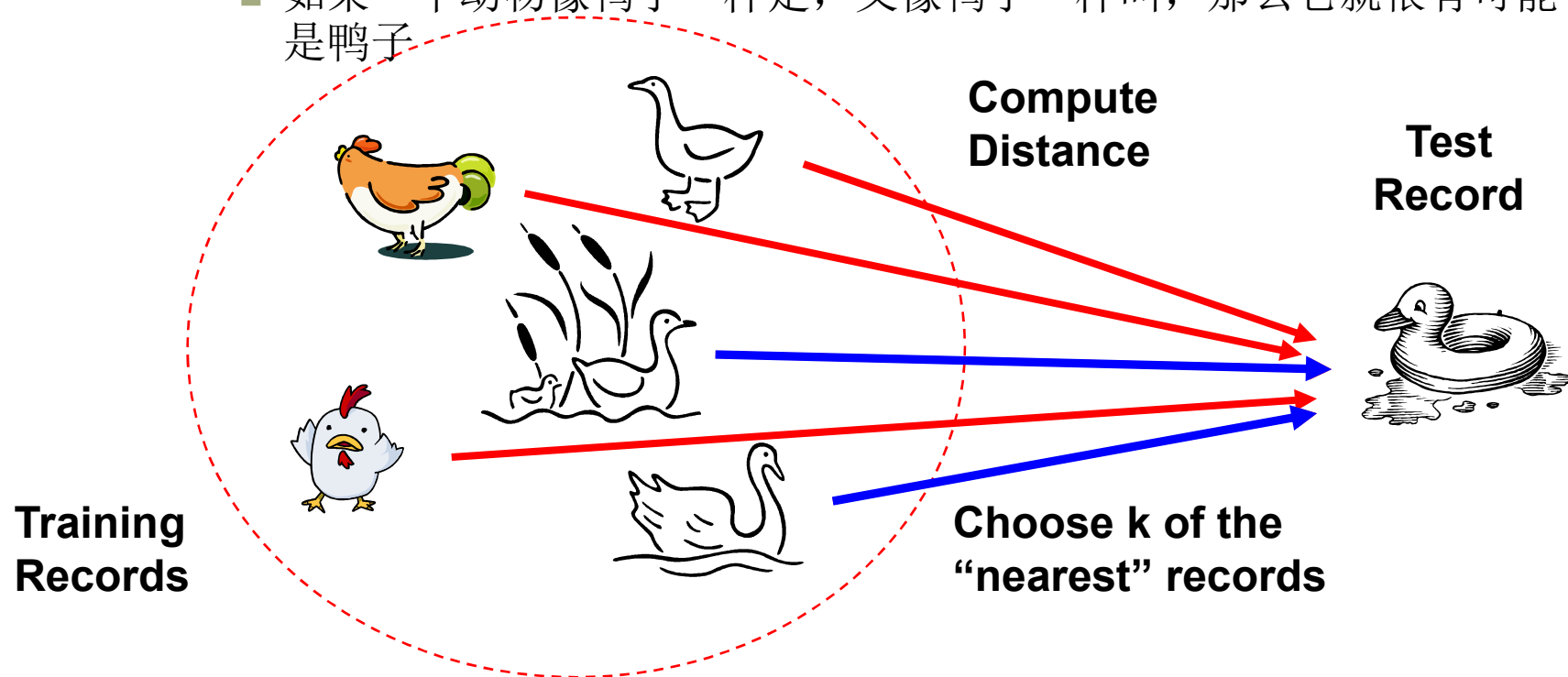
16

□ 基于用户的协同过滤算法

□ 关键：计算用户间的相似度（对用户聚类）

■ K近邻（KNN）

- 如果一个动物像鸭子一样走，又像鸭子一样叫，那么它就很有可能是鸭子





推荐算法设计

17

□ 协同过滤推荐算法

□ 基于项目(Item-based)的算法

- 利用历史评分信息计算项之间的相似性;
- 根据相似度得到邻居项集, 利用目标用户对这些项的评分信息来预测目标用户对目标项的喜好程度;
- 根据计算所得的喜好程度对目标用户进行推荐。

□ 需要数据:

- 用户评分矩阵user-item

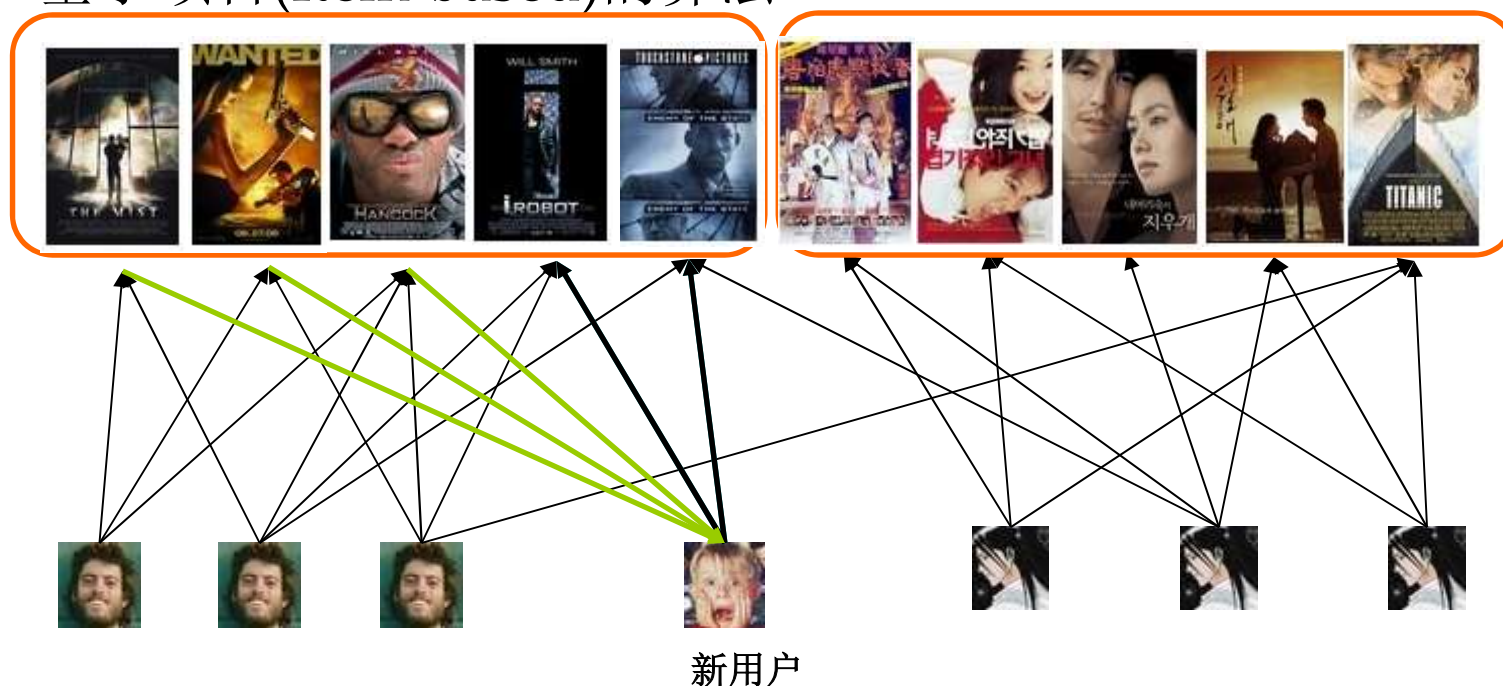
	Item-1	Item-2	Item-3	Item-4	Item-5	Item-6
User-1	4	*	2	5	*	*
User-2	3	2	1	*	*	3
User-3	*	2	*	3	*	4
User-4	*	3	3	5	4	*
User-5	5	*	3	4	*	*



推荐算法设计

18

- 协同过滤推荐算法
 - 基于项目(Item-based)的算法



与基于内容推荐算法（根据项目的配置文件计算项目相似度）不同，**ItemCF**根据用户的行为计算项目相似度



推荐算法设计

19

□ 基于项目的协同过滤算法

- 用户集合: $C = \{c_1, c_2, \dots, c_N\}$; 项集合: $S = \{s_1, s_2, \dots, s_M\}$
- 用户c对项s的评分预测值为:

$$r_{c,s} = \frac{\sum_{all\ similar\ items, t} (sim(s, t) \cdot r_{c,t})}{\sum_{all\ similar\ items, t} (|sim(s, t)|)}$$

其中, $sim(i, j)$ 是项 i 和 j 之间的相似性程度



推荐算法设计

20

□ 基于项目的协同过滤算法

□ 项i和项j之间的相似性计算（项目的距离度量）：

■ Pearson相似性：

$$sim(i, j) = \frac{\sum_{c \in C_{ij}} (r_{c,i} - \bar{r}_i)(r_{c,j} - \bar{r}_j)}{\sqrt{\sum_{c \in C_{ij}} (r_{c,i} - \bar{r}_i)^2 (r_{c,j} - \bar{r}_j)^2}}$$

■ 改进的夹角余弦相似性：

$$sim(i, j) = \frac{\sum_{c \in C_{ij}} (r_{c,i} - \bar{r}_c)(r_{c,j} - \bar{r}_c)}{\sqrt{\sum_{c \in C_{ij}} (r_{c,i} - \bar{r}_c)^2 (r_{c,j} - \bar{r}_c)^2}}$$

其中， $C_{ij} = C_i \cap C_j$ 定义为对项i和j都打过分的用户集合。



推荐算法设计

21

□ 基于用户/项目的协同过滤算法

□ 常用相似度

■ 基于评分(Rating-based)的相似度

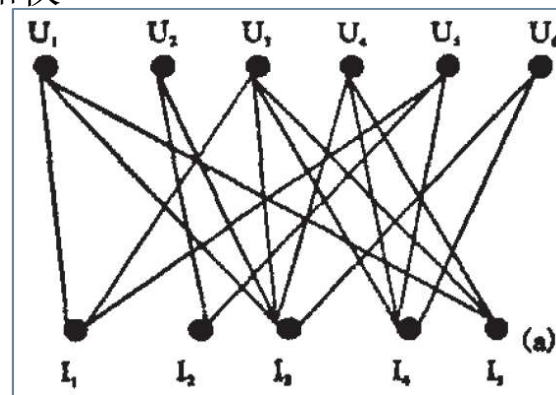
- 在很多电子商务网站中，用户都可以给消费的商品评分
 - 夹角余弦
 - Pearson相似度
 - 各种变形：例如根据活跃度、时间等加权

■ 结构性相似度

- 根据数据的网络结果来计算相似度
 - Node-dependent similarity
 - Path-dependent similarity
 - Random-walk-based similarity

■ 外部信息相似度

- 用户profile信息相似度
- 项目的特征信息相似度





推荐算法设计

22

□ UserCF与ItemCF的比较

	UserCF	ItemCF
原理	用户与朋友的爱好相同	与用户之前喜欢的产品类似的项目
典型应用	推荐文章的Digg	Amazon、Netflix
推荐的侧重点	小型兴趣群体的社会化爱好	更个性化，反映用户兴趣的传承
技术上	维护用户-用户相似度	维护项目-项目相似度
性能	适用于用户较少的场合	适用于项目较少的场合
领域	时效性强、兴趣不够个性化的领域	长尾物品丰富、用户个性化需求强烈的领域
推荐理由	经常有“惊喜”、不好解释	根据用户历史行为的中规中矩推荐，容易被接受



推荐算法设计

23

□ 其他协同过滤算法

□ 基于模型(Model-based)

■ 基于对已有数据应用统计和机器学习得到的模型进行预测

■ 把推荐当成一个聚类问题

■ K-means...

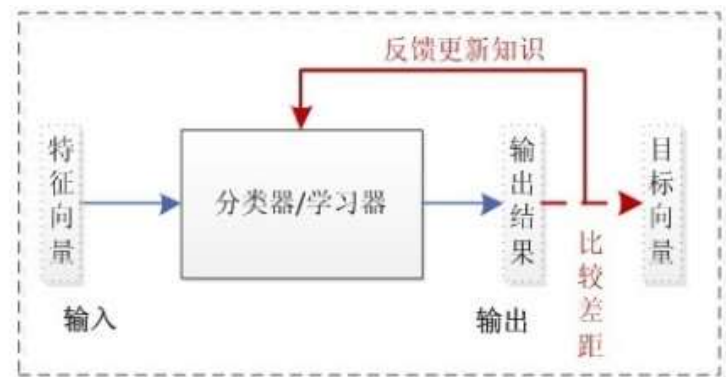
■ 把推荐当成一个分类问题：买？ Or 不买？

■ SVM、贝叶斯分类器、决策树...

■ 把推荐当成一个回归预测问题

■ 评分预测分解模型：预测一个数值

■ 学习排序分解模型：预测一个排序





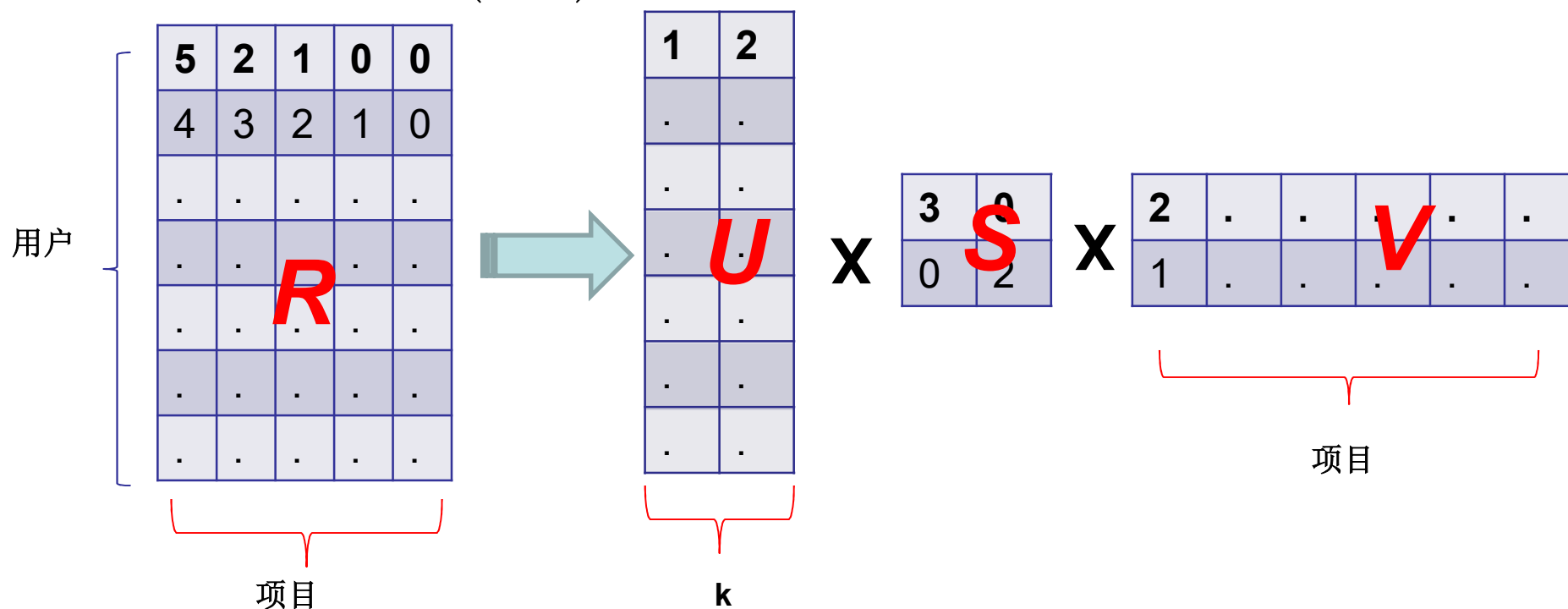
推荐算法设计

24

□ 基于矩阵分解的协同过滤算法

□ 面向评分预测的模型

■ 奇异值分解(SVD)

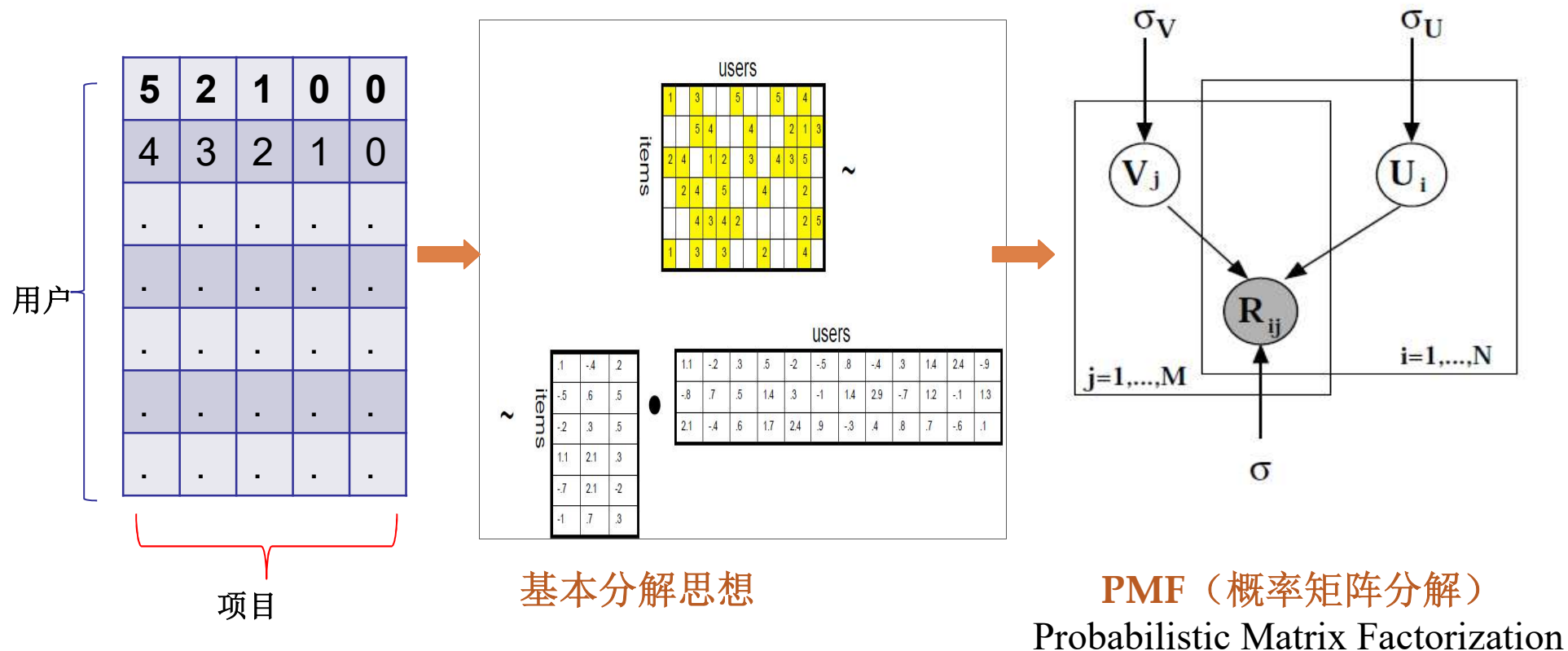




评分预测算法设计

25

- 基于矩阵分解的协同过滤算法
 - 面向评分预测的模型





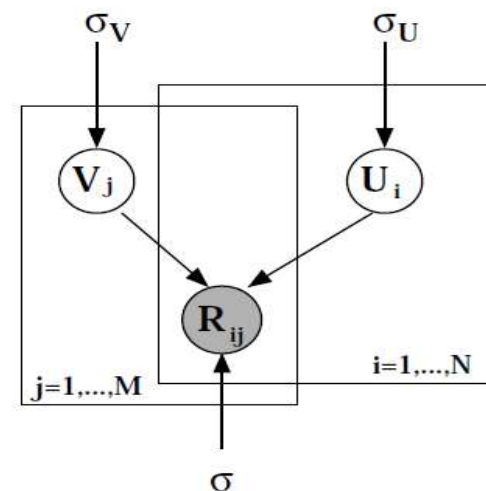
评分预测算法设计

26

基于矩阵分解的协同过滤算法

PMF Solution

$$p(R|U, V, \sigma^2) = \prod_{i=1}^N \prod_{j=1}^M \left[\mathcal{N}(R_{ij} | U_i^T V_j, \sigma^2) \right]^{I_{ij}}$$



How to get U and V?

- The log-posterior of user and item features over fix parameters

$$p(U, V | R, \sigma^2, \sigma_U^2, \sigma_V^2)$$

$$\propto p(R | U, V, \sigma^2) * p(U | \sigma_U^2) * p(V | \sigma_V^2)$$

Likelihood!

Prior

$$p(V | \sigma_V^2) = \prod_{j=1}^M \mathcal{N}(V_j | 0, \sigma_V^2 \mathbf{I})$$

$$p(U | \sigma_U^2) = \prod_{i=1}^N \mathcal{N}(U_i | 0, \sigma_U^2 \mathbf{I})$$



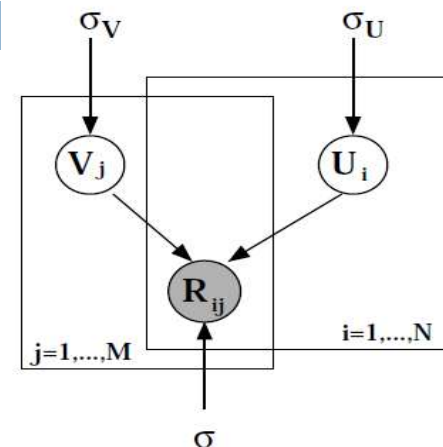
评分预测算法设计

27

□ 概率矩阵分解

□ MAP learning

$$\ln p(U, V | R, \sigma^2, \sigma_U^2, \sigma_V^2) = -\frac{1}{2\sigma^2} \sum_{i=1}^N \sum_{j=1}^M I_{ij} (R_{ij} - U_i^T V_j)^2 - \frac{1}{2\sigma_U^2} \sum_{i=1}^N U_i^T U_i - \frac{1}{2\sigma_V^2} \sum_{j=1}^M V_j^T V_j - \frac{1}{2} \left(\left(\sum_{i=1}^N \sum_{j=1}^M I_{ij} \right) \ln \sigma^2 + ND \ln \sigma_U^2 + MD \ln \sigma_V^2 \right) + C, \quad (3)$$



□ Equivalent to minimize sum-of-squared-errors with quadratic regularization terms.

$$E = \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^M I_{ij} (R_{ij} - U_i^T V_j)^2 + \frac{\lambda_U}{2} \sum_{i=1}^N \|U_i\|_{Fro}^2 + \frac{\lambda_V}{2} \sum_{j=1}^M \|V_j\|_{Fro}^2$$

$$\lambda_U = \frac{\sigma^2}{\sigma_U^2}, \lambda_V = \frac{\sigma^2}{\sigma_V^2}$$

regularization term
to avoid over fitting



评分预测算法设计

28

□ 概率矩阵分解

1) Initialize U, V with small, random values

2) repeat

for each record in the training data

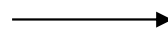
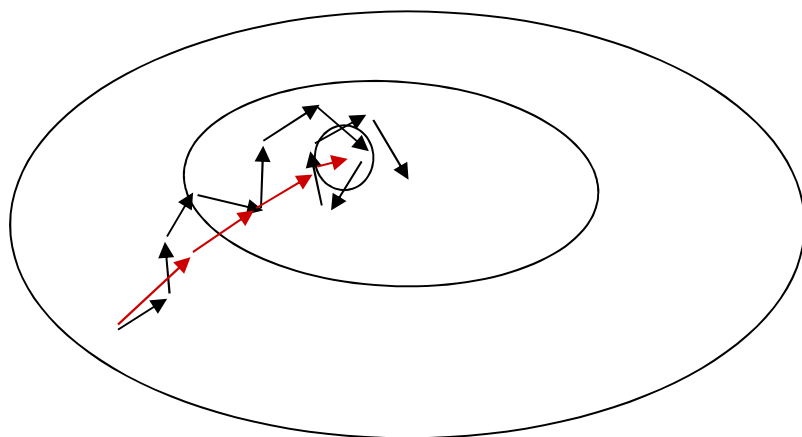
$$2.a) U_i = U_i - a \frac{\partial E}{\partial U_i} = U_i - a \left(\sum_j I_{ij} (R_{ij} - U_i^T V_j) (-V_j) + \lambda_U U_i \right)$$

$$2.b) V_j = V_j - a \frac{\partial E}{\partial V_j} = V_j - a \left(\sum_i I_{ij} (R_{ij} - U_i^T V_j) (-U_i) + \lambda_V V_j \right)$$

until convergence

$$E = \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^M I_{ij} (R_{ij} - U_i^T V_j)^2 + \frac{\lambda_U}{2} \sum_{i=1}^N \|U_i\|_{Fro}^2 + \frac{\lambda_V}{2} \sum_{j=1}^M \|V_j\|_{Fro}^2$$

**Optimize: stochastic
gradient descent**



stochastic updates



full updates (averaged over all data-items)

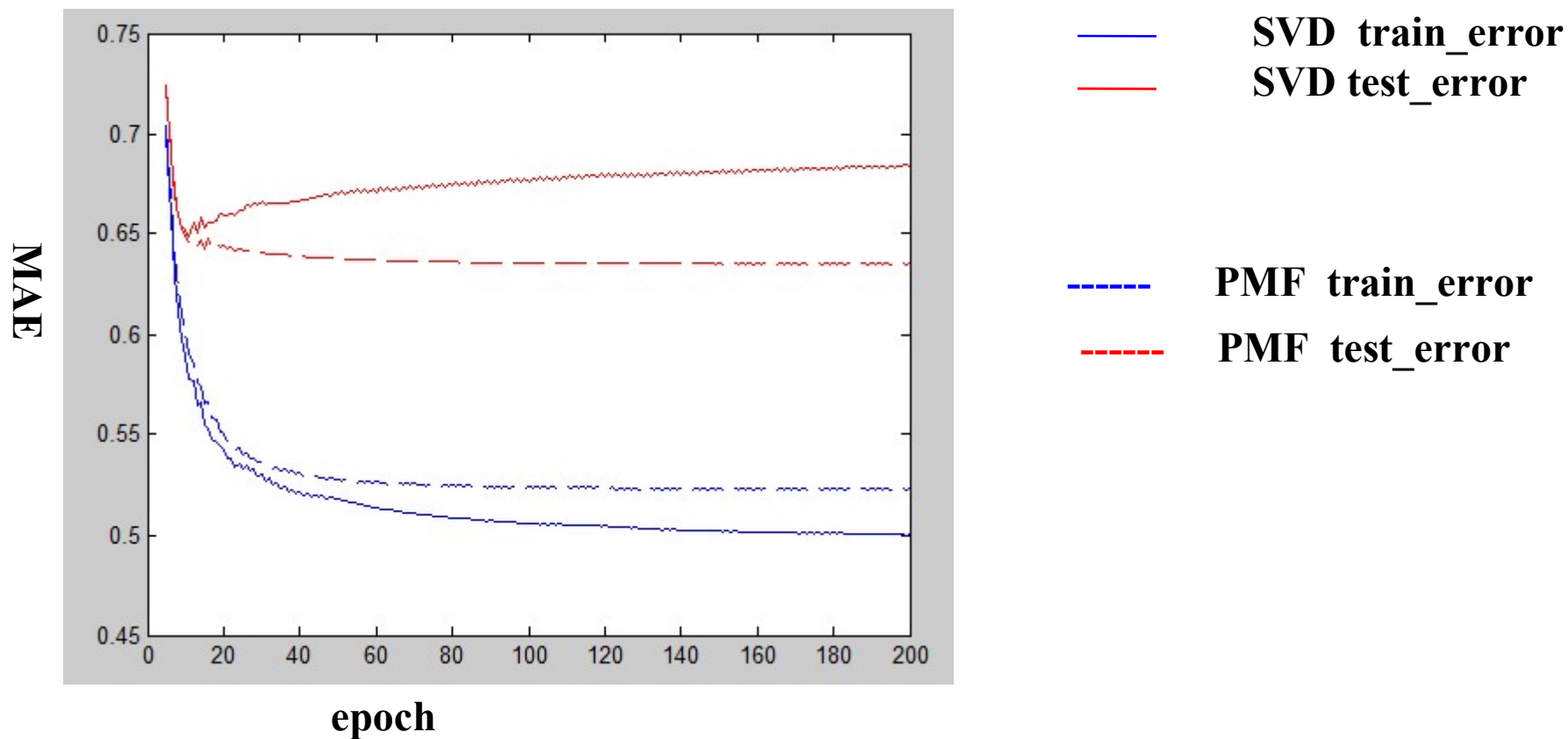


评分预测算法设计

29

□ 概率矩阵分解测试

Using Movielens dataset





排序预测算法设计

30

基于排序学习的协同过滤

- 无显式的评分，只有隐式的购买行为，如何产生推荐？
 - 用一个模型预测产品之间的排序关系，把排在前边的产品 用户推荐给用户----BPR模型

- 低秩分解的时候去拟项目pair的排序关系

$$\hat{x}_{ui} = \langle w_u, h_i \rangle = \sum_{f=1}^k w_{uf} \cdot h_{if}$$

$$\hat{x}_{uij} := \hat{x}_{ui} - \hat{x}_{uj}$$

目标函数: $p(i >_u j | \Theta) := \sigma(\hat{x}_{uij}(\Theta))$

where σ is the logistic sigmoid:

$$\sigma(x) := \frac{1}{1 + e^{-x}}$$

- 然后用分解的结果去预测未知的项目pair排序关系，由此生成排序序列

	A	B	C	D	E
U1	1	0	1	0	0
U2	1	0	0	1	1

项目

项目pair的排序关系

(U1,A,B)	$A >_{U1} B$
(U1,A,D)	$A >_{U1} D$
(U1,A,E)	$A >_{U1} E$
(U1,C,B)	$C >_{U1} B$
(U1,C,D)	$C >_{U1} D$
(U1,C,E)	$C >_{U1} E$
(U2,A,B)	$A >_{U2} B$
.....	

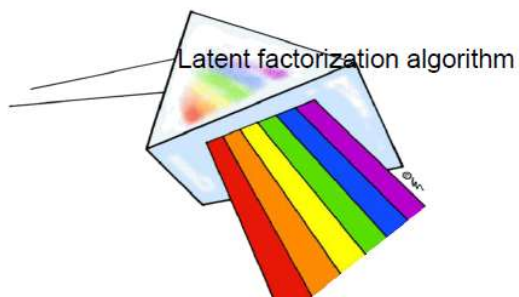


推荐算法设计

31

□ 基于矩阵分解的协同过滤算法

**Rating
Matrix**



Latent factors

Item-1	1.2	0.8	2.1	1.7	0.01	0.25
Item-2	1.1	0.01	1.19	1.35	1.25	0.37
Item-3	0.95	2.1	0.1	0.37	0.55	1.1
⋮	⋮	⋮	⋮	⋮	⋮	⋮
Item-17,770	0.44	0.12	0.43	0.76	0.87	0.17
User-1	0.08	0.49	0.37	1.2	0.67	1.3
User-2	0.77	1.1	0.04	0.97	1.05	1.95
User-3	0.19	0.13	0.88	1.2	1.87	1.1
⋮	⋮	⋮	⋮	⋮	⋮	⋮
User-480K	1.4	1.9	1.4	0.37	0.95	0.7

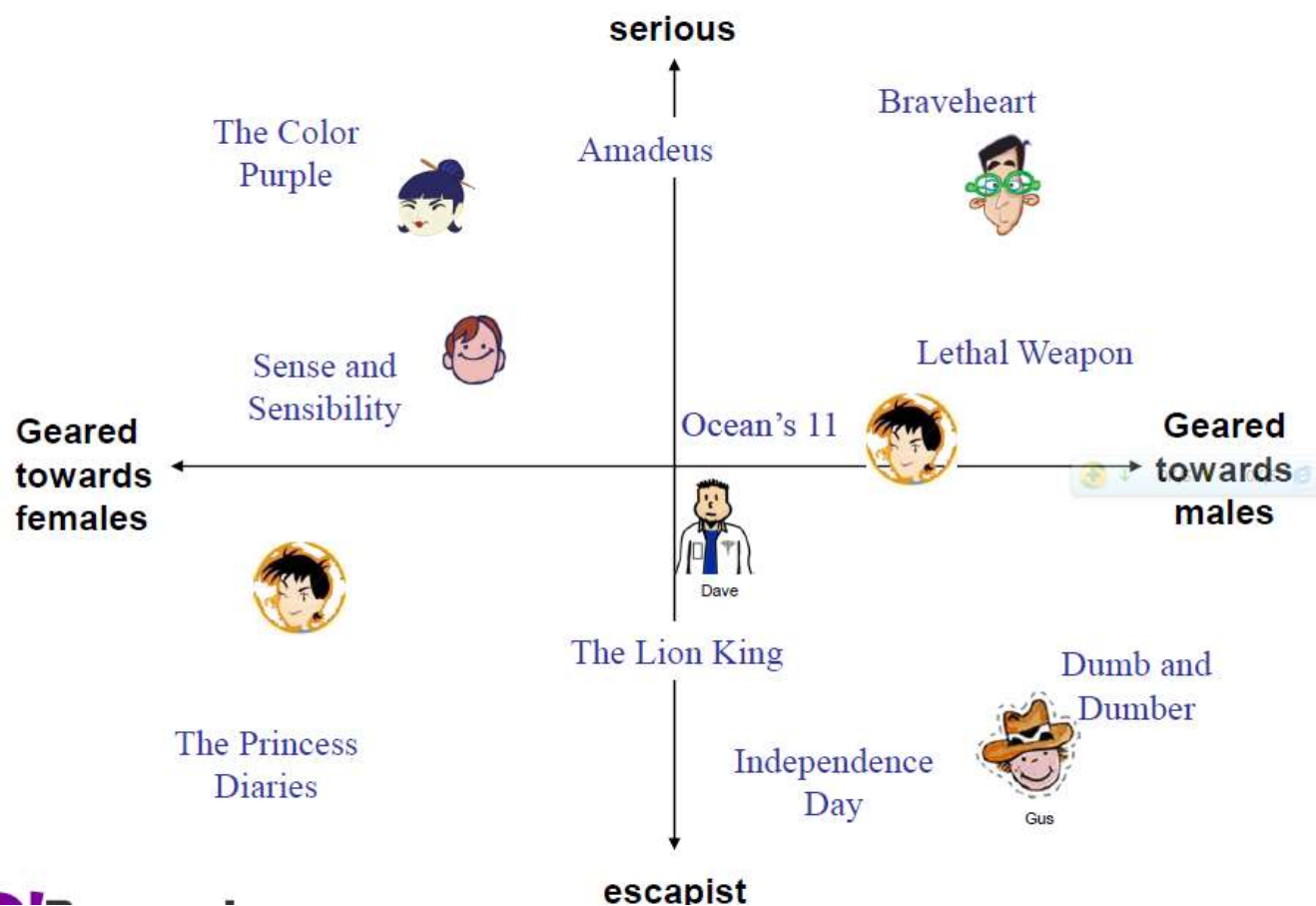
Users/items are arranged in dense latent space



推荐算法设计

32

□ 基于矩阵分解的协同过滤算法





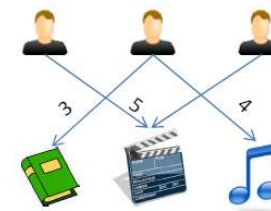
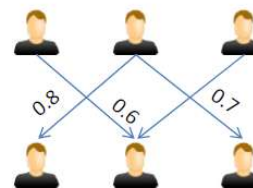
推荐算法设计

33

- 面向社交的个性化推荐
 - 社交推荐



- 项目推荐 vs 社交推荐
 - 用户-项目推荐 → 用户-用户推荐
 - 静态属性 → 网络动态演变
 - 单向行为 → 双向互动

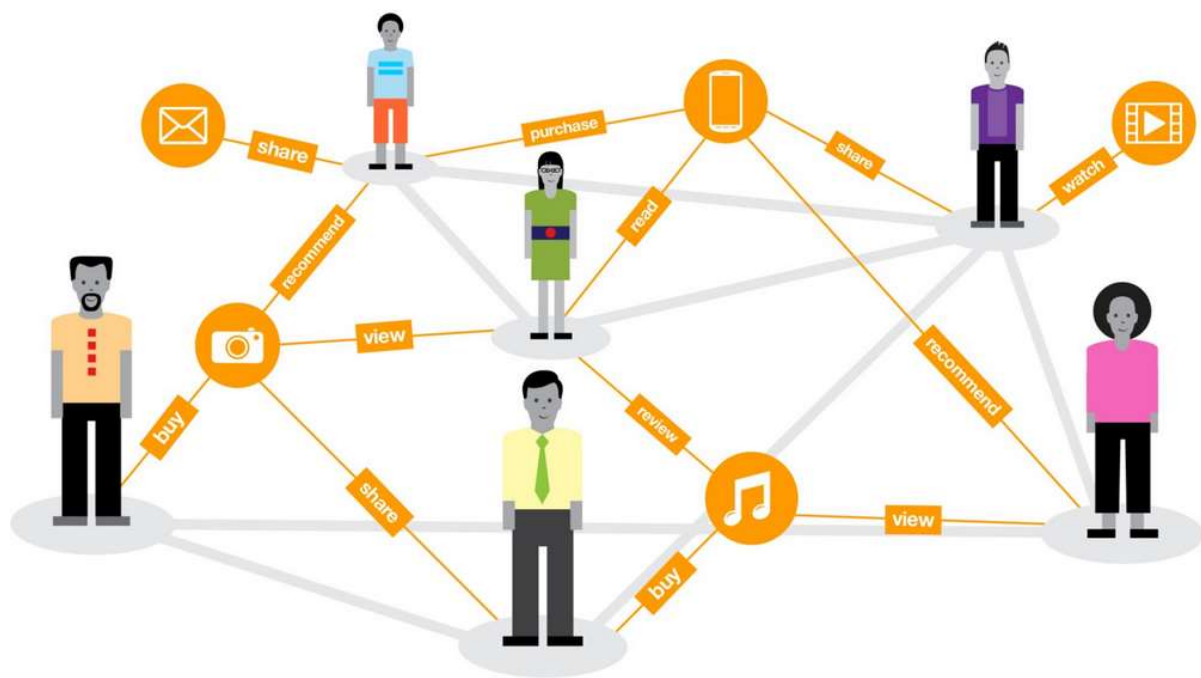




推荐算法设计

34

- 面向社交的个性化推荐
 - 融合社交关系的推荐



社交吸引力

社交影响力

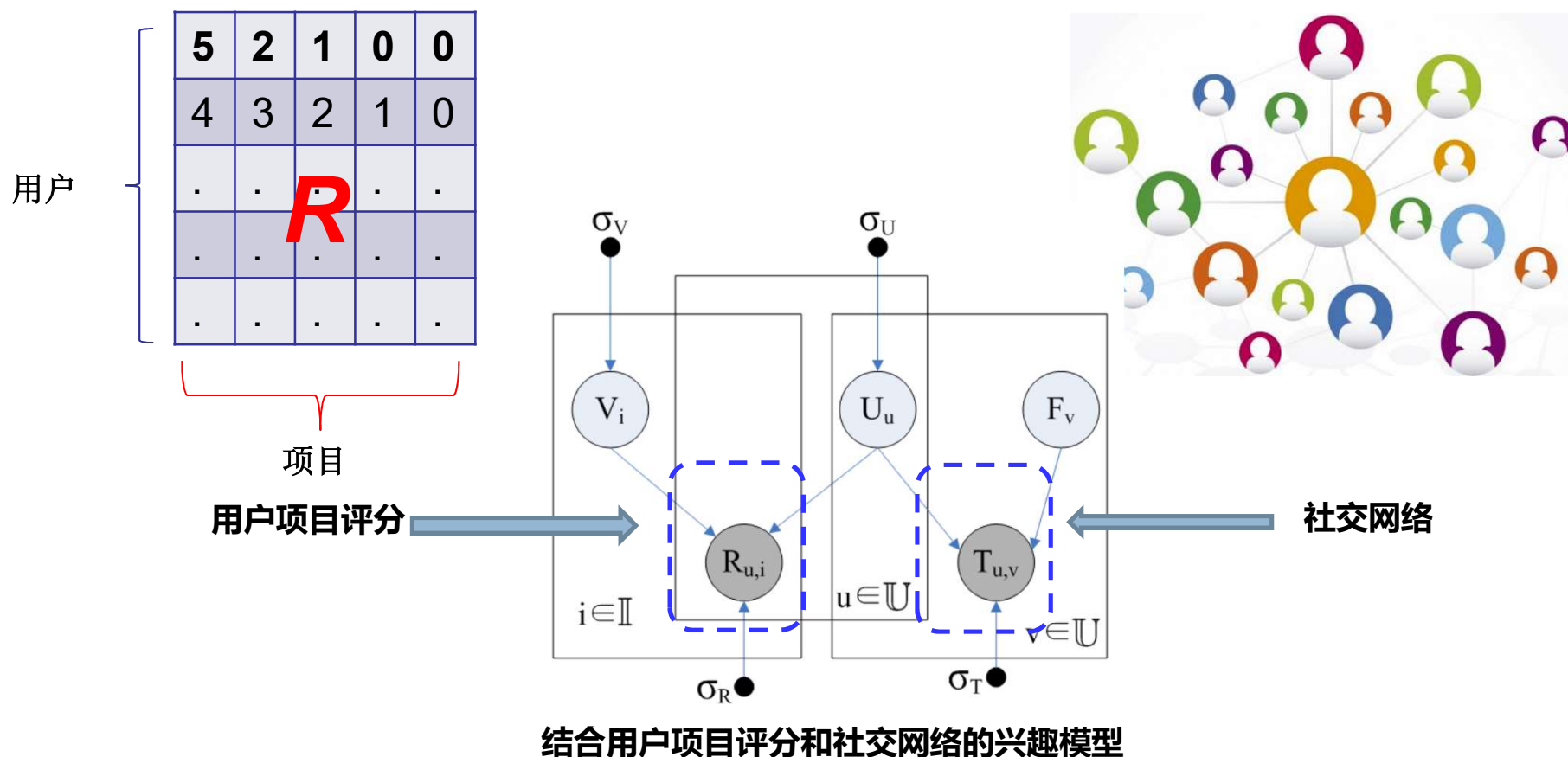
社交偏好



推荐算法设计

35

□ 面向社交的个性化推荐方法





推荐算法设计

36

□ 协同过滤算法

□ 优点：

- 具有推荐新信息的能力，可以发现用户潜在的但自己尚未觉察的兴趣偏好；
- 能够推荐艺术品、音乐、电影等难以进行内容分析的项；

□ 缺点：

- 新用户、新项问题（冷启动）；
- 消费数据的稀疏性问题；
- 算法的可扩展性问题。



推荐算法设计

37

- 混合推荐算法
 - 独立系统相结合——多种方法取其优
 - 协同过滤结合基于内容的特性
 - 基于内容算法结合协同过滤的特性
 - 构建一种统一算法：结合考虑协同过滤和基于内容算法

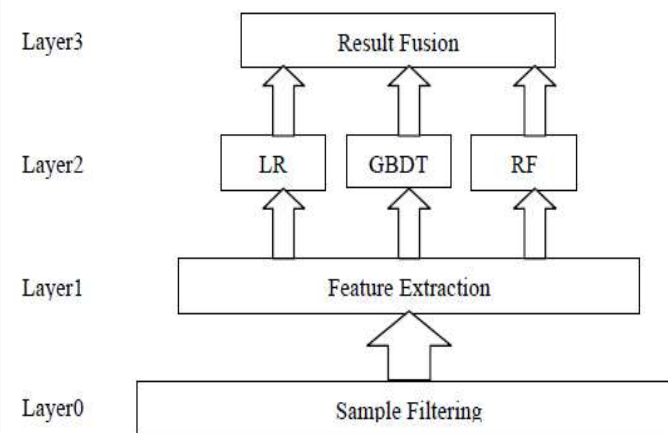
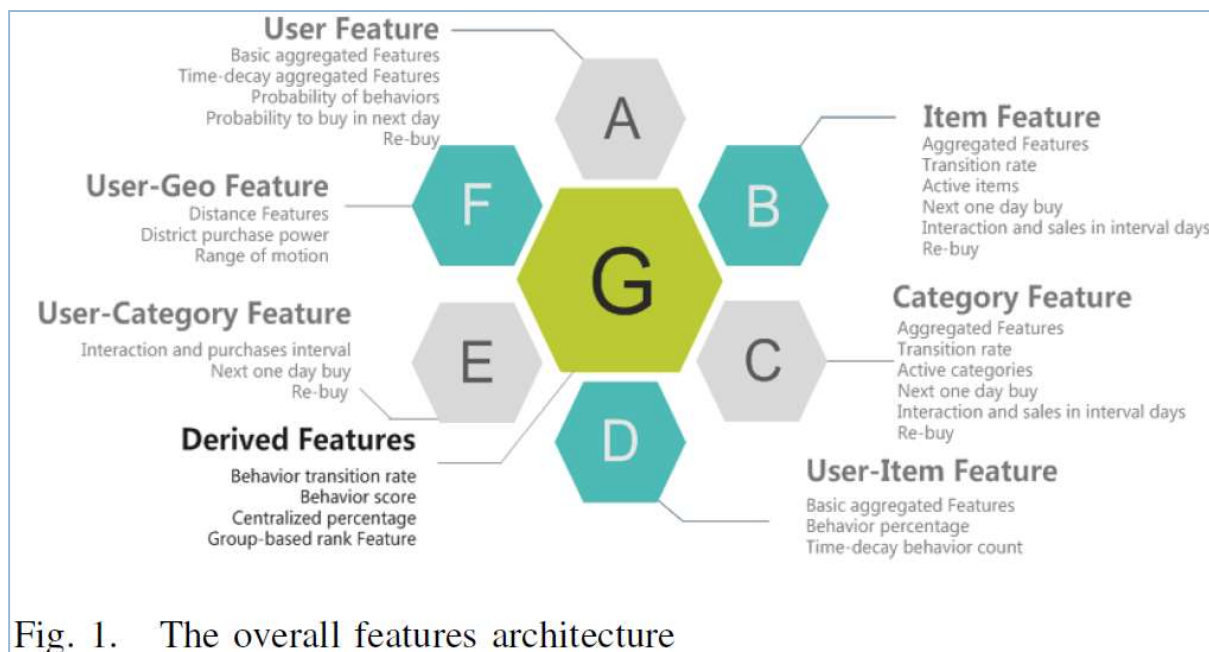




推荐算法设计

38

- 工程应用中的混合推荐（如点击率预测）方法
 - 成千上百万特征的提取和建模
 - Logistic Regression (LR)
 - Factorization Machines(FM、Field-aware FM)
 - GBDT(LR+GBDT)--Gradient Boosting Decision Tree



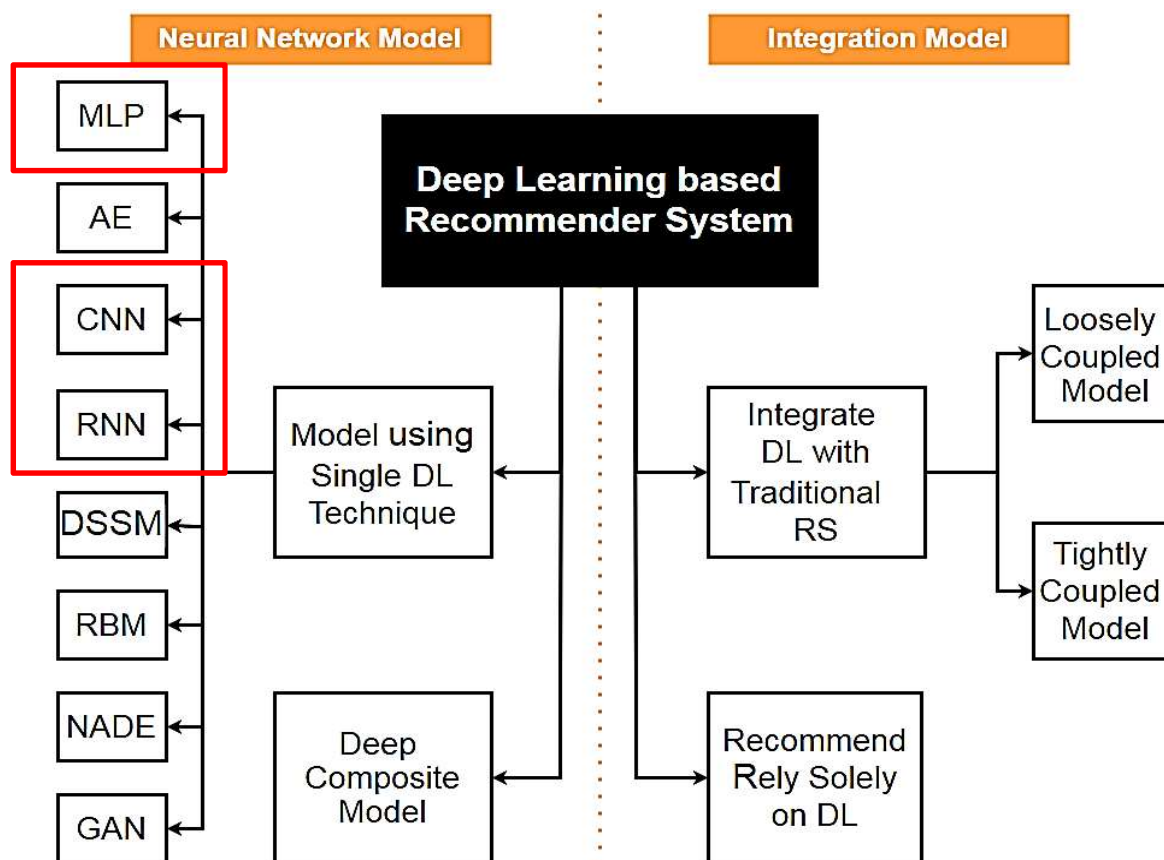


推荐算法设计

39

□ 基于深度学习的推荐算法

□ 框架：





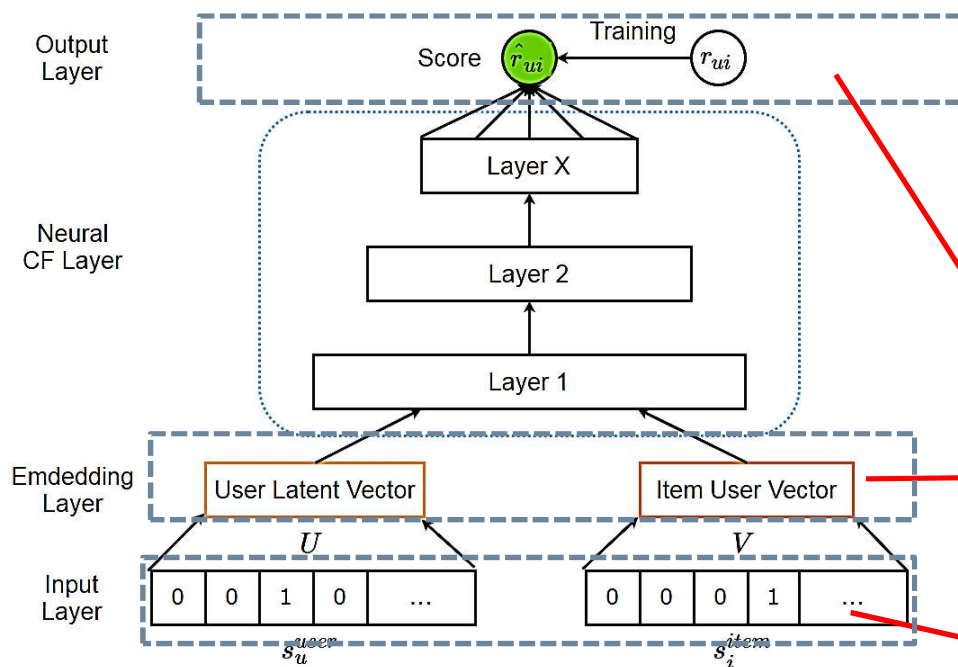
推荐算法设计

40

基于深度学习的推荐算法

多层感知机

神经协同过滤算法 (NCF)



用户

5	2	1	0	0
4	3	2	1	0
.
.
.

项目

用户在项目
上的评分

用户和项目的嵌
入表示层 (稠密)

用户和项目的特
征向量 (稀疏)



推荐算法设计

41

□ 基于深度学习的推荐算法

□ 多层感知机

■ 神经协同过滤算法 (NCF)

■ 改变矩阵分解算法中的线性关系

$$r_{ui} = U_u^T \cdot V_i$$

矩阵分解：点积

$$r_{ui} = f(U^T \cdot s_u^{user}, V^T \cdot s_i^{item} | U, V, \theta)$$

神经协同过滤：多层感知机

用户

5	2	1	0	0
4	3	2	1	0
.
.
.

项目



推荐算法设计

42

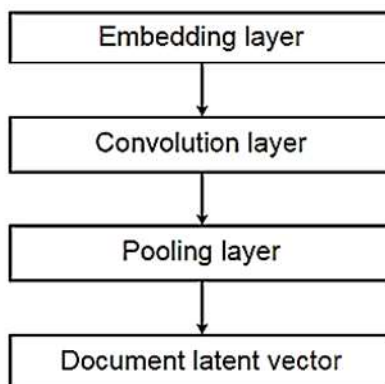
基于深度学习的推荐算法

卷积神经网络

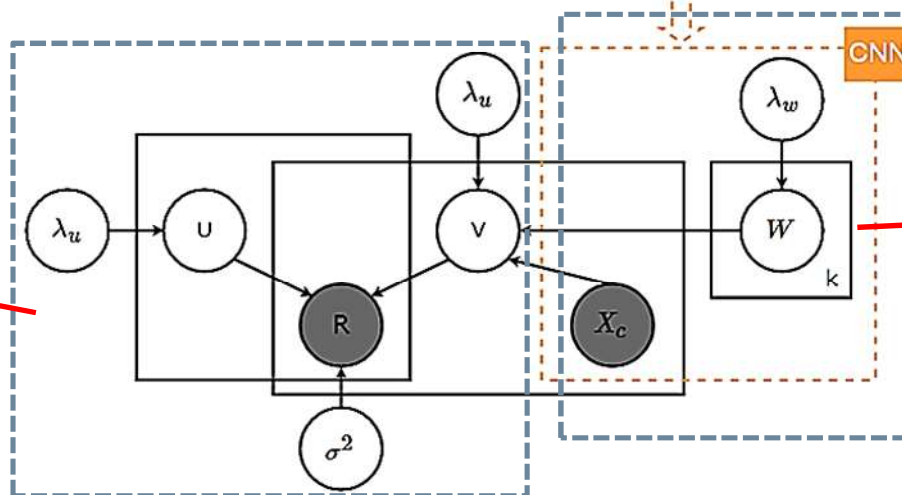
卷积矩阵分解 (ConvMF)

- 与矩阵分解算法融合
- 神经网络提取项目特征

$$L(U, V, W) = \sum_i^N \sum_j^M \frac{I_{ij}}{2} (r_{ij} - u_i^T v_j)_2 + \frac{\lambda_U}{2} \sum_i^U \|u_i\|_2^2 + \frac{\lambda_V}{2} \sum_j^U \|v_j - \text{cnn}(W, X_j)\|_2^2 + \frac{\lambda_W}{2} \sum_k^{w_k} \|w_k\|_2^2$$



矩阵分解框架



卷积神经网络
提取项目特征



推荐算法设计

43

□ 基于深度学习的推荐算法

□ 卷积神经网络

■ 卷积矩阵分解 (ConvMF)

- 利用神经网络提取项目特征
 - 抽取项目的文本、图片表述以及上下文信息
- 融合矩阵分解
 - 实现协同过滤和推荐

$$P(V|W, X, \sigma_v^2) = \prod_j^M N(v_j | \text{cnn}(W, X_j), \sigma_v^2 I)$$

项目原始特征（文本、图片表述以及上下文信息）

卷积神经网络参数



推荐算法设计

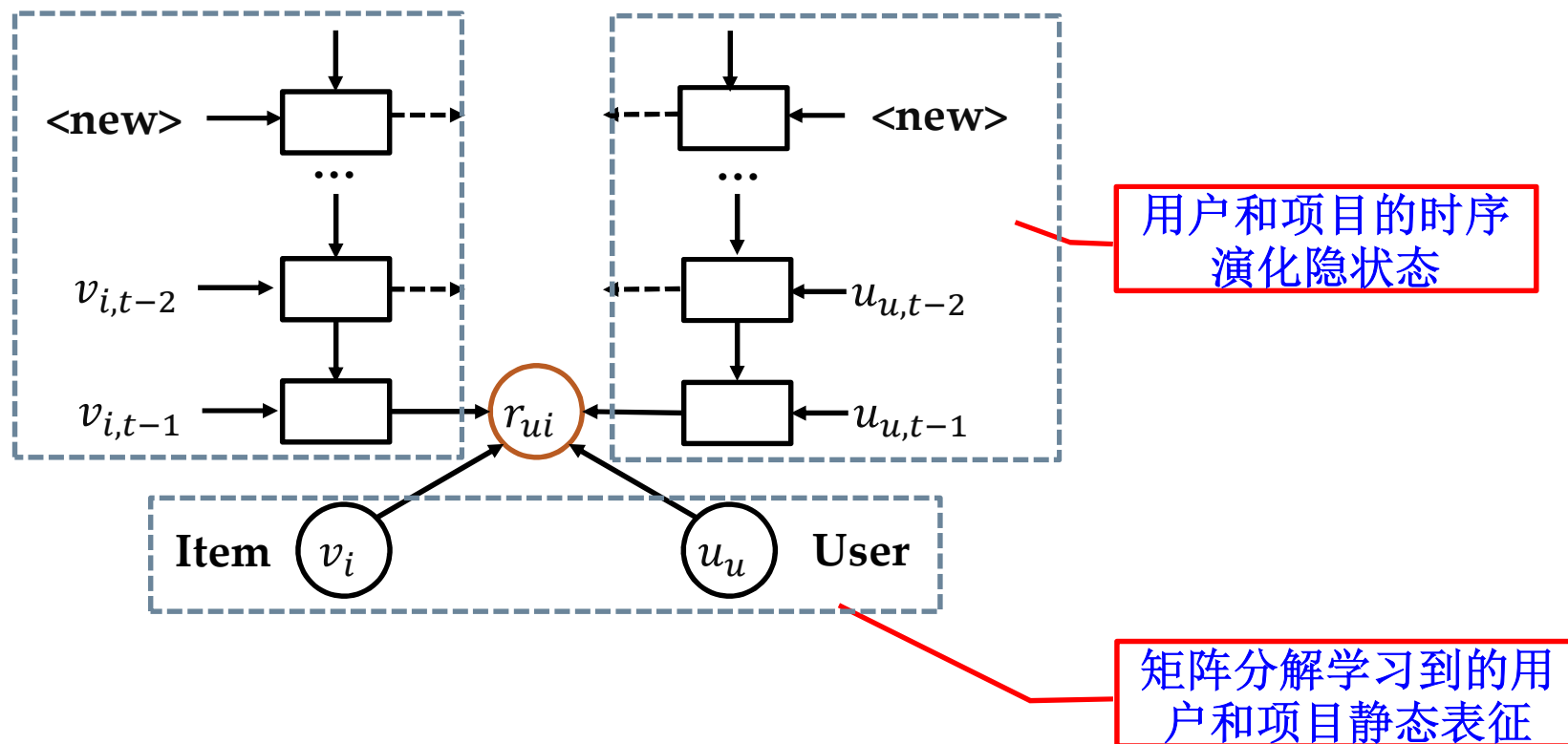
44

基于深度学习的推荐算法

循环神经网络

循环推荐网络 (RRN)

利用矩阵分解学习用户和项目的静态表征，
利用循环神经网络学习时序演化隐状态





推荐算法设计

45

□ 基于深度学习的推荐算法

□ 循环神经网络

■ 循环推荐网络 (RRN)

- 建模用户和项目的时序演化隐状态
- 利用循环神经网络学习用户和项目每个时刻的隐状态
- 利用矩阵分解学习用户和项目的静态表征

$$\hat{r}_{ui|t} = f(u_{ut}, v_{it}, u_u, v_i)$$

用户和项目的静态表征

用户和项目的时序隐状态



主要内容

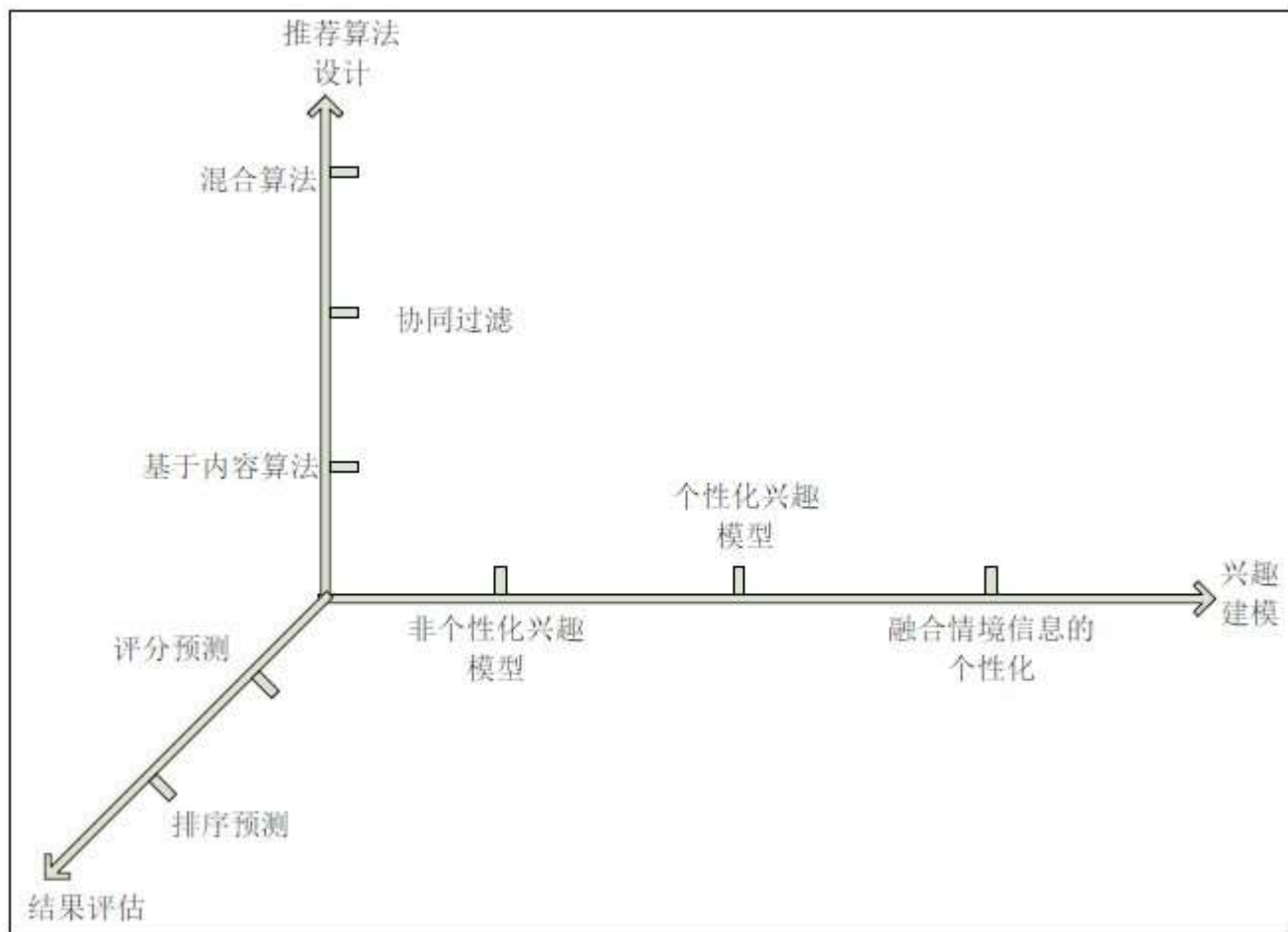
46

- 什么是推荐系统
 - 背景、定义、应用场景
- 推荐方法概述
 - 兴趣建模
 - 推荐算法设计
 - 推荐结果的评估
- 案例学习
 - 基于用户兴趣扩展的个性化推荐方法
 - 面向推荐系统的纠结心理挖掘
- 小结及未来的路
- 资料推荐



结果评估

47





结果评估

48

□ 以精度为目标

□ 评价预测精度

■ MAE: Mean Absolute Error

$$\text{MAE} = \sqrt{\frac{1}{|\mathcal{J}|} \sum_{(u,i) \in \mathcal{J}} |\hat{r}_{ui} - r_{ui}|}$$

■ RMSE: Root Mean Squared Error

$$\text{RMSE} = \sqrt{\frac{1}{|\mathcal{J}|} \sum_{(u,i) \in \mathcal{J}} (\hat{r}_{ui} - r_{ui})^2}$$

		users							
items	1	3		5		5	4		
			5		4		2	1	3
	2	4		1	2	3	4	3	5
		2	4		5		4		2
			4	3	4	2		2	5
	1	3		3		2		4	



结果评估

49

□ 以精度为目标

□ 排序预测精度

■ Precision、Recall、F-measure

■ Degree of Agreement(DOA)

$$DOA_{U_i} = \frac{\sum_{j \in E_{U_i}, k \in NW_{U_i}} check_order_{U_i}(I_j, I_k)}{|E_{U_i}| \times |NW_{U_i}|}$$

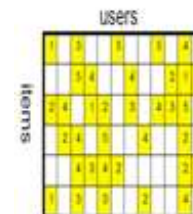
$$check_order_{U_i}(I_j, I_k) = \begin{cases} 1 & \text{if}(PR_{I_j} \geq PR_{I_k}) \\ 0 & \text{otherwise} \end{cases}$$

$$NW_{U_i} = I - (L_{U_i} \cup E_{U_i})$$

$$\text{Precision (p)} = \frac{TP}{TP + FP}$$

$$\text{Recall (r)} = \frac{TP}{TP + FN}$$

$$\text{F-measure (F}_1\text{)} = \frac{2rp}{r + p} = \frac{2 \times TP}{2 \times TP + FP + FN}$$



■ NDCG、MAP

■ Spearman's Correlation、Kendall's τ 、Jaccard Similarity

■ Click Ratio

■ ...



结果评估

50

□ 以精度为目标

□ 评分预测 VS 排序预测

■ 基于不同的假设

- 评分预测经常假定用户有机会评价所有的项目，所以使用评分代表用户偏好
- 排序预测算法一般认为用户喜欢那些他们已经消费过的项目

■ 不足之处

- 评分预测:用户不可能接触所有项目，而且只是中间步骤
- 排序预测：用户没消费的不一定代表不喜欢，消费的不一定是喜欢（正负反馈）

□ 其它问题

- 具体展示方式
- 结果是否可解释



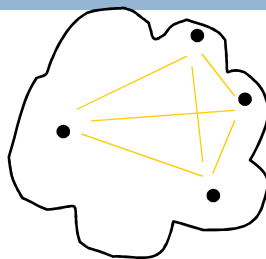
结果评估

51

□ 以多样性为目标

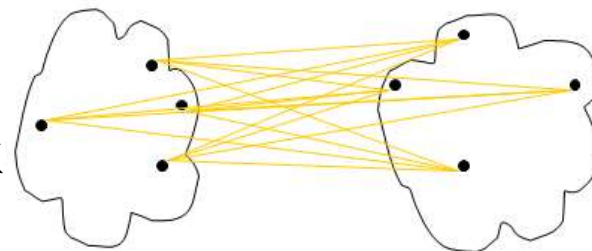
□ 列表内部多样性

■ 内部相似度



□ 列表外部多样性

■ 前后两次推荐的相似性



□ 兴趣覆盖率

$$IDiv_u@N = \frac{|\{categories \text{ in top-}N \text{ list of } u\} \cap \{categories \text{ liked by } u\}|}{|\{categories \text{ liked by } u\}|}$$

□ 新颖性、意外性

$$Novelty@k = \frac{1}{|U|} \sum_{u \in U} \sum_{i \in R_{uk}} \frac{\log_2(|U|/c_i)}{k}$$



结果评估

52

- 其他约束目标
 - 潜在消费量
 - 潜在收益
 - 效率、可扩展性
 - 算法鲁棒性
 - 用户满意度
 - A/B Test
 - 多个目标的优化
 - 。 。 。
- 实用推荐算法的三步验证
 - 离线实验验证其在离线指标上的表现
 - 用户调查确定用户满意度
 - 在线的A/B测试



主要内容

53

- 什么是推荐系统
 - 背景、定义、应用场景
- 推荐方法概述
 - 兴趣建模
 - 推荐算法设计
 - 推荐结果的评估
- 案例学习
 - 基于用户兴趣扩展的个性化推荐方法
 - 面向推荐系统的纠结心理挖掘
- 小结及未来的路
- 资料推荐



基于用户兴趣扩展的个性化推荐方法

54

研究动机

当前协同过滤推荐算法的不足

- 用户-项目 (User-Item) 表示模型--》无法捕获用户真正需求
- 仅考虑用户当前兴趣--》容易导致兴趣过拟合



(a)



(b)

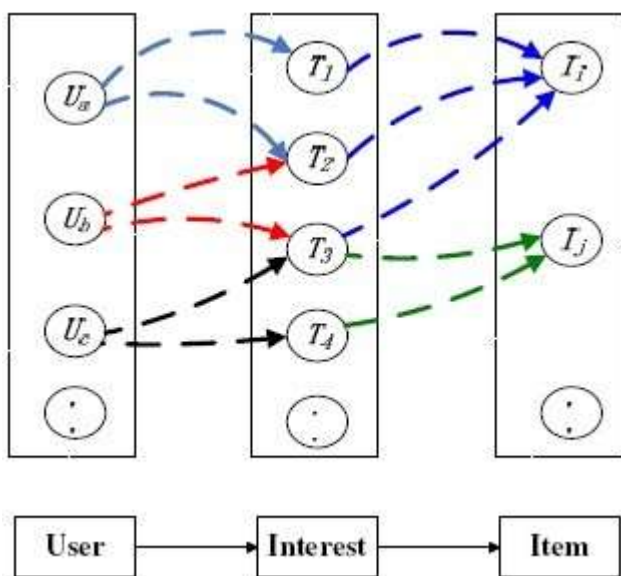


基于用户兴趣扩展的个性化推荐方法

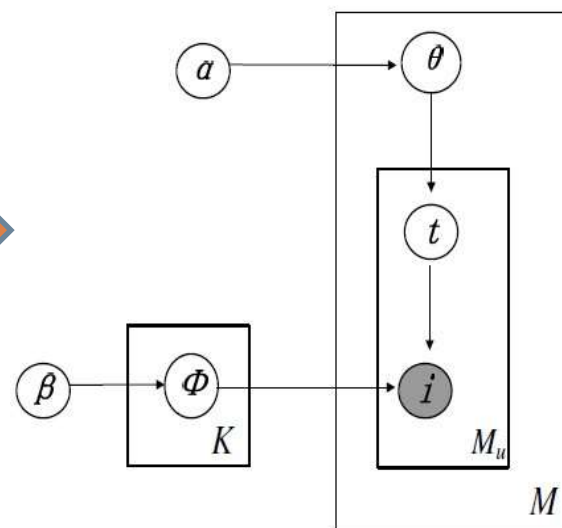
55

研究假设

- 用户的消费行为由一组隐式兴趣决定
 - 用户可以表示成兴趣的分布
 - 兴趣可以表示成项目的分布



兴趣抽取



LDA概率图模型



基于用户兴趣扩展的个性化推荐系统

56

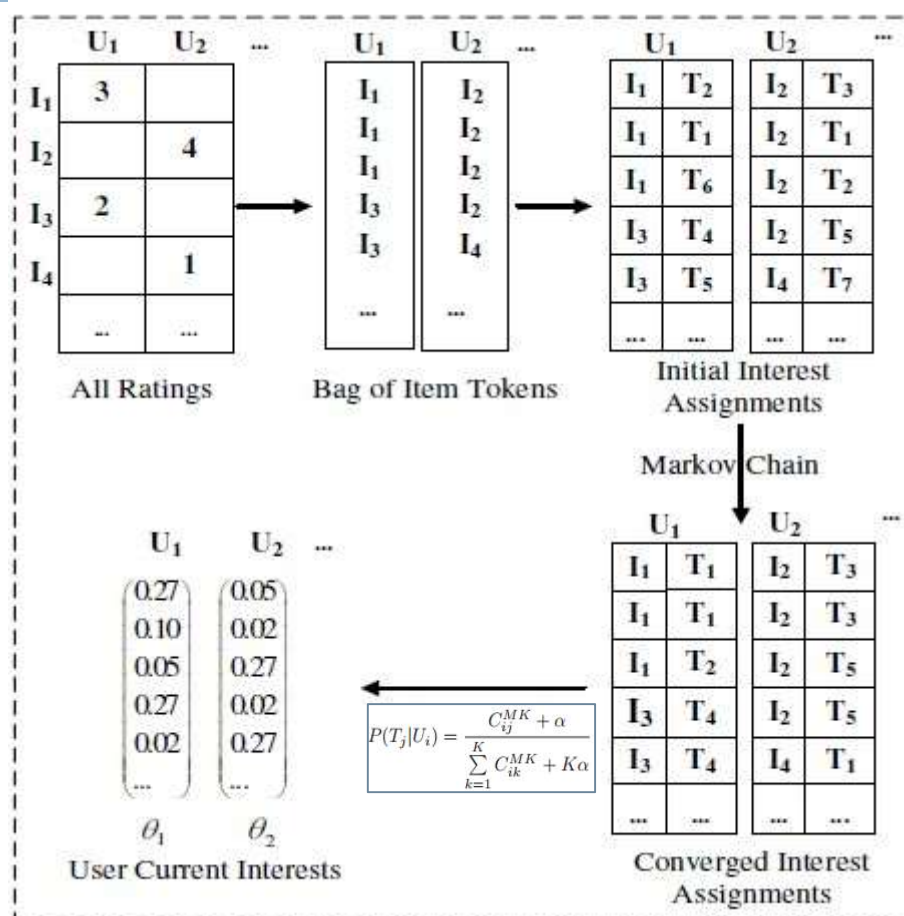
兴趣表示

LDA模型假设

- 用户(**U**) ~ 文档
- 项目(**I**) ~ 单词
- 兴趣(**T**) ~ 主题

抽取结果

- $\Theta_{ij} = P(T_j | U_i)$
- $\Phi_{ij} = P(I_i | T_j)$
- $\vec{\mathcal{G}}_i = P(T_i)$



基于Gibbs 抽样的LDA 推导过程实例



基于用户兴趣扩展的个性化推荐系统

57

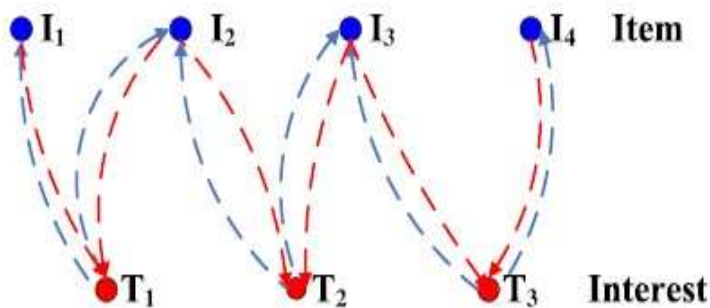
兴趣扩展

构建兴趣关联图

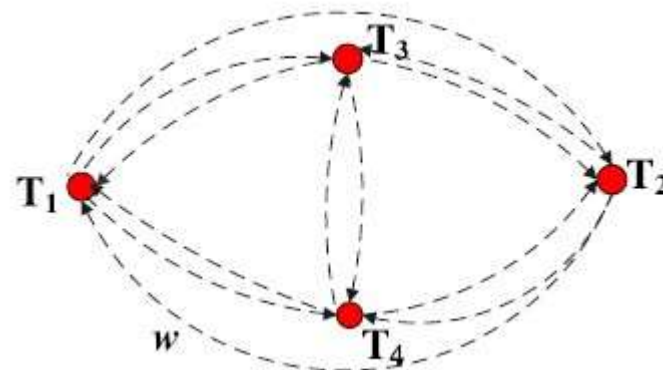
■ 由项目-兴趣二部图投影得到

用户兴趣扩展

■ 给定用户当前兴趣向量，在兴趣关联图



项目-兴趣二部图



兴趣关联图

$$\varphi_{ij} = P(T_j|I_i) = \frac{P(T_j, I_i)}{P(I_i)} = \frac{\phi_{ij}\vec{\vartheta}_j}{\sum_{k=1}^K \vec{\vartheta}_k \phi_{ik}}$$

$$\phi_{ij} = P(I_i|T_j) = \frac{C_{ij}^{NK} + \beta}{\sum_{n=1}^N C_{nj}^{NK} + N\beta}$$

}



$$\psi_{ij} = P(T_j|T_i) = \sum_{n=1}^N P(T_j|I_n)P(I_n|T_i) = \sum_{n=1}^N \varphi_{nj}\phi_{ni}$$



基于用户兴趣扩展的个性化推荐方法

58

- 生成推荐列表
 - 直接对项目进行排序
 - 计算 $P(I_j|U_i)$
 - 由评分预测生成推荐
 - 面向用户的协同过滤算法
 - 计算用户相似度，寻找最近邻居
 - 根据近邻爱好进行评分预测
- 在线推荐过程
 - 离线进行模型推导
 - LDA推导
 - 兴趣关联图
 - 实时更新用户兴趣

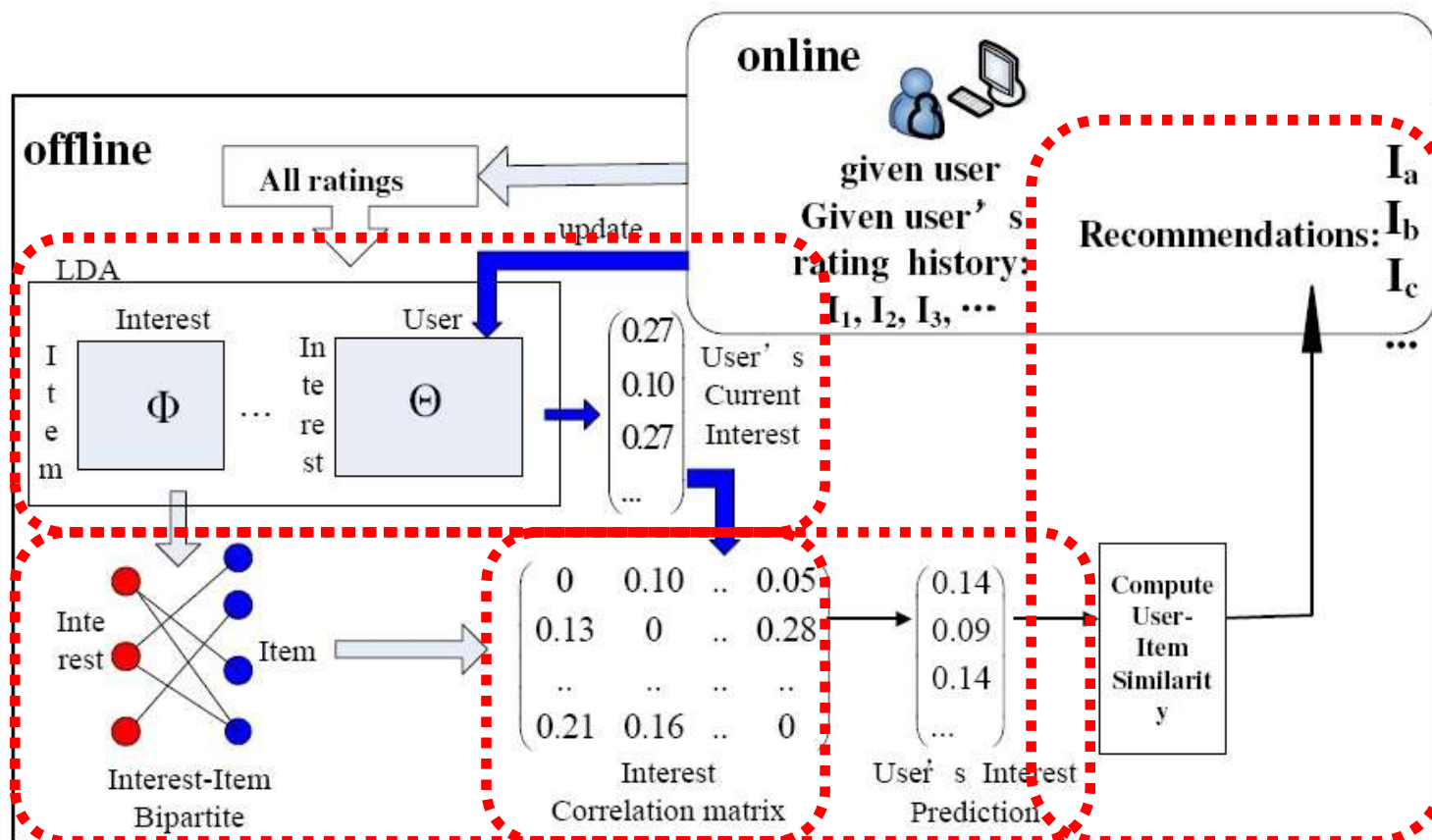


基于用户兴趣扩展的个性化推荐方法

59

推荐算法iExpand流程

第四步：构建
兴趣扩展矩阵
推荐列表





基于用户兴趣扩展的个性化推荐方法

60

□ 实验设置

三个数据集的统计信息

Data Set	Domain	#Users	#Items	#Records	Sparsity(%)
MovieLens	Movie	943	1,682	100,000	93.70
Book-Crossing	Book	996	1,696	91,084	94.61
Jester	Joke	2,000	100	36,596	81.70

□ 对比方法

□ 基于图模型的推荐算法

■ ItemRank、 L^+

□ 基于降维的推荐算法

■ LDA、SVD

□ 评价指标

□ DOA、Recall、TOP-K



基于用户兴趣扩展的个性化推荐方法

61

□ DOA、Recall结果

使用两个评价指标，iExpand在三个数据集的不同稀疏度划分上表现都为最优！

(a) MovieLens 数据集的结果比较 (左表: DOA in %, 右表: Recall in %)

Split.\Alg	SVD	ItemRank	L^+	LDA	iExpand	Split.\Alg	SVD	ItemRank	L^+	LDA	iExpand
10 - 90	52.50	76.69	66.85	75.76	79.04	10 - 90	10.52	24.90	13.46	21.41	28.19
20 - 80	67.99	84.40	84.02	83.47	85.78	20 - 80	13.57	31.94	22.26	33.09	35.51
30 - 70	75.33	86.35	86.96	87.28	87.99	30 - 70	14.99	32.61	25.68	35.14	36.10
40 - 60	79.88	87.39	88.77	88.68	89.06	40 - 60	17.45	31.63	28.05	34.95	35.24
50 - 50	83.93	88.21	89.59	89.23	89.86	50 - 50	17.82	30.46	27.19	33.02	33.53
60 - 40	85.56	88.72	90.34	89.99	90.62	60 - 40	17.53	28.18	25.24	30.46	31.50
70 - 30	84.83	88.76	90.60	90.74	91.20	70 - 30	14.19	24.95	23.32	27.36	28.48
80 - 20	85.24	89.00	90.63	90.82	91.31	80 - 20	11.36	20.76	19.34	23.00	23.46
90 - 10	86.91	88.73	90.60	91.11	91.65	90 - 10	6.94	14.54	12.88	15.69	16.16

(b) Book-Crossing 数据集的结果比较 (左表: DOA in %, 右表: Recall in %)

Split.\Alg	SVD	ItemRank	L^+	LDA	iExpand	Split.\Alg	SVD	ItemRank	L^+	LDA	iExpand
10 - 90	46.35	57.73	52.83	57.27	60.17	10 - 90	5.05	9.21	5.66	7.83	12.77
20 - 80	52.71	62.14	60.36	59.89	62.67	20 - 80	5.70	12.24	6.46	10.13	13.38
30 - 70	58.63	64.85	62.84	64.71	65.68	30 - 70	6.60	13.01	6.54	12.36	13.89
40 - 60	62.47	66.99	65.61	67.16	68.06	40 - 60	6.87	12.50	7.07	12.52	13.58
50 - 50	65.66	67.61	67.29	68.29	69.12	50 - 50	7.07	11.50	7.62	12.08	12.77
60 - 40	67.38	68.15	69.15	69.88	70.43	60 - 40	6.99	9.81	7.85	11.65	12.17
70 - 30	69.07	68.69	70.77	70.54	71.61	70 - 30	5.93	8.29	7.67	10.57	11.03
80 - 20	69.88	69.57	71.34	71.92	72.58	80 - 20	4.54	6.41	6.51	8.18	8.89
90 - 10	71.45	69.64	72.70	72.43	73.37	90 - 10	2.48	4.16	4.62	5.56	6.27

(c) Jester 数据集的结果比较 (左表: DOA in %, 右表: Recall in %)

Split.\Alg	SVD	ItemRank	L^+	LDA	iExpand	Split.\Alg	SVD	ItemRank	L^+	LDA	iExpand
10 - 90	47.66	84.19	56.23	85.98	86.18	10 - 90	17.72	55.43	20.71	60.25	60.61
20 - 80	39.84	87.16	60.93	87.12	87.32	20 - 80	9.86	58.47	13.58	58.04	58.96
30 - 70	36.37	86.75	61.25	86.49	86.95	30 - 70	8.00	56.70	14.07	56.79	56.83
40 - 60	35.11	87.19	68.53	86.78	87.35	40 - 60	6.48	54.21	14.78	52.97	54.78
50 - 50	37.28	87.25	75.56	86.79	87.31	50 - 50	6.29	50.21	18.63	50.06	50.33
60 - 40	35.81	87.70	78.93	87.29	87.71	60 - 40	4.36	46.42	17.76	45.33	46.66
70 - 30	36.60	87.70	80.90	87.09	87.78	70 - 30	3.31	41.36	14.58	40.84	41.97
80 - 20	41.17	87.54	81.59	87.14	87.65	80 - 20	2.31	35.35	10.02	33.62	36.29
90 - 10	48.26	88.46	82.98	88.22	88.51	90 - 10	1.57	23.82	4.80	23.49	24.59

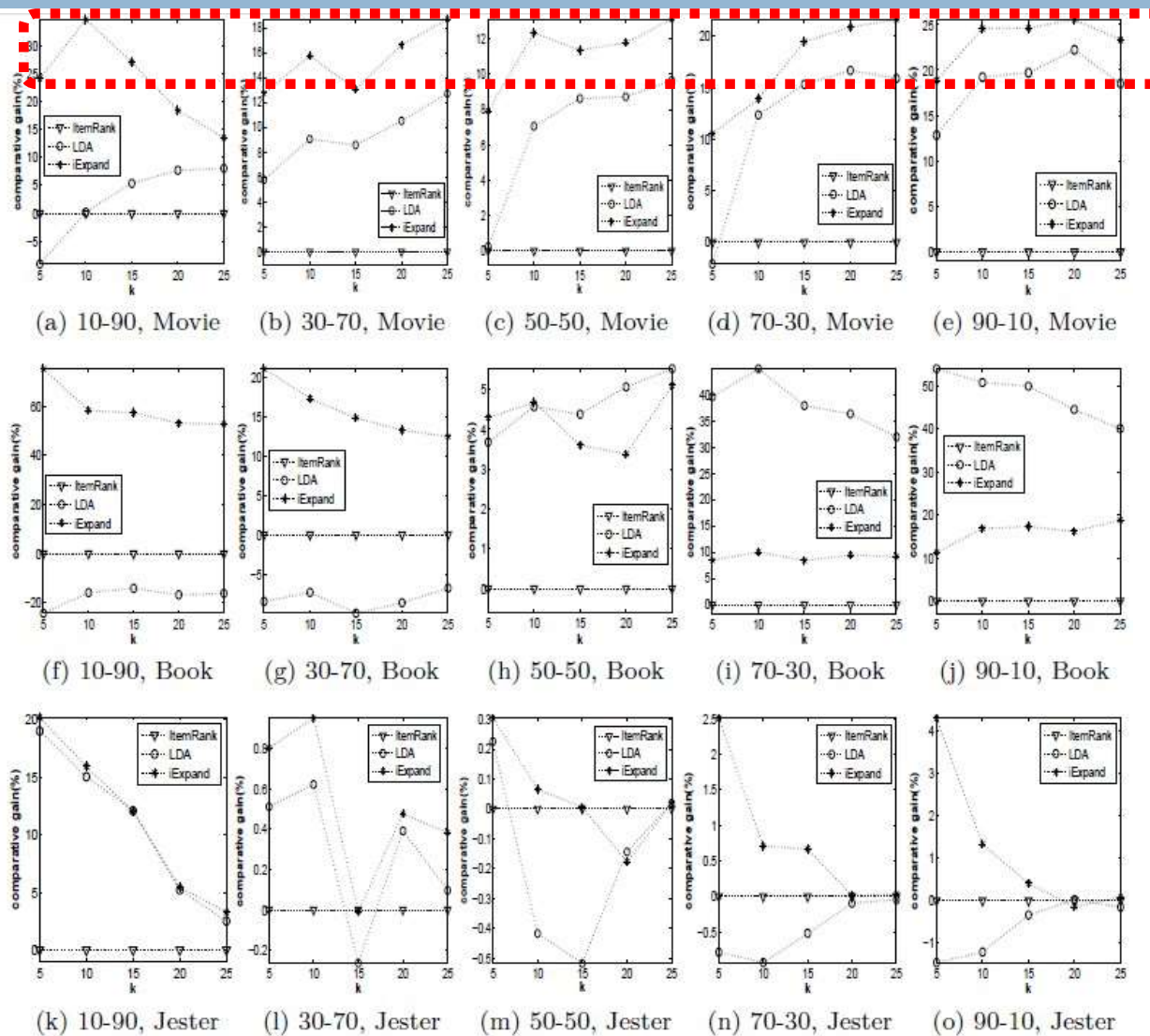


基于用户兴趣扩展的个性化推荐方法

62

Top-K结果

使用Top-K评价指标，iExpand
在三个数据集的
绝大部分划分上
表现都为最优！





基于用户兴趣扩展的个性化推荐方法

63

实例学习

兴趣理解

三个兴趣中概率最大的5部电影

Latent interests	Interest 1	Interest 2	Interest 3
movies	Back to the Future Return of the Jedi Raiders of the Lost Ark Star Wars The Empire Strike Back	Secrets & Lies Il Postino: The Postman My Life as a Dog Sunset Blvd A Room with a View	Star Wars The English Patient The Silence of the Lambs Godfather Pulp Fiction

推荐实例

U_{140} 及相应的推荐结果

Training set	Test set	Top Recommendations	
		Without Interests Expansion	Interests Expansion
The Devil's Own Contact Scream L.A. Confidential Murder at 1600 Crash Kiss the Girls U Turn	The English Patient Evita Liar Liar In & Out Ulee's Gold Fly Away Home Everyone Says I Love You Mother	Air Force One Titanic <i>LiarLiar</i> <i>TheEnglishPatient</i> Conspiracy Theory The Full Monty The Game Seven Years in Tibet	Titanic <i>LiarLiar</i> <i>TheEnglishPatient</i> The Game The Full Monty <i>In&Out</i> Air Force One <i>Evita</i>



基于用户兴趣扩展的个性化推荐方法

64

□ 小结 [IEEE Trans. SMC-B]

□ 提出iExpand协同过滤推荐算法

- 采用用户-兴趣-项目三层推荐模型
- 利用随机游走扩展用户当前兴趣
 - 可应对用户兴趣过拟合问题
- 实验表明iExpand可以取得更优的排序推荐效果



主要内容

65

- 什么是推荐系统
 - 背景、定义、应用场景
- 推荐方法概述
 - 兴趣建模
 - 推荐算法设计
 - 推荐结果的评估
- 案例学习
 - 基于用户兴趣扩展的个性化推荐方法
 - 面向推荐系统的纠结心理挖掘
- 小结及未来的路
- 资料推荐



面向推荐系统的纠结心理挖掘

66

研究背景

- 用户的“纠结”无处不在
 - 体现在消费（如点击记录）过程中
- 建模纠结心理可以提供更好的服务
 - 量化产品（商家）竞争关系
 - 引导用户的消费、增加消费成功率

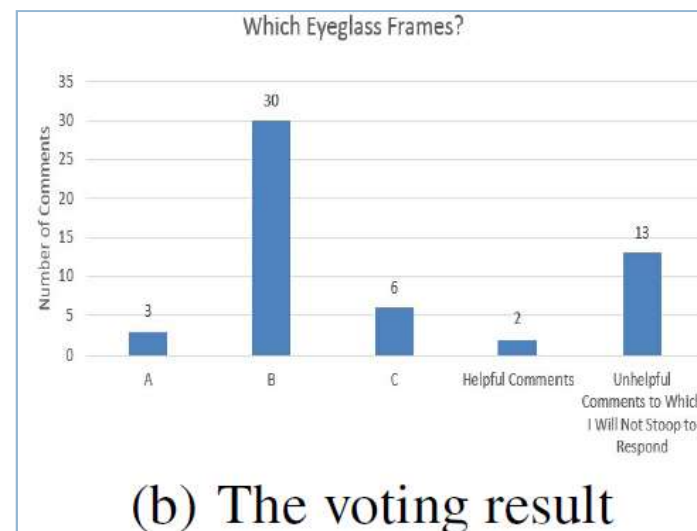


Figure 1: An example of Indecisiveness from Dr. John Krumm.



面向推荐系统的纠结心理挖掘

67

研究挑战

如何定义和
量化纠结度

- 心理学专家给出定义
- 用户完成调查问卷
- **行为数据驱动的定义？**

如何发现纠
结的原因

- **哪些因素导致了纠结？**
- 用户自身原因？
- 相似竞争产品的原因？

如何使用纠
结度

- 如何帮助商家？
- 如何帮助消费者？

Table 1. Items for the indecisiveness scale.

1. I find it easy to make decisions.
2. It is hard for me to come to a decision.
3. I don't know how to make decisions.
4. I know which steps to take when making a decision.
5. I would characterize myself as an indecisive person.
6. I don't hesitate much when I have to make a decision.
7. While making a decision, I feel certain.
8. While making a decision, I feel uncertain.
9. It takes a long time to weigh the pros and cons before making a decision.
10. I make decisions quickly.
11. I delay deciding.
12. I don't postpone making decisions to a later date.
13. I try to avoid making a decision.
14. I don't avoid situations where decisions have to be made.
15. I tend to leave decisions to someone else.
16. I cut the knot myself in a decision instead of leaving the decision to others.
17. Once I have taken a decision, I stick to that decision.
18. I often reconsider my decision.
19. Once I have made a decision, I stop worrying about it.
20. After making a decision, I can't get it out of my mind.
21. After I have decided something, I believe I took the wrong decision.
22. After making a decision, I don't regret the decision.



面向推荐系统的纠结心理挖掘

68

□ 数据和平台

- Tmall大规模真实数据集
- 分布式数据处理服务平台

天池
大数据科研平台

[首页](#)[大赛介绍](#)[寻找任务](#)[Geek攻略](#)[开放式科研](#)

[中文](#)[English](#)

[开放式科研](#) > [Rec-Tmall](#)

Rec-Tmall

数据集owner: 阿里巴巴集团天猫事业部 发布时间: 2014-11-15

数据集介绍

天猫（TMALL）是中国最大的B2C购物平台。Rec-Tmall是由天猫提供的一组推荐系统相关的数据集，全称为Rec-Tmall。

数据表编号	数据表名称	数据表字段
Tianchi_2014001_Rec_Tmall_Product	天猫商品基本信息表	商品ID、商品标题、商品主图url、商品所属类目名称、商品所属品牌ID、商品
Tianchi_2014002_Rec_Tmall_Log	天猫用户行为表	商品ID、用户ID、交互行为类型、行为发生时间
Tianchi_2014003_Rec_Tmall_Review	天猫用户评价表	商品ID、用户ID、评价文字、评价图片url、评价时间



面向推荐系统的纠结心理挖掘

69

□ 数据预处理

Table 1: A toy example of the customer behavioral records.

UserId	ItemId	CategoryId	Action	Timestamp
U_1	a	C_1	Click	2014-07-08 20:05:20
U_1	b	C_1	Click	2014-07-08 20:06:40
...
U_1	a	C_1	Cart	2014-07-08 20:13:55
U_1	b	C_1	Collect	2014-07-08 20:14:20
U_1	b	C_1	Buy	2014-07-08 20:14:38
U_2	f	C_2	Click	2014-07-09 10:21:13
U_2	f	C_2	Buy	2014-07-09 10:21:20

格式转换

Table 2: Examples of the customer behavior sessions.

SessionId	UserId	CategoryId	Item Sequence	Item Bundle
S_1	U_1	C_1	a,b,b,a,b,b,a,a,b,b	{a,b}
S_2	U_1	C_1	a,c,a,c,e,b,e,e,c,e	{a,b,c,e}
S_3	U_1	C_2	f,g,f	{f,g}
S_4	U_2	C_1	a,b,c,a	{a,b,c}
S_5	U_2	C_3	d,d	{d}

一个session里的
不同item组成的
集合



面向推荐系统的纠结心理挖掘

70

- **Step 1:**定义观察到的“纠结”
 - 对每个用户的一个session进行定义

DEFINITION 2 (OBSERVED INDECISIVENESS). *Given a customer U_i and one of her behavior session S_j where the corresponding (largest) item bundle is B_k , we define the observed indecisiveness of this session as*

$$D_{ij}^S = F(\text{Length}(S_j), \text{Entropy}(S_j), \text{Trans}(S_j)). \quad (1)$$

依据:

- 点击数目
- 相似产品的均衡度 (熵)
- 产品间的转移数

$$F(\text{Length}(S_j), \text{Entropy}(S_j), \text{Trans}(S_j)) = \text{Length}(S_j) * \text{Entropy}(S_j) * \text{Trans}(S_j),$$

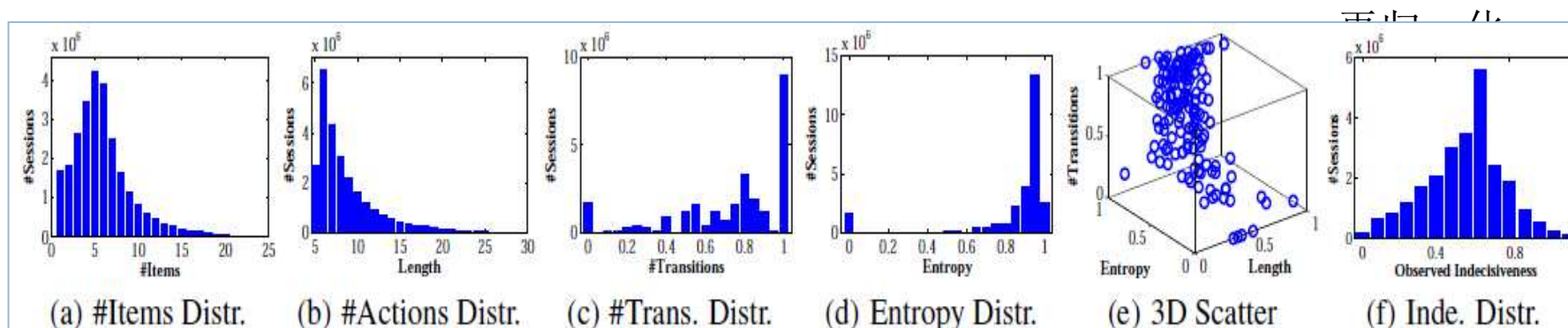


Figure 4: Some statistics from the observed records.



面向推荐系统的纠结心理挖掘

71

Step 2: 发现纠结背后的原因

- 有的用户容易纠结
- 有些产品的组合(Bundle)容易让用户纠结

} 隐含的纠结指数

如何识别“纠结指数”？

■ 建立矩阵分解模型

■ 观察值

- 观察到的“纠结” D_{ij}^S

■ 隐含变量

- 用户纠结指数 Y_i
- 产品组合的纠结指数 Z_j^S

■ 优化目标

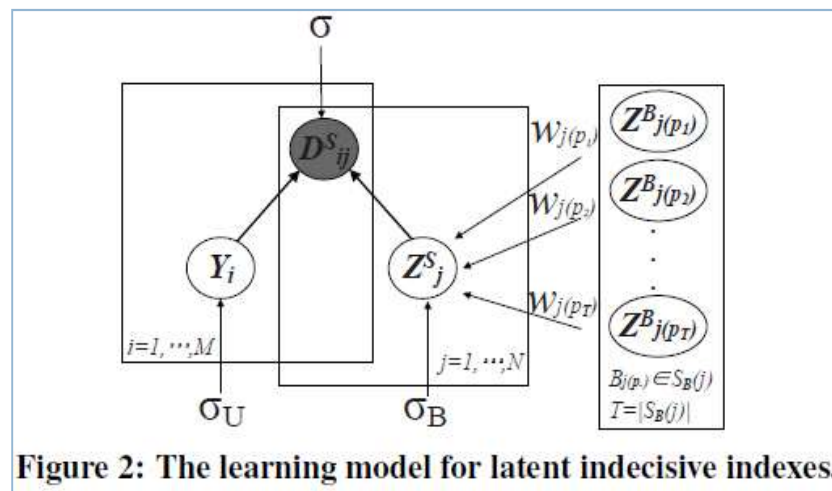


Figure 2: The learning model for latent indecisive indexes.

$$E = \frac{1}{2} \sum_{i=1}^M \sum_{j=1}^N H_{ij} [D_{ij}^S - g(Y_i (\sum_{B_k \in S_B(j)} w_{jk} Z_k^B))]^2 + \frac{\lambda_U}{2} \sum_{i=1}^M Y_i^2 + \frac{\lambda_B}{2} \sum_{k=1}^P Z_k^{B^2},$$



面向推荐系统的纠结心理挖掘

72

Step 3: 如何使用纠结度

客户分群

需要用户的性别、年龄等属性数据

发现竞争产品（商家）

根据产品组合的纠结指数(Z^B_j)



Figure 10: Three pairs of typical items that are competitive to each other.



Figure 11: The competitors of one skirt and one couch, respectively.



面向推荐系统的纠结心理挖掘

73

□ Step 3: 如何使用纠结度

□ 提高推荐准确率

- 用户纠结于多个产品时，越让他纠结的产品，他最终从中购买的比率越高

Assumption Given a customer and one of her behavior session consisting of several item bundles, the higher indecisiveness of one item bundle, the higher probability that the finally consumed item locates in this bundle.

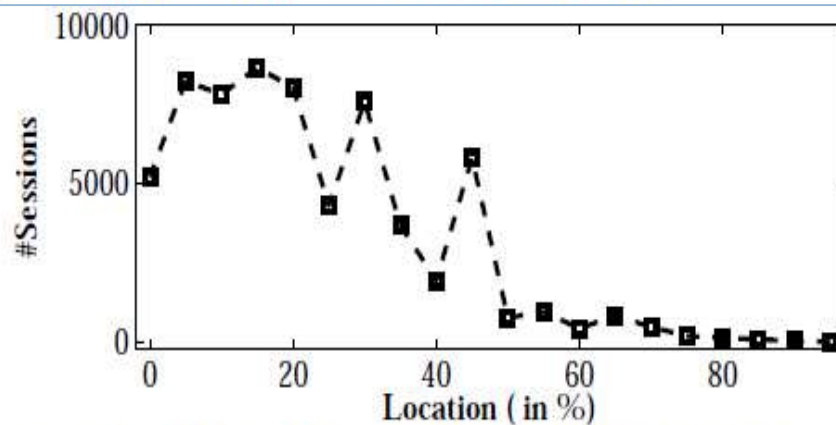


Figure 8: Location of the consumed items in buying sessions.



面向推荐系统的纠结心理挖掘

74

Step 3: 如何使用纠结度

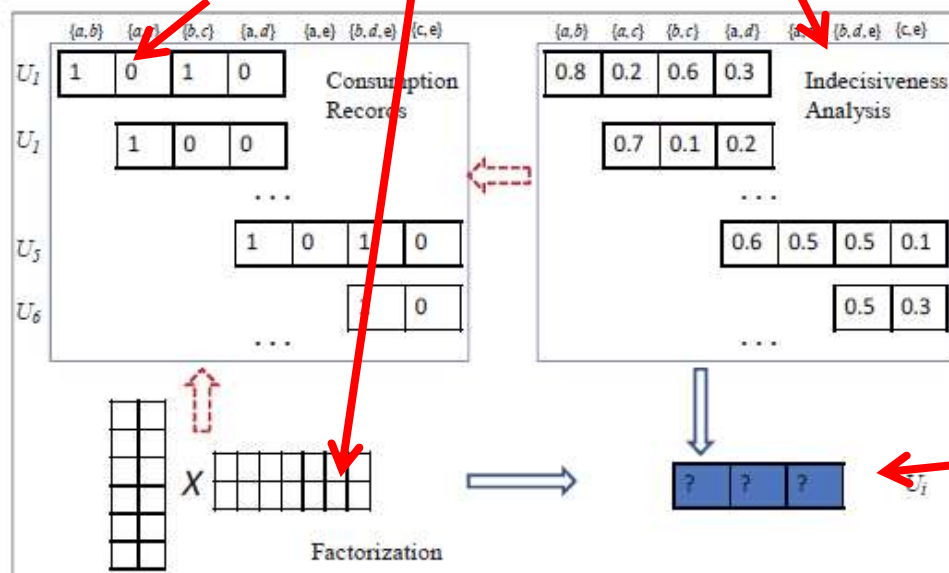
提高推荐准确率

- 用户纠结于多个产品时，越让他纠结的产品，他最终从中购买的比率越高
- 可以利用纠结度做为额外特征去选择产品（产品组合）进行实时推荐

用户最终的购买结果

$$\hat{r}_{ui}^S = (Q'_u \cdot V_i) + \alpha_u (Y'_u \cdot w_{si} \cdot Z_{si}^B),$$

由纠结指数计算得到的纠结度





面向推荐系统的纠结心理挖掘

75

□ 实验验证

□ 实验数据

- Tmall网站2013年4月至9月全部的用户行为记录

□ 平台介绍

- 阿里云研发的分布式开放数据处理服务平台一天池（MapReduce）

□ 预处理

- 例如，去除冷启动用户和产品等
- 80%训练--20%测试

TABLE IV

THE STATISTICS OF THE DATASET BEFORE AND AFTER PRUNING.

	Original Data	Pruned Data
#Customers	9,774,184	1,998,112
#Items	8,133,507	50,700
#Sessions	234,496,841	4,182,243
#Actions	1,333,729,303	44,000,123
#Item Bundles	\	141,951



面向推荐系统的纠结心理挖掘

76

实验结果

“纠结”定义的有效性

- 缺乏Ground Truth，所以使用User Study（在线调查表）验证
 - 让志愿者选择哪一个session看上去更纠结

1、The following are two behavioral sessions, and please choose the session that seems to be more indecisive for you. * (必填, 单选)

- ☐ a,b,b,c,d,e,f,g,h,h,h
- ☐ a,b,c,d,d

- 收集到106个志愿者的共计954次选择
- 以志愿者的选择为Ground Truth，计算各个方法与它的相似度

TABLE V
USER STUDY RESULTS.

Metrics		<i>Length(·)</i>	<i>Trans(·)</i>	<i>Entropy(·)</i>	<i>Indecisiveness</i>
Jaccard	Mean	0.736	0.372	0.063	0.849
Similarity	z	4.091	20.791	33.095	\
Euclidean	Mean	1.100	2.033	2.829	0.596
Distance	z	5.603	19.035	28.739	\

本文的定义
与人工结果
最相似



面向推荐系统的纠结心理挖掘

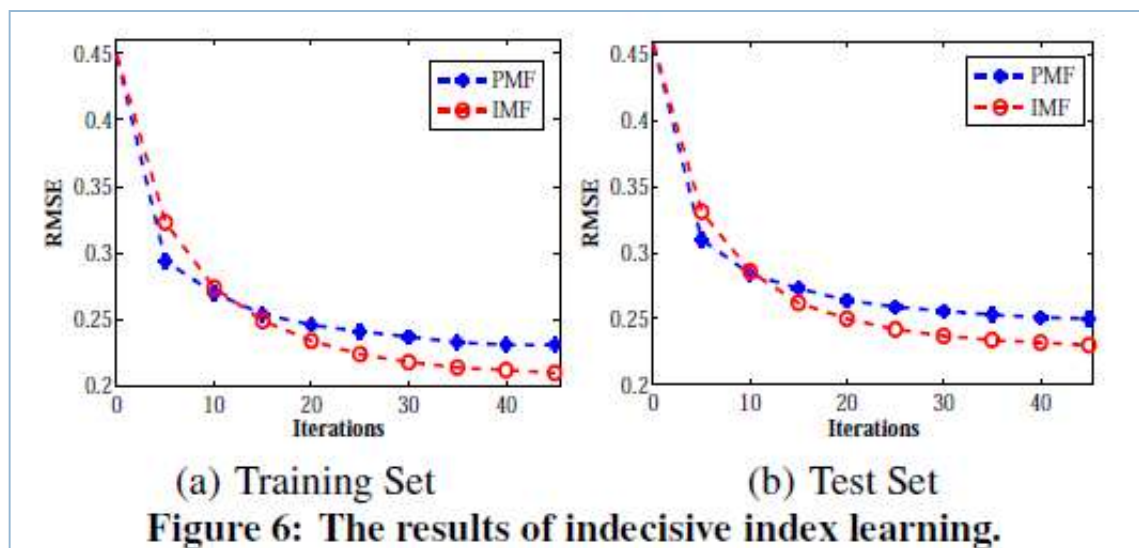
77

实验结果

“纠结指数”矩阵分解模型（IMF）有效性

对比方法：PMF

评价指标：RMSE



比PMF的
预测效果提
升约10%



面向推荐系统的纠结心理挖掘

78

实验结果

- 用户最终购买的产品子集（Bundle）推荐（预测）的准确性

TABLE VI

THE RESULTS OF ITEM BUNDLE RECOMMENDATION.

Method \ Metric	REC	FRE	KNN	IKNN	IND	BPR	IBR
AUC	0.757	0.802	0.521	0.615	0.818	0.826	0.893
Top-1	0.762	0.869	0.288	0.406	0.895	0.647	0.924
RunTime(Sec.)	100	102	130	135	102	110	113

根据直观的
统计结果

K近邻相关
的推荐方法

仅考虑纠结
或学习排序

本文提出的方法取得了最高的推荐精度，而且平均把用户的候选购买集合缩小了60%

- Qi Liu, Xianyu Zeng, Chuanren Liu, Henghu Zhu, Enhong Chen, Hui Xiong, Xing Xie, Mining Indecisiveness in Customer Behaviors, ICDM 2015, Accepted as a regular paper (Acceptance Rate 8.4%).



面向推荐系统的纠结心理挖掘

79

□ 小结

定义和量化纠结度

- 行为数据驱动的定义，考虑用户表现出的三种心理和行为特征

发现纠结的原因

- 通过拟和观察到的纠结值，识别用户和产品所隐含的纠结指数（或称其为纠结偏好）

使用纠结度

- 将纠结度做为额外特征去选择产品（产品集合），以进行针对用户最终购买预测的实时推荐

□ 未来工作

更合理的纠结定义方法

寻找更多的应用场景

进行更完备的实验验证

...



主要内容

80

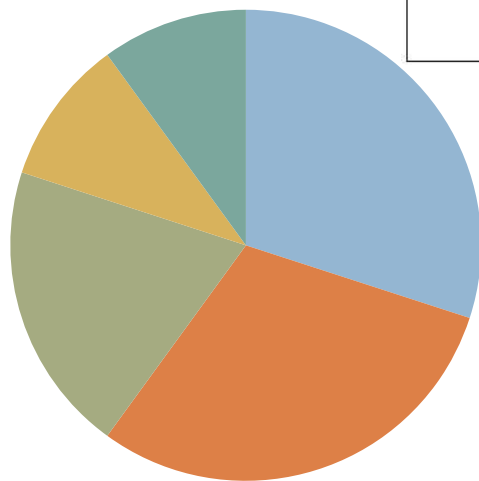
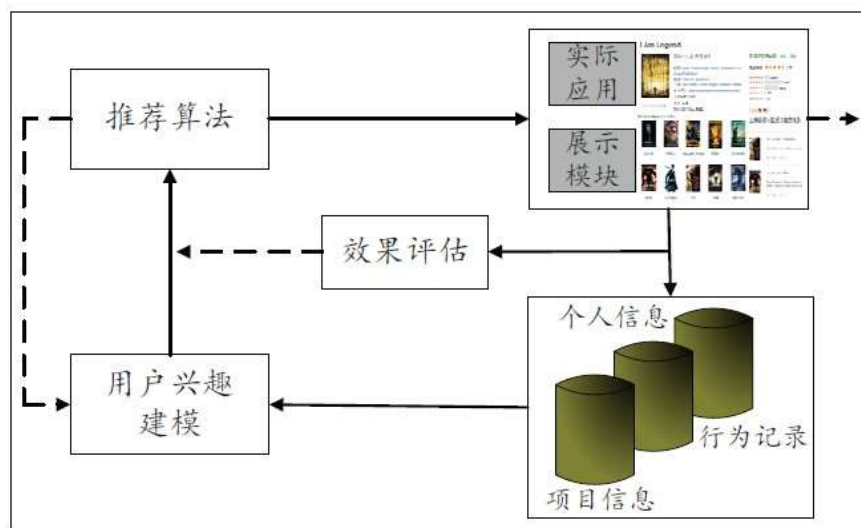
- 什么是推荐系统
 - 背景、定义、应用场景
- 推荐方法概述
 - 兴趣建模
 - 推荐算法设计
 - 推荐结果的评估
- 案例学习
 - 基于用户兴趣扩展的个性化推荐方法
 - 面向推荐系统的纠结心理挖掘
- 小结及未来的路
- 资料推荐



推荐方法小结

81

□ 推荐系统是个基于数据挖掘方法的系统



UI/UE	30%
Data	30%
Domain Knowledge	20%
Algorithm	10%
Others	10%

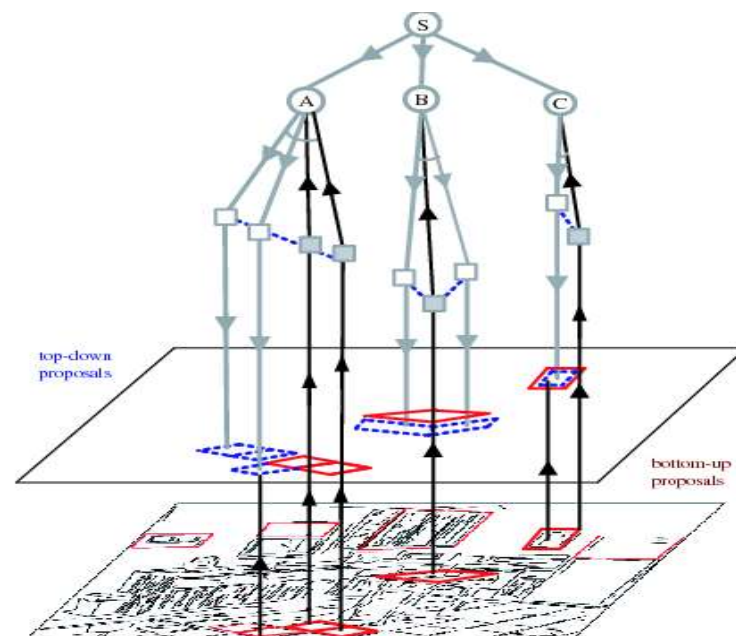


推荐方法小结

82

□ 设计推荐算法不能盲目

- 最终目标不同，设计的算法也不同
 - 评分预测—概率矩阵分解PMF
 - 排序预测—学习排序模型BPR
- 推荐具有很强的领域特性
 - 面向教育、旅游、社交好友的推荐
 - 融合领域知识、特征
 - 自底向上的设计推荐算法
 - 从问题出发
- 特征有时比算法更重要
- 集成的推荐方法一般比较稳定





未来的路

83

□ 开拓新的应用领域

□ 学习和结合领域知识

- 医疗推荐
- 投资推荐
- 房产推荐
- 在线教育
- 。 。 。



未来的路

84

在线教育

致力于学生能力提高的试题推荐、协作学习推荐策略



1. 函数 $f(x) = \frac{1}{x-2}$ 的反函数为 $f^{-1}(x) = \underline{\hspace{2cm}}$ 。

答案: $y = \frac{1}{x} + 2$

知识点: 函数

标签: 定义法

解析: 本题主要考查反函数的意义。
 $y = \frac{1}{x-2}$ 的反函数是 $x = \frac{1}{y-2}$, 整理得 $y = \frac{1}{x} + 2$ 。
故本题正确答案为 $y = \frac{1}{x} + 2$ 。

相似题目

137349. (2011上海八校联考) 函数 $y = 2^x + 1$ 的反函数为 $\underline{\hspace{2cm}}$ 。

答案: $y = \log_2(x-1)$

知识点: 函数

解析: 本题主要考查求反函数。
由函数解析式可得: $y-1 = 2^x$, $x = \log_2(y-1)$, 故反函数为 $y = \log_2(x-1)$ 。

875. (2011大纲卷) 函数 $y = 2\sqrt{x} (x \geq 0)$ 的反函数为 ()。

A. $y = \frac{x^2}{4} (x \in \mathbb{R})$
B. $y = \frac{x^2}{4} (x \geq 0)$
C. $y = 4x^2 (x \in \mathbb{R})$
D. $y = 4x^2 (x \geq 0)$

答案: B

知识点: 函数

易错试题个性化推荐



未来的路

85

□ 完善推荐算法设计

□ 借鉴更多的机器学习方法

- Reinforcement learning、Online learning

□ 利用多源数据指导推荐算法设计



□ 深入理解各角色之间的约束关系

- 如何保证项目的“机会公平性”
- 如何保护用户的隐私、安全
- 如何对商家进行“推荐”？
 - 推荐最能盈利的项目、用户
 - 推荐生产、营销的策略和库存、物流安排





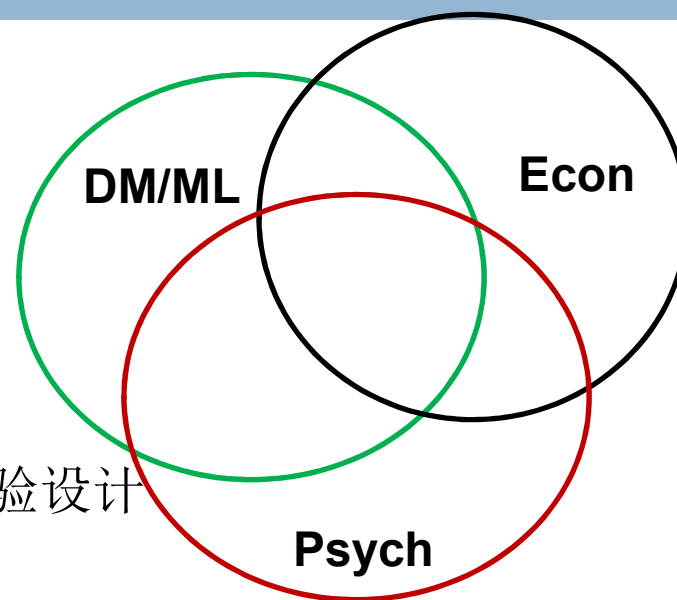
未来的路

86

□ 多学科知识的必然交叉

□ The “Sweet” Spot

- 挖掘数据背后隐含的用户心理因素
 - 纠结?
 - 猎奇?
- 使用经济学、社会学原理指导算法与实验设计
 - 增益递减性质
 - 投资组合原理
- 从消费行为间的关联关系到因果关系的挖掘转变
 - 兴趣引导
 - 增加心理暗示? “Hot”





未来的路

87

- Opportunities and Challenges Facing Recommender Systems: Where Can We Go from Here? ---Alex Tuzhilin





主要内容

88

- 什么是推荐系统
 - 背景、定义、应用场景
- 推荐方法概述
 - 兴趣建模
 - 推荐算法设计
 - 推荐结果的评估
- 案例学习
 - 基于用户兴趣扩展的个性化推荐方法
 - 面向推荐系统的纠结心理挖掘
- 小结及未来的路
- 资料推荐



推荐资料

89

□ Conference

- ACM SIGKDD, IEEE ICDM, ACM RecSys, SIGIR, WWW

□ Journals

- IEEE TKDE, KAIS, DMKD、ACM TKDD, ACM TOIS...

□ Persons

- Very diversify

□ Data sets

- Yelp, TripAdvisor, MovieLens, Book-crossing, Flixster, Douban、数据堂.....

□ Competitions

- Netflix
- Kaggle <http://www.kaggle.com/>
- KDD Cup 2011—Yahoo music, 2012—Weibo Recommendation
- 阿里巴巴-天池

□ Software Packages

- libFM、MyMediaLite、Duine、EasyRec



推荐资料

90

□ 1. 中文综述

- a) 个性化推荐系统的研究进展
- b) 个性化推荐系统评价方法综述
- c) 《推荐系统实践》
- d) 《推荐系统》

2. 英文综述

- a) 2004ACM TOIS-Evaluating collaborative filtering recommender systems
- b) 2004ACM TOIS-Introduction to Recommender Systems - Algorithms and evaluation
- c) 2005IEEE TKDE -Toward the next generation of recommender systems - A survey of the state-of-the-art and possible extensions
- d) Recommender Systems Handbook
- e) A survey of Context-aware Mobile Recommendations
- f) Deep Learning based Recommender System: A Survey and New Perspectives

3. 快速实践

- a) 2004ACM TOIS-- Item-based top-N recommendation algorithms
- b) 2007PRE-- Bipartite network projection and personal recommendation
- c) 2007 NIPS --Probabilistic Matrix Factorization
- d) Matrix Factorization Techniques for Recommender systems
- e) 2009 UAI- BPR: Bayesian personalized ranking from implicit feedback



推荐资料

91

- The Recommender Problem Revisited
---*KDD 2014 Tutorial* Netflix

Index
1. The Recommender Problem
2. Traditional Recommendation Methods
2.1. Collaborative Filtering
2.2. Content-based Recommendations
2.3. Hybrid Approaches
3. Beyond Traditional Methods
3.1. Learning to Rank
3.2. Similarity
3.3. Deep Learning
3.4. Social Recommendations
3.5. Page Optimization
3.6. Tensor Factorization and Factorization Machines
3.7. MAB Explore/Exploit
4. References

- Collaborative Filtering and other approaches
---*MLSS 2014* Xavier Amatriain

Index
1. Introduction: What is a Recommender System
2. "Traditional" Methods
2.1. Collaborative Filtering
2.2. Content-based Recommendations
3. Novel Methods
3.1. Learning to Rank
3.2. Context-aware Recommendations
3.2.1. Tensor Factorization
3.2.2. Factorization Machines
3.3. Deep Learning
3.4. Similarity
3.5. Social Recommendations
4. Hybrid Approaches
5. A practical example: Netflix
6. Conclusions
7. References

<http://www.kdd.org/kdd2014/tutorials/KDD%20-%20The%20Recommender%20Problem%20Revisited.pdf>



推荐资料

92

- **New Directions in Recommender Systems** □
---WSDM 2015 Jure Leskovec



- **Foundations of Web Personalization and Recommender Systems**

---IJCAI 2015 Jill Freyne & Shlomo Berkovsky

Outline

Part 1: Introduction, Information Overload, User Modelling

Part 2: Personalization for Information Filtering, Information Access & Content Delivery

Part 3: Recommender Systems

<http://ijcai-15.org/downloads/tutorials/T11-WebPersonalization.pdf>



部分参考文献

- Chen, Shuo, and Thorsten Joachims. "Modeling intransitivity in matchup and comparison data." Proceedings of the ninth acm international conference on web search and data mining. ACM, 2016.
- Chen, S., & Joachims, T. (2016, August). Predicting Matchups and Preferences in Context. In KDD
- Ma, Hao, et al. "Sorec: social recommendation using probabilistic matrix factorization." Proceedings of the 17th ACM conference on Information and knowledge management. ACM, 2008.
- Martin Ester, Recommendation in Social Networks, Tutorial at RecSys 2013
- Zhepeng (Lionel) Li, Xiao Fang, and Olivia R. Liu Sheng. 2017. A Survey of Link Recommendation for Social Networks: Methods, Theoretical Foundations, and Future Research Directions. ACM Trans. Manage. Inf. Syst. 9, 1, Article 1 (October 2017), 26 pages. DOI: <https://doi.org/10.1145/3131782>
- **Zhang S, Yao L, Sun A. Deep Learning based Recommender System: A Survey and New Perspectives. arXiv:1707.07435**
- Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat-Seng Chua. 2017. Neural collaborative filtering. In Proceedings of the 26th International Conference on World Wide Web. International World Wide Web Conferences Steering Committee, 173–182
- Donghyun Kim, Chanyoung Park, Jinoh Oh, Sungyoung Lee, and Hwanjo Yu. 2016. Convolutional matrix factorization for document context-aware recommendation. In Proceedings of the 10th ACM Conference on Recommender Systems. ACM, 233–240.
- Chao-Yuan Wu, Amr Ahmed, Alex Beutel, Alexander J Smola, and How Jing. 2017. Recurrent recommender networks. In Proceedings of the Tenth ACM International Conference on Web Search and Data Mining. ACM, 495–503



内容回顾

94

- 什么是推荐系统
 - 背景、定义、应用场景
- 推荐方法概述
 - 兴趣建模
 - 推荐算法设计
 - 推荐结果的评估
- 案例学习
 - 基于用户兴趣扩展的个性化推荐方法
 - 面向推荐系统的纠结心理挖掘
- 小结及未来的路
- 资料推荐