推荐系统实践学习笔记系列(六)

标签：推荐系统

6.1 获取社交网络数据的途径

- 电子邮件
- 用户注册信息
- 用户的位置数据
- 论坛和讨论组
- 即时聊天工具
- 社交软件
  ♦社交图谱(双向，Facebook)
  ♦兴趣图谱(单向，Twitter)

6.2 社交网络数据简介

1. 3种不同的社交网络数据
   - 双向确认的社交网络数据
   - 单向关注的社交网络数据
   - 基于社区的社交网络数据

2. 社交网络数据中的长尾分布
   社交网络中用户的入度和出度的分布也是满足长尾部分布的。

6.3 基于社交网络的推荐

1. 社会化推荐的优点
   - 好友推荐可以增加推荐的信任度
   - 社交网络可以解决冷启动问题

6.3.1 基于邻域的社会化推荐算法

最简单的算法是给用户推荐好友喜欢的物品集合。即用户u对物品i的兴趣p_{ui}可以通过如下公式进行计算：

$$p_{ui} = \sum_{w \epsilon out(u)} r_{vi}$$

其中out(u)是用户u的好友集合，如果用户v喜欢物品i，则$r_{vi} = 1$,否则$r_{vi} = 0$.
不同好友和用户u的熟悉程度和兴趣相似度不同，因此应该在推荐算法中考虑好友和用户的熟悉程度以及兴趣相似度：

$$p_{ui} = \sum_{w \epsilon out(u)} w_{uv} r_{vi}$$

其中w_{uv}有两部分组成，一部分是用户u对用户v的熟悉程度(familiarity)，另一部分是用户u和用户v的兴趣相似度(similiarity)。

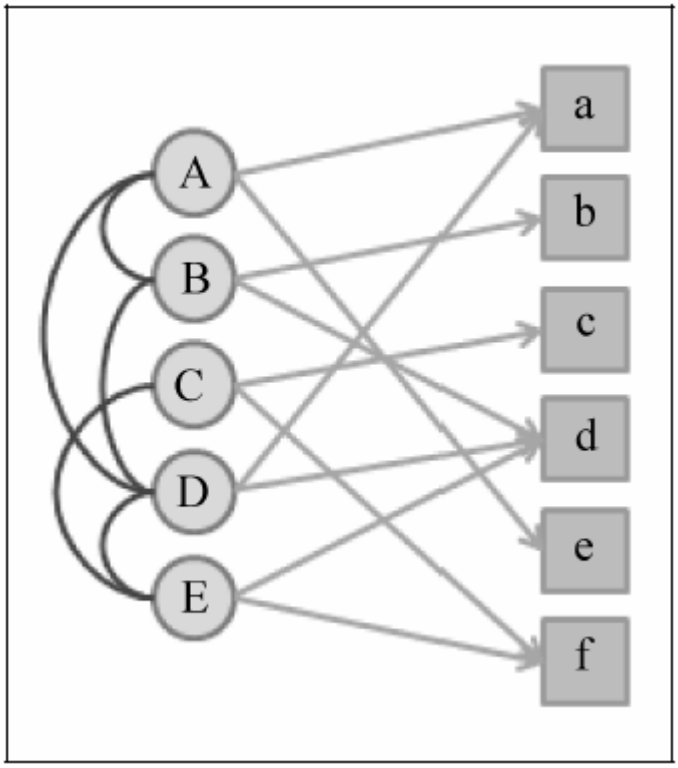
$$familiarity(u,v) = \frac{out(u) \bigcap out(v)}{out(u) \bigcup out(v)}$$

$$similiarity(u,v) = \frac{N(u) \bigcap N(v)}{N(u) \bigcup N(v)}$$

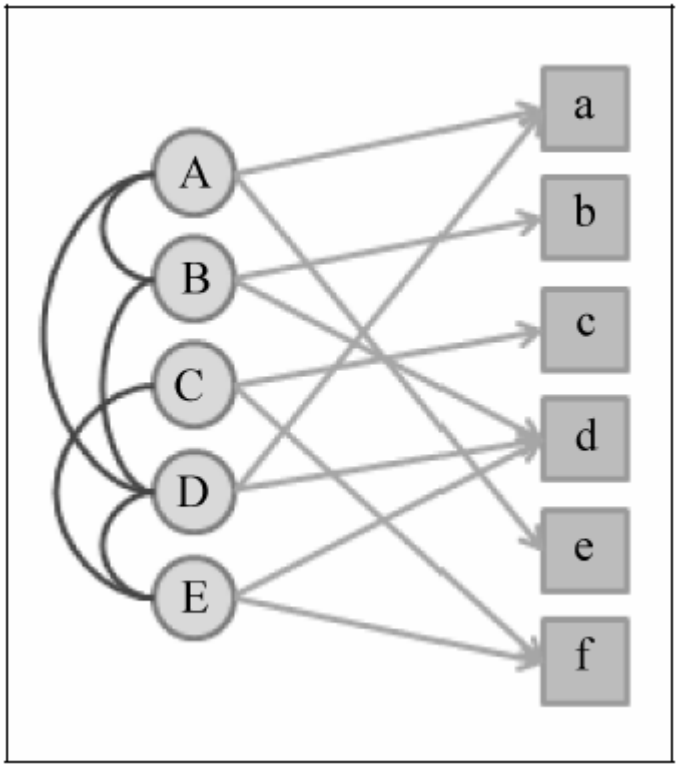
其中N(u)是用户u喜欢的物品集合。

6.3.2 基于图的社会化推荐算法

在社交网站中存在两种关系，一种是用户对物品的兴趣关系，一种是用户之间的社交网络关系。



社交网络图和用户物品二分图的结合



融合两种社交网络信息的图模型

6.3.3 实际系统中的社会化推荐算法

基于邻域的社会化推荐算法看起来非常简单，但在实际系统中却是很难操作的，这主要是因为该算法需要拿到用户所有好友的历史行为数据，而这一操作在实际系统中是比较重的操作。

解决方法：

- 简单地说，就是可以做两处截断。第一处截断就是在拿用户好友集合时并不拿出用户所有的好友，而是只拿出和用户相似度最高的N个好友。这里N可以取一个比较小的数。从而给该用户做推荐时可以只查询N次用户历史行为接口。此外，在查询每个用户的历史行为时，可以只返回用户最近1个月的行为，这样就可以在用户行为缓存中缓存更多用户的历史行为数据，从而加快查询用户历史行为接口的速度。此外，还可以牺牲一定的实时性，降低缓存中用户行为列表过期的频率。
- 要重新设计数据库。
  （1）首先，为每个用户维护一个消息队列，用于存储他的推荐列表；
  （2） 当一个用户喜欢一个物品时，就将（物品ID、用户ID和时间）这条记录写入关注该用户的推荐列表消息队列中；
  （3）当用户访问推荐系统时，读出他的推荐列表消息队列，对于这个消息队列中的每个物品，重新计算该物品的权重。计算权重时需要考虑物品在队列中出现的次数，物品对应的用户和当前用户的熟悉程度、物品的时间戳。同时，计算出每个物品被哪些好友喜欢过，用这些好友作为物品的推荐解释。

6.3.4 社会化推荐系统和协同过滤推荐系统

社会化推荐系统的效果往往很难通过离线实验评测，因为社会化推荐的优势不在于增加预测准确度，而是在于通过用户的好友增加用户对推荐结果的信任度，从而让用户单击那些很冷门的推荐结果。

6.3.5 信息流推荐

在Twitter和Facebook中，每个用户有一个信息墙，展示了用户好友最近的言论。我们只关心这些言论与自己相关的部分。，信息流的个性化推荐要解决的问题就是如何进一步帮助用户从信息墙上挑选有用的信息。
目前最流行的信息流推荐算法是Facebook的EdgeRank，该算法综合考虑了信息流中每个会话的时间、长度与用户兴趣的相似度。Facebook将其他用户对当前用户信息流中的会话产生过行为的行为称为edge，而一条会话的权重定义为：

$$\sum_{edgese} u_e w_e d_e$$

其中，u_e指产生行为的用户和当前用户的相似度，这里的相似度主要是在社交网络图中的熟悉度；
w_e指行为的权重，这里的行为包括创建、评论、like（喜欢）、打标签等，不同的行为有不同的权重;
d_e指时间衰减参数，越早的行为对权重的影响越低。

6.4 给用户推荐好友

好友推荐算法在社交网络上被称为链接预测（link prediction）。

6.4.1 基于内容的匹配

我们可以给用户推荐和他们有相似内容属性（用户人口统计学属性、用户兴趣、用户的位置信息）的用户作为好友。

6.4.2 基于共同兴趣的好友推荐

在Twitter和微博为代表的以兴趣图谱为主的社交网络中，用户往往不关心对于一个人是否在现实社会中认识，而只关心是否和他们有共同的兴趣爱好。因此，在这种网站中需要给用户推荐和他有共同兴趣的其他用户作为好友。
此外，也可以根据用户在社交网络中的发言提取用户的兴趣标签，来计算用户的兴趣相似度。

6.4.3 基于社交网络图的好友推荐

最简单的好友推荐算法是给用户推荐好友的好友。 三种算法：这些相似度的计算无论时间复杂度还是空间复杂度都不是很高，非常适合在线应用使用。

- 对于用户u和用户v，我们可以用共同好友比例计算他们的相似度：

$$w_{\text{out}}(u,v)=\frac{|\text{out}(u)\cap\text{out}(v)|}{\sqrt{|\text{out}(u)||\text{out}(v)|}}$$

- $w_{\text{out}}(u,v)$ 公式中out(u)是在社交网络图中用户u指向的其他好友的集合。我们也可以定义in(u)是在社交网络图中指向用户u的用户的集合。在无向社交网络图中，out(u)和in(u)是相同的集合。但在微博这种有向社交网络中，这两个集合就不同了，因此也可以通过in(u)定义另一种相似度：

$$w_{\text{in}}(u,v)=\frac{|\text{in}(u)\cap\text{in}(v)|}{\sqrt{|\text{in}(u)||\text{in}(v)|}}$$

- 这两种相似度的定义有着不同的含义，我们用微博中的关注来解释这两种相似度。如果用户u关注了用户v，那么v就属于out(u)，而u就属于in(v)。因此，$w_{\text{out}}(u,v)$ 越大表示用户u和v关注的用户集合重合度越大，而 $w_{\text{in}}(u,v)$ 越大表示关注用户u和关注用户v的用户的集合重合度越大。

  前面两种相似度都是对称的，也就是 $w_{\text{in}}(u,v)=w_{\text{in}}(v,u)$ ， $w_{\text{out}}(u,v)=w_{\text{out}}(v,u)$ 。同时，我们还可以定义第三种有向的相似度：

$$w_{\text{out,in}}(u,v)=\frac{|\text{out}(u)\cap\text{in}(v)|}{|\text{out}(u)|}$$

  这个相似度的含义是用户u关注的用户中，有多大比例也关注了用户v。但是，这个相似度有一个缺点，就是在该相似度的定义下所有人都和名人有很大的相似度。这是因为这个相似度在分母的部分没有考虑|in(v)|的大小。因此，我们可以用如下公式改进上面的相似度：

$$w'_{\text{out,in}}(u,v)=\frac{|\text{out}(u)\cap\text{in}(v)|}{\sqrt{|\text{out}(u)||\text{in}(v)|}}$$