推荐系统中的召回：综述

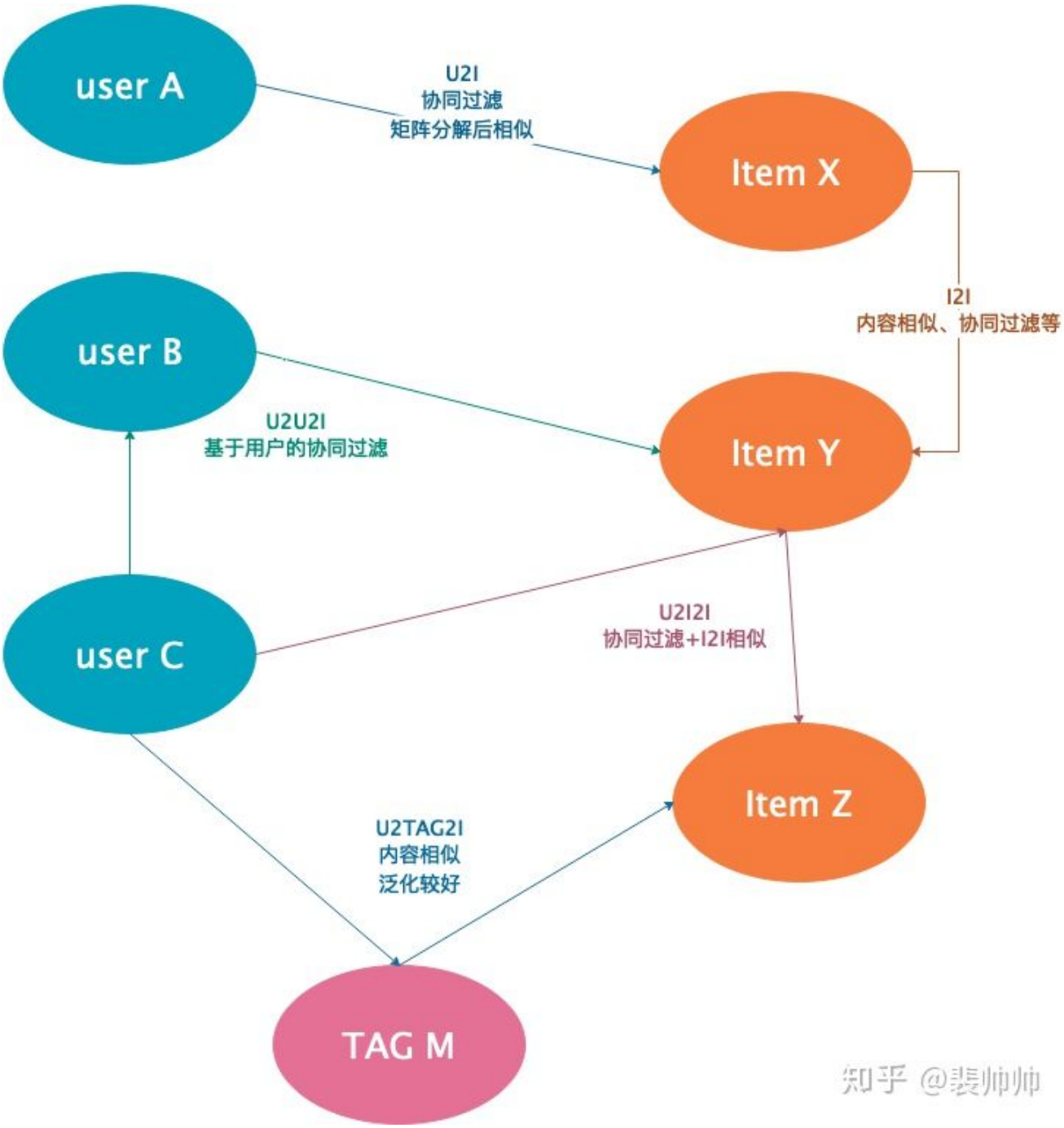
召回是推荐系统的第一阶段，主要根据用户和商品部分特征，从海量的物品库里，**快速**找回一小部分用户潜在感兴趣的物品，然后交给排序环节。这部分需要处理的数据量非常大，速度要求快，所有使用的策略、模型和特征都不能太复杂。

1. 基于内容的召回

使用item之间的**content相似性**来推荐与用户喜欢的item相似的item（i2i召回），一般也叫做标签召回。这类召回的核心思想是基于item自身的属性，将内容表达为连续向量的方式进行召回；或者更简单的，直接用倒排索引去召回那些相同**tag**的item。

例如：如果用户A看了《绣春刀2》这部杨幂主演的电影后，则会为他推荐杨幂主演的其他电影或电视剧, 因为这些商品与《绣春刀2》在内容上的embedding是类似的。这类方法的优点在于，具有比较好的可解释性，因为我们可以告诉用户是因为他买过什么物品，我们才做的如此推荐。

2. 协同过滤



- u2i：基于矩阵分解，得到user embedding & item embedding，然后求和user点积最大的item，推荐给user。
- i2i：计算item-item的内容相似度/矩阵分解后隐向量的相似度（“item相似当且仅当它们的受众相似”）。常用于seed item召回相似item。
- u2u2i：基于用户的协同过滤，先找相似用户，把相似用户喜欢的item推荐给用户；
- u2tag2i：基于标签的泛化推荐，先统计用户偏好的**tag向量**，然后匹配所有的Item，这个tag一般是item的标签、分类、关键词等tag；但是这样如果tag粒度过粗，会召回太多不相关的item。

然而路径的边其实可以不止一条两条，可以设置为4、5、6，让一个游标沿着顶点游走，并且每次都挑选一个概率游走，其实就是得到了U2I和I2I，只不过中间跨过了很多节点挖掘到了他们的关系。

其实，这个图就跟我们之前讲图网络做协同过滤非常相似，我们不仅有user-item关系，还有user-user、item-item相似度关系，这样我们就可以实现u2u2i或者i2i2u的图网络“游走”。

对于召回，一般都是多路召回，每一路去把握query和item不同方面的相似度。例如youtube next-video watch这个问题，我们根据一个seed video去找到用户可能感兴趣的其他内容，这就是一个i2i问题。那么，我们可以根据这个video的一些tag，去召回有相同tag的item；也可以通过item协同过滤，去找到user-item矩阵分解中，和item隐向量相似的那些item（intuition是，“受众相似的item也相似”）。

3. 基于FM模型的召回

对于FM，其优势可分以下三点:

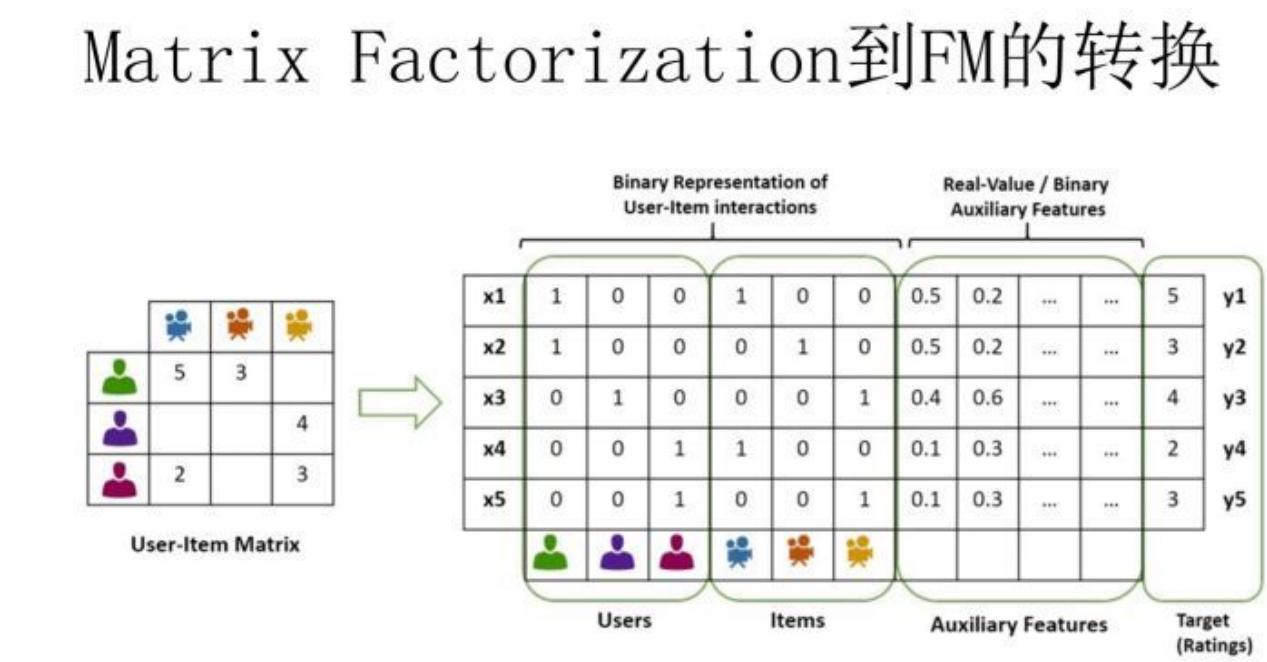
1. FM能处理数据高度稀疏场景，SVM则不能；
2. FM具有线性的计算复杂度，而SVM依赖于support vector。
3. FM能够在任意的实数特征向量中生效。

第一步，对于某个用户，我们可以把属于这个用户子集合的特征，查询离线训练好的FM模型中这个用户对应的**特征embedding向量**（FM模型求解出的隐向量，即 v_i ，其长度为 k，包含 k 个描述特征的因子），然后将这个用户对应的n个特征embedding向量累加，形成这个用户的兴趣向量U，这个向量维度和每个特征的维度是相同的。

类似的，我们也可以把每个物品，其对应的物品子集合的特征，查询离线训练好的FM模型对应的特征embedding向量，然后将m个物品子集合的特征embedding向量累加，形成物品向量I，这个向量维度和每个特征的维度也是是相同的。

第二步，对于每个用户以及每个物品，我们可以利用步骤一中的方法，将每个用户的兴趣向量离线算好，存入在线数据库中比如Redis（用户ID及其对应的embedding），把物品的向量逐一离线算好，存入Faiss(Facebook开源的embedding高效匹配库)数据库中，进行knn索引，然后高效检索。

第三步，当用户登陆或者刷新页面时，可以根据用户ID取出其对应的兴趣向量embedding，然后和Faiss中存储的物料embedding做内积计算，按照得分由高到低返回得分Top K的物料作为召回结果。



可以认为FM是加了content特征的矩阵分解（MF），原来用户和物品侧都只有一个id特征，现在用户侧加了年龄、性别、学历等特征，物品侧加了品类、店铺等特征，然后进一步融入到FM模型后，它将所有的特征转化为embedding低维向量表达，然后用户侧的特征和物品侧特征两两矩阵分解，即两两特征embedding的内积，得到特征组合的权重。

4. 基于神经网络的召回

双塔召回，多兴趣召回