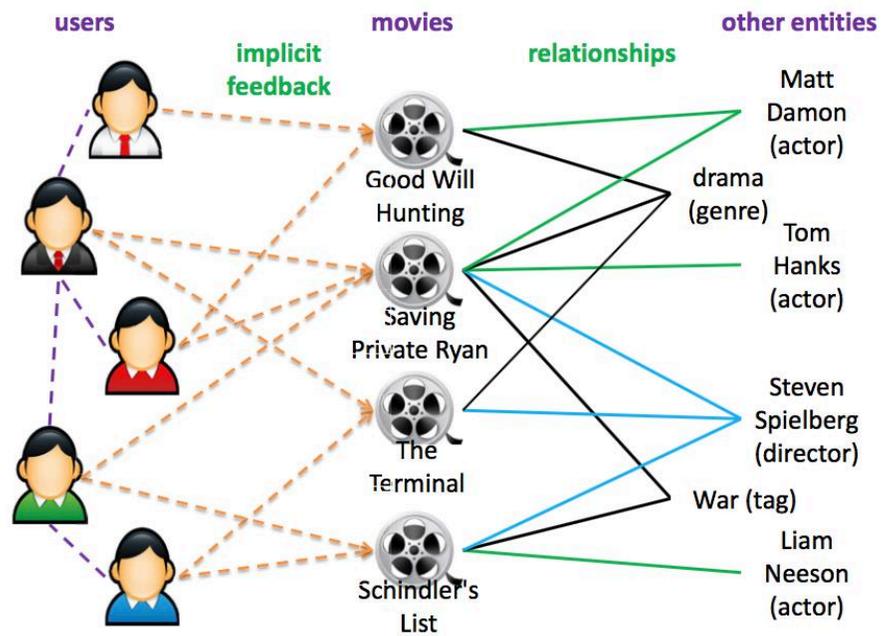


# Personal Entity Recommendation: A Heterogeneous Information Network Approach

@ WSDM' 2014

# 目标



- 通过挖掘异构网络上用户、商品和其他实体间的复杂关系实现给用户推荐潜在的感兴趣的商品。

## Implicit Feedback 矩阵分解

- 网络中有 $m$ 个用户 $\mathcal{U} = \{u_1, \dots, u_m\}$ 和 $n$ 个商品 $\mathcal{J} = \{e_1, \dots, e_n\}$ , Implicit Feedback矩阵 $R$ 的定义如下:

$$R_{ij} = \begin{cases} 1, & \text{if } (u_i, e_j) \text{ interaction is observed;} \\ 0, & \text{otherwise.} \end{cases}$$

- 需要注意的是 $R_{ij} = 1$ 并不一定表示 $u_i$ 对 $e_j$ 感兴趣,  $R_{ij} = 0$ 也不一定表示 $u_i$ 对 $e_j$ 不感兴趣。
- 此时, 对矩阵 $R$ 作非负矩阵分解(NMF), 使 $R \approx UV^T$ , 即可使用 $U$ 作为用户的embedding向量,  $V$ 作为商品的embedding向量, 点乘可得到一个推荐score, 按score排序即可。
- 这个方法效果并不好, 还有改进的空间。

## 基于Meta-Path的 潜在特征

- 从User出发随机游走，可以发现User和Item之间的一些潜在的联系，比如：

User<sub>1</sub>-钢铁侠-Robert Downey Jr.-复仇者联盟<sub>3</sub>

- 需要构造一个模型通过这些游走路径表达User和Item之间的潜在联系。

## 基于Meta-Path推荐模型

- 一类Meta-Path必须具有以下形式:

User-Item-\*...\*-Item

- 以电影推荐为例, 手动选择两类Meta-Path

User-Movie-Actor-Movie

User-Movie-Director-Movie

- 对于第 $q$ 类Meta-Path  $\mathcal{P}^{(q)}$ , 计算User  $u_i$  和 Item  $e_j$  间的相似度

$$s^{(q)}(u_i, e_j | \mathcal{P}^{(q)}) = \sum_{e \in \mathcal{J}} \frac{2 \times R_{ik} \times |\{p_{e_k \sim e_j}\}|}{|\{p_{e_k \sim e_k}\}| + |\{p_{e_j \sim e_j}\}|}$$

## 基于Meta-Path的 推荐模型

- 定义第 $q$ 类路径的矩阵 $\tilde{R}_{ij}^{(q)} = s^{(q)}(u_i, e_j | \mathcal{P}^{(q)})$ 。
- 对 $\tilde{R}_{ij}^{(q)}$ 作NMF得到 $\tilde{R}_{ij}^{(q)} \approx \hat{U}^{(q)} \hat{V}^{(q)T}$ ，使用 $\hat{U}^{(q)}$ 和 $\hat{V}^{(q)}$ 作为User和Item的embedding向量。
- Item  $e_j$  对User  $u_i$  的推荐score计算如下

$$r(u_i, e_j) = \sum_{q=1}^L \theta_q \cdot \hat{U}_i^{(q)} \hat{V}_j^{(q)T}$$

其中 $\theta_q$ 为第 $q$ 类路径的权重。

- 得到推荐score后，为User推荐前 $k$ 个Item即可。

## 基于Meta-Path的 个性化推荐模型

- 假设：不同群体的User兴趣可能存在一些差异，同一个群体内部的User兴趣可能比较相似。
- 改进：先对矩阵 $R$ 分解得到的 $U$ 进行聚类，得到 $k$ 个不同的用户群体，个性化推荐score计算方式如下：

$$r^*(u_i, e_j) = \sum_{k=1}^c \text{sim}(C_k, u_i) \sum_{q=1}^L \theta_q^{\{k\}} \cdot \hat{U}_i^{(q)} \hat{V}_j^{(q)T}$$

- 其中 $C_k$ 为第 $k$ 个用户群体的聚类中心， $\text{sim}(\cdot)$ 为余弦相似度， $\theta^{\{k\}}$ 为第 $k$ 个群体的 $\theta$ 。
- 核心思想是为不同的用户群体设置不同的 $\theta$ 来体现不同群体的差异。

## $\theta$ 参数估计

- 运用Learning-to-Rank思想。
- 使用 $p(e_a > e_b; u_i | \theta)$ 表示 $u_i$ 偏好 $e_a$ 大于 $e_b$ 的概率。
- 对于矩阵 $R$ 的每一行，使值为1的项排在值为0的项前面。

## 算法总结

- 输入：矩阵 $R$ ，异构网络 $G$ ；
- 设置 $L$ 类不同的Meta-Path，具有形式（User-Item-\*-Item）；
- 在网络中游走采样得到Meta-Path样本集合；
- 对于每一类Meta-Path，计算 $\tilde{R}^{(q)}$ ；
- 分解矩阵 $R$ 得到 $U$ 和 $V$ ；
- 在 $U$ 上运行k-means算法得到聚类结果 $C$ ；
- 对于聚类结果中的每个群体计算参数 $\theta$ ；
- 计算 $r^*(u_i, e_j)$ ，为 $u_i$ 推荐前 $k$ 个结果。