

# 推荐系统期末报告（论文调研）

---

## 推荐系统期末报告（论文调研）

引言

MF

背景:

方法:

IF-MF

背景:

方法:

CB-MF

背景:

方法:

UIS-MF

背景:

方法:

DuLDA-MF

背景:

方法:

从MF到BPR

矩阵分解的缺陷:

使用BPR解决:

BPR-OPT

背景:

方法:

SBPR

背景:

方法:

BPRDR

背景:

方法:

TNDBPR

背景:

方法:

从MF到FM

FM与MF的不同

AFM

背景:

方法:

IFM

背景:

方法:

PRM

背景:

方法:

总结

发展方向

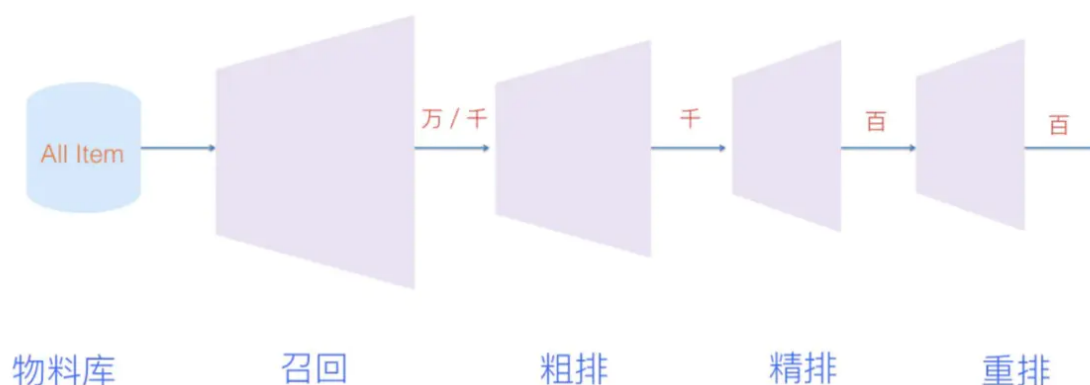
方法的出发点

## 引言

---

大型的推荐系统分为4个阶段：召回 --> 粗排 --> 精排 --> 重排

# 推荐系统细分四阶段：召回+粗排+精排+ReRanker



这次我主要关注点在最后的排序上面。

## MF

Koren Y, Bell R, Volinsky C. Matrix Factorization Techniques for Recommender Systems[J]. Computer, 2009, 42(8):30-37.

### 背景：

MF的提出是为了缓解用户物品矩阵的数据稀疏性以及为了提升推荐的准确性。

### 方法：

- 该方法把用户物品评分矩阵拆解成两个矩阵，分别是用户矩阵P和物品矩阵Q。用户矩阵P中的每一行都是一个用户向量表示，物品矩阵Q中的每一列都是一个物品向量表示，我们使用得到的P中的第i行和Q中的第j列相乘可以得到用户i对物品j的一个预测评分值。
- 在训练的时候，通过不断的缩小实际值与预测值的一个最小平方误差我们可以训练出最后的P和Q矩阵。

## IF-MF

Hu Y, Koren Y, Volinsky C. Collaborative Filtering for Implicit Feedback Datasets[C]// Eighth IEEE International Conference on Data Mining. IEEE, 2009.

### 背景：

这一篇论文的提出主要是为了处理隐性反馈数据，隐性反馈数据在现实生活中其实是比显性反馈数据要多很多的，所以我们应该去充分的利用这份数据去进行推荐工作，但是目前来说，我们对于隐性反馈数据的利用率不是很高，主要是因为**隐性反馈数据并不能和显性反馈数据一样去显性的表示出用户的偏好**。

### 方法：

作者认为隐性反馈数据的数值大小多少能够反映出用户对物品的偏好，这篇论文中利用的数据集是作者爬取的一个关于用户看电视节目的这么一个隐性反馈数据集，按照现实的理解，如果数据显示用户观看某一个电视节目1-2次，那么有可能是因为用户喜欢这个电视节目，也很有可能只是因为用户换台的时候经过或者无聊的时候在这个电视节目上做了停留，我们确实不能从这份数据中获取到用户的一个偏好。但是若是用户观看某个电视节目10多次呢？这确实是能够充分体现出用户对这个电视节目的一个喜好程度的。基于此，作者提出了一个叫置信程度的东西，置信程度在这里用 $C_{ui}$ 来表示。如果某一条隐性反馈数据能够反映用户偏好的话，则置信程度值会高，否则就会低，而这里的置信程度值完全和隐性

反馈值是成正相关的。

$$p_{ui} = \begin{cases} 1 & r_{ui} > 0 \\ 0 & r_{ui} = 0 \end{cases}$$

$$c_{ui} = 1 + \alpha r_{ui}$$

$$\min_{x_{\star}, y_{\star}} \sum_{u,i} c_{ui} (p_{ui} - x_u^T y_i)^2 + \lambda \left( \sum_u \|x_u\|^2 + \sum_i \|y_i\|^2 \right)$$

## CB-MF

Zhao G, Lee M L, Hsu W, et al. Community-based user recommendation in uni-directional social networks[J]. ACM, 2013.

### 背景:

这篇工作是2013年发表在CIKM上的一篇工作。这篇工作关注的是社交网络上的用户推荐任务，**由于关注者-被关注者矩阵非常稀疏，利用普通的MF进行矩阵分解操作并不能缓解准确率不高的事实。**

### 方法:

作者提出是否可以把该关注者-被关注者矩阵按照用户兴趣分成多个小矩阵，然后再来进行矩阵分解，作者最后利用LDA技术来获取到所有用户的关注者们的一个兴趣主题分布以及兴趣主题分布下的用户群体。再利用兴趣主题分布下的用户群体去进行小矩阵的划分，最后再在每个小矩阵上面进行矩阵分解。

|     | u1 | u2 | u3 | u4 | u5 | u6 | u7 | u8 | u9 | u10 |
|-----|----|----|----|----|----|----|----|----|----|-----|
| u1  |    | 1  | 1  | 1  |    | 1  |    |    |    |     |
| u2  | 1  |    | 1  |    | 1  | 1  |    |    |    |     |
| u3  |    | 1  |    | 1  | 1  | 1  |    |    |    |     |
| u4  | 1  |    | 1  |    |    | 1  |    |    |    |     |
| u5  | 1  | 1  | 1  |    |    | 1  |    |    |    |     |
| u6  |    |    |    |    |    |    | 1  |    | 1  | 1   |
| u7  |    |    | 1  |    |    | 1  |    |    | 1  |     |
| u8  |    |    |    |    |    |    | 1  |    |    | 1   |
| u9  |    |    |    |    |    |    | 1  | 1  |    | 1   |
| u10 |    |    |    |    |    | 1  |    |    | 1  |     |

## UIS-MF

Xu K, Zheng X, Cai Y, et al. Improving User Recommendation by Extracting Social Topics and Interest Topics of Users in Uni-Directional Social Networks[J]. Knowledge-Based Systems, 2017:S0950705117305002.

## 背景:

UIS-MF这篇工作也是用来处理社交网络上的用户推荐任务。这篇论文提出的其中之一原因与CB-MF原因一致，即由于关注者-被关注者矩阵非常稀疏，利用普通的MF进行矩阵分解操作并不能缓解准确率不高的事实，另外一个原因是CB-MF也没有考虑到的，即关注者关注被关注者可能是因为兴趣原因，但仍然有可能是因为社交原因，因为在线下认识的朋友，我们倾向于在线上也是朋友，而这一点在目前的论文中都没有被考虑到。

## 方法:

使用基于原始LDA得到的兴趣分布下的用户群体显然是很粗糙的，如果我们能够使用改进的UIS-LDA进行训练，则我们可以得到兴趣分布下的用户群体以及社交分布下的用户群体，基于这两个群体去进行小矩阵的划分，会比前者考虑的更加细致，得到的群体矩阵更加精致和准确，最后用户推荐任务的准确率也会更好。

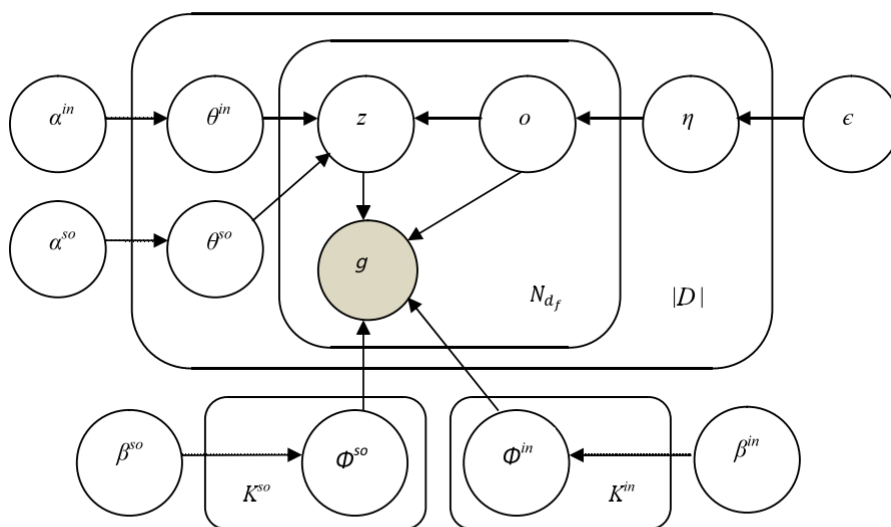


Fig. 1. The plate notation for UIS-LDA.

## DuLDA-MF

Ke X, Yi C, Min H, et al. Top-N Trustee Recommendation with Binary User Trust Feedback[M]. 2018.

## 背景:

在信任网络中，用户去信任并且关注某一个人有时候并不仅仅是因为用户本身的一个兴趣原因，也有可能是因为被关注者本身影响力很高值得信任的原因。而现有的大部分做法，不管是在社交网络中也好，或者是在信任网络中也好，都只考虑了关注者（信任者）本身兴趣偏好，而没有考虑过被关注者（被信任者）所拥有的影响力的原因。

## 方法:

作者提出这个想法之后，对原始的LDA进行了改进操作，在工作UIS-LDA中我们只输入了用户关注者们表示的文档，得到了一个用户关注者的兴趣主题分布，而这里，我们对LDA进行训练，分别输入两个文档，一个是信任者的被信任者们的一个文档表示，另一个是被信任者的信任者们的一个文档表示，由此可以得到两个主题分布，分别是信任者们的兴趣分布，以及被信任者们的一个影响力分布，同时我们可以获得这两个分布下的用户分布。利用该用户分布，我们可以参照CB-MF的做法，划分小矩阵群体，把大的信任者-被信任者用户矩阵划分成很多的小的兴趣矩阵以及影响力矩阵，分别进行矩阵分解操作。

# 从MF到BPR

---

## 矩阵分解的缺陷：

矩阵分解是通过预测用户对候选物品的评分，然后根据这个预测评分去排序，最后再推荐给用户。这种方法是一种典型的 Pointwise 方法，无论是预测评分还是预测隐式反馈，本质上都是在预测用户对一个物品的偏好程度。

但是这种方法有很大的问题，因为很多时候我们只能收集到少数正例样本，剩下的数据其实是真实负例和缺失值的混合构成（这里的缺失值是指训练数据中除正例和负例外的未知数据，可以理解为未曝光或者曝光了的但是用户可能没有注意到缺失数据，所以缺失值中的样本即有可能是正例，也有可能是负例），而我们用这种方法构建训练数据的时候，往往无法确定负例到底是哪些，就只能把除正例以外的其他部分都当作是负例，这就会使得训练数据中负例的一部分其实是缺失值。把缺失值当作是负样本，再以预测误差为评判标准去使劲逼近这些样本。逼近正样本没问题，但是同时逼近的负样本只是缺失值而已，真正呈现在用户面前，并不能确定是不喜欢还是喜欢。而且，这样的模型仅能预测正例或负例，对于类别内的样本无法深入区别其重要性，不利于排序。

当然，对于这种情况，我们也可以用一些其他方法来规避这些问题，比如负例采样，比如按预测概率排序，但这些方法也仅仅是缓兵之计，对于解决排序问题来说并不完善。

## 使用BPR解决：

- BPR 利用 Pairwise 的思想来构建偏序关系，它依然没有从无反馈数据中去区分负例样本和缺失值，不过和之前的方法不一样的是，BPR 不是单纯地将无反馈数据都看做是负例，而是与正例结合起来构建偏序关系。
- 偏序关系是反对称的，因此隐式地考虑了负样本。

## BPR-OPT

---

Rendle S, Freudenthaler C, Gantner Z, et al. BPR: Bayesian personalized ranking from implicit feedback[C]// UAI 2009, Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence, Montreal, QC, Canada, June 18-21, 2009. AUAI Press, 2009.

## 背景：

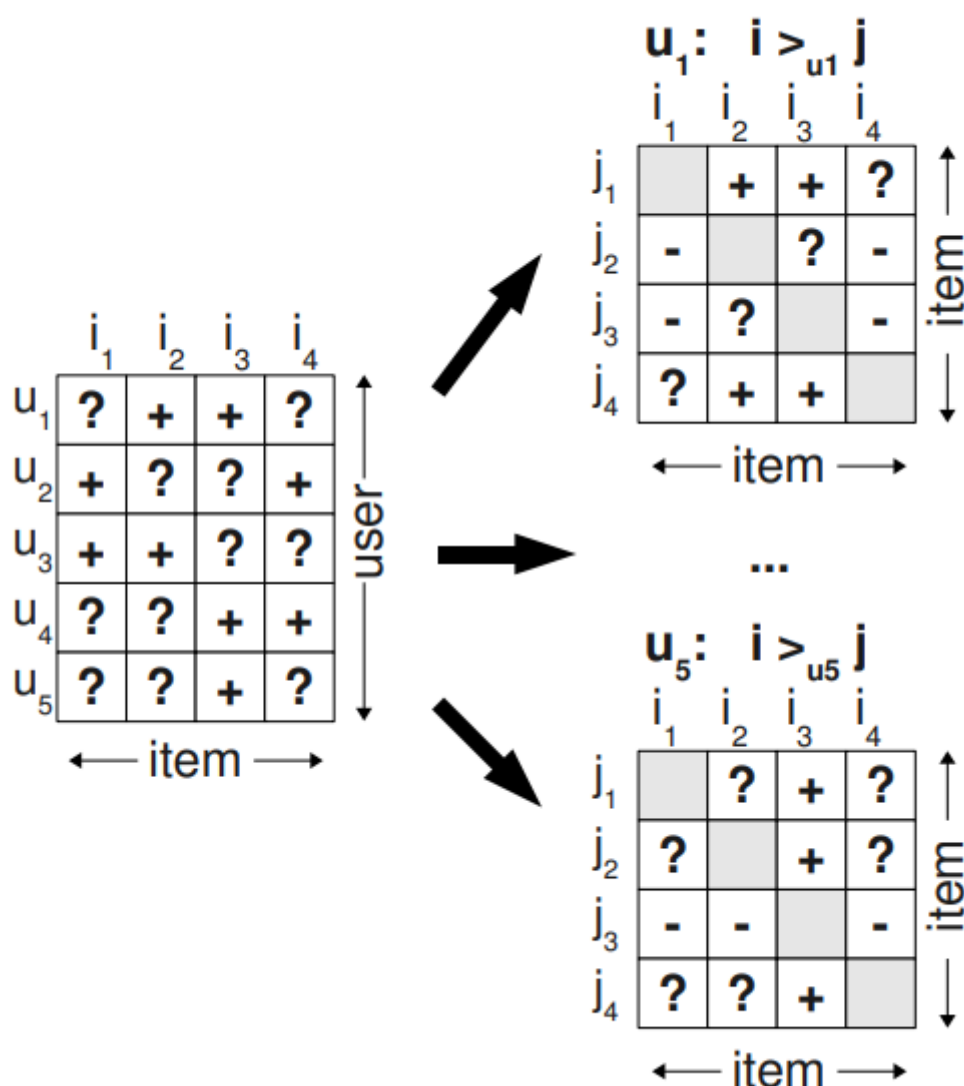
全称叫做贝叶斯个性化排序，此方法的提出主要是针对TOPN推荐任务，当时的TOPN推荐任务主要是取预测评分的TopN来进行推荐，而这些预测评分基本上都是利用前面提及到的通过降低实际值与预测值之间的最小平方差而求到的，而事实上，这样求出来的预测评分高低并不能代表用户心目中的一个排序的高低。这样的排序是没有用的，所以即使的RMSE等指标都很好，但是准确率以及召回率等TopN评价指标都很低。

所以作者提出的BPR专门针对TopN的排序任务，**核心假设是，某用户对他有过反馈的物品的偏好程度一定比没有反馈过的物品高**，提出如果你面对 i 物品和 j 物品的时候选择了 i 物品，则表示相对于 j 物品来说，你更喜欢 i 物品。所以我们的做法是在训练的时候加大用户对 i 物品的喜欢程度和对 j 物品喜欢程度的一个差值，并且这个差值越大越好。

## 方法：

$$\begin{aligned}
\text{BPR-OPT} &:= \ln p(\Theta | >_u) \\
&= \ln p(>_u | \Theta) p(\Theta) \\
&= \ln \prod_{(u,i,j) \in D_S} \sigma(\hat{x}_{uij}) p(\Theta) \\
&= \sum_{(u,i,j) \in D_S} \ln \sigma(\hat{x}_{uij}) + \ln p(\Theta) \\
&= \sum_{(u,i,j) \in D_S} \ln \sigma(\hat{x}_{uij}) - \lambda_{\Theta} \|\Theta\|^2
\end{aligned}$$

上面的公式是bpr的通用公式，可以把它套在任何一个现存的方法上进行训练。



把用户-物品矩阵分解成单独的用户矩阵，表示用户对物品对之间的喜爱的偏好。

**核心假设是，某用户对他有过反馈的物品的偏好程度一定比没有反馈过的物品高**（这里的反馈一般指隐式反馈，如点击浏览等，不涉及负反馈），未反馈的物品包括真正的负例以及缺失值。BPR 试图通过用户的反馈矩阵  $S$  来为每一个用户构建出完整的偏序关系，也称全序关系。BPR 其实可以看作在准确率之上更加高层的排序算法框架。

## SBPR

---

Tong Z, Mcauley J, King I. Leveraging Social Connections to Improve Personalized Ranking for Collaborative Filtering[C]// the 23rd ACM International Conference. ACM, 2014.

### 背景：

这篇工作第一次把社交网络关系加入到了BPR的猜想中，该作者认为**用户都会更加倾向于用户朋友喜欢的东西**，而目前的很多工作并没有考虑到这一点。

### 方法：

$$x_{ui} \succeq x_{uk}, x_{uk} \succeq x_{uj}, i \in P_u, k \in SP_u, j \in N_u,$$

1、用户更加喜欢自己有过正向反馈的物品而不是有过负向反馈或者没有过反馈的物品（与BPR的猜想一致）；2、用户更加偏向于用户的朋友喜欢的物品而不是自己有负向反馈或者没有反馈的物品。

## BPRDR

---

Xu K, Xu Y, Min H, et al. Improving Item Ranking by Leveraging Dual Roles Influence[J]. IEEE Access, 2018, 6:57434-57446.

### 背景：

**作者认为用户选择某一个物品不仅仅会受自己信任的人所影响，也会受信任自己的人所影响**，所以这两点因素都应该要被考虑进去。

### 方法：

- $u$  prefers her observed item  $i$  to any of her trustee's observed item  $k$ ;
- $u$  prefers her observed item  $i$  to any of her truster's observed item  $s$ ;
- $u$  prefers her trustee's observed item  $k$  over item  $j$  that neither herself nor her trusters/trustees observed.
- $X(u, i) \geq X(u, k), X(u, i) \geq X(u, s), X(u, k) \geq X(u, j)$

1、用户更加倾向于自己喜欢的物品而不是信任他的人喜欢的物品；2、用户更加倾向于自己喜欢的物品而不是他信任的人喜欢的物品；3、用户更加倾向于他信任的人喜欢的物品而不是自己、他信任、信任他的人都不喜欢的物品。

## TNDBPR

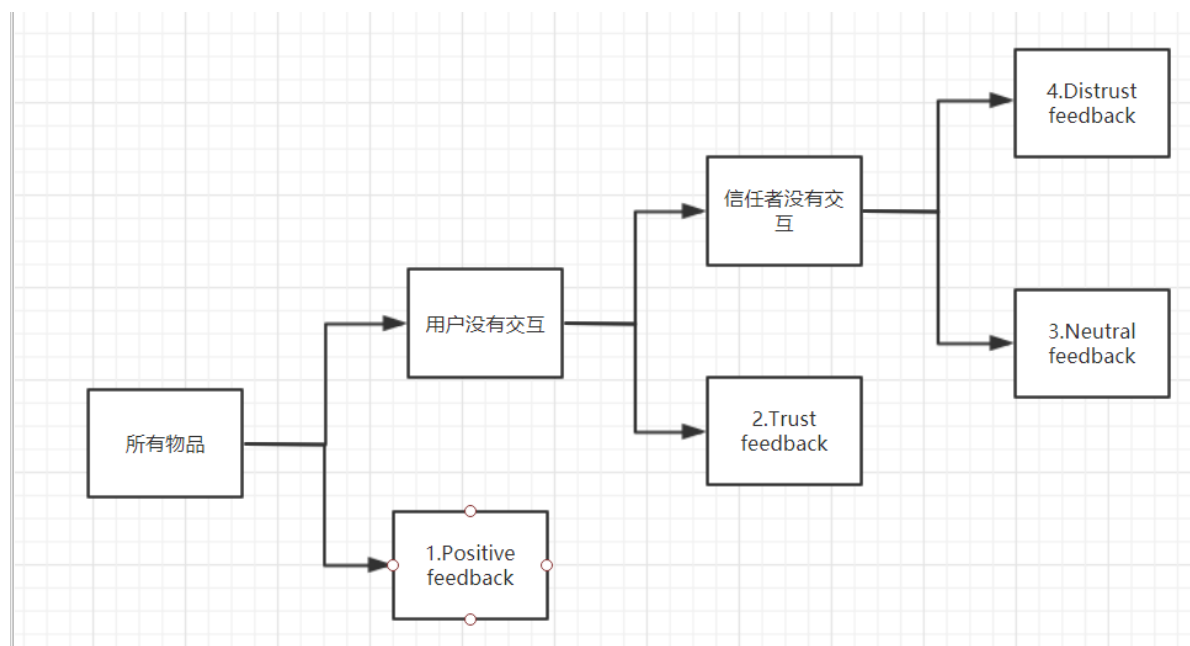
---

Yangjun, Ke, Cai, et al. Leveraging Distrust Relations to Improve Bayesian Personalized Ranking[J]. Information, 2018.

## 背景：

在以往的算法中往往都只考虑了利用信任信息，比如前一篇文章BPRDR，而没有人利用过用户之间的不信任信息。而事实上，利用这一块信息也是至关重要的，利用用户之间的不信任信息可以帮助用户排除掉用户不信任的人喜欢的物品。

## 方法：



- $x_{ui} \succeq x_{uk}, x_{uk} \succeq x_{uj}, x_{uk} \succeq x_{uz}$ ,
- 作者把物品分为了四种类型，分别是：1、Positive feedback，这一类物品是用户有过用户行为的物品；2、Trust feedback，这一类物品是用户没有过用户行为的物品，但是至少有他的一个信任者对该物品有过用户行为；3、Distrust feedback，这一类物品是用户和用户信任者都没有过用户行为的物品，但是至少有一个用户的不信任者对其有过用户行为的物品；4、Neutral feedback，这一类物品是用户和用户信任的人以及用户不信任的人都没有过用户行为的物品；
- 作者提出了以下三个猜想：1、用户更加倾向于喜欢有过Positive feedback的东西而不是有过Trust feedback的东西；2、用户更加倾向于喜欢有过Trust feedback的东西而不是有过Neutral feedback的东西；3、用户更加倾向于喜欢有过Trust feedback而不是有过Distrust feedback的东西；

## 从MF到FM

### FM与MF的不同

- 输入数据的形式不同：

一般来说，FM的输入数据是一个实值特征向量，相当于是对多个one-hot特征的拼接；MF的输入数据是一个二元组 $(u, i)$ ， $u$ 为用户ID， $i$ 为物品ID，也就是说，MF中的数据只有user和item。

- 融合附加信息的方式不同：

他们输入数据的形式决定了他们融合附加信息的形式不同。直观来讲，FM融合附加信息的方式更直接。比如FM融合边信息直接在列的维度拼接特征即可，比如增加性别、年龄等信息；而MF融合边信息不能直接在输入数据上拼接，而是需要通过增加正则项约束，或者在预测函数上做文章。

- MF是FM的特例：



MF可以被认为是只有User ID 和Item ID这两个特征信息时的FM模型。FM是更全能的MF。

## AFM

Xiao J, Ye H, He X, et al. Attentional Factorization Machines: Learning the Weight of Feature Interactions via Attention Networks[J]. 2017.

### 背景:

AFM以FM为基础,认为FM通过以相同权重建模所有特征交互的方式存在问题,因为不是所有特征交互都是同样有价值的,相反没有多大用的交互特征反而会引入噪声;AFM (Attentional Factorization Machine) 通过对FM不同的特征交互引入不同重要性来改善FM,重要性通过注意力机制来学习。

### 方法:

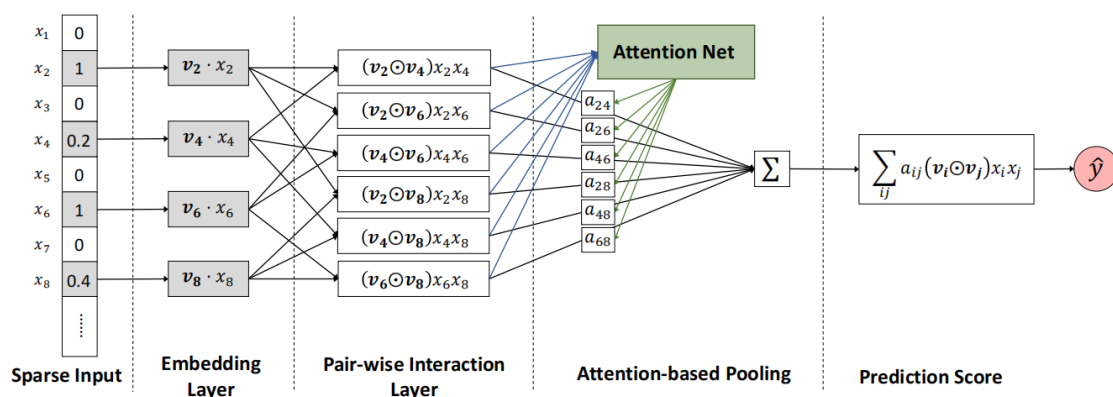
下图是FM的公式, AFM主要在 $\hat{\omega}_{ij}$ 上面做文章, 我们需要修改 $\hat{\omega}_{ij}$ 对应的权重

$$\hat{y}_{FM}(\mathbf{x}) = \underbrace{w_0 + \sum_{i=1}^n w_i x_i}_{\text{linear regression}} + \underbrace{\sum_{i=1}^n \sum_{j=i+1}^n \hat{\omega}_{ij} x_i x_j}_{\text{pair-wise feature interactions}}, \quad (1)$$

作者在一番推导之后把上述公式改写成了

$$\hat{y}_{AFM}(\mathbf{x}) = w_0 + \sum_{i=1}^n w_i x_i + \mathbf{p}^T \sum_{i=1}^n \sum_{j=i+1}^n a_{ij} (\mathbf{v}_i \odot \mathbf{v}_j) x_i x_j, \quad (6)$$

为了解决FM不能区分特征交互重要性的问题, 论文中在二阶交互特征计算完成后加入了注意力网络 (Attention Net), 使用多层感知器 (MLP) 参数化注意力得分。注意力网络可以给予不同的特征交互不同的贡献。这些“贡献”大小用 $a_{ij}$ 表示,  $a_{ij}$ 表示第 $i$ 个特征和第 $j$ 个特征的组合特征对于结果 $\hat{y}$ 的重要程度。



## IFM

### 背景:

作者提出了同一个特征在不同的情况下通常不具有同样的预测性和有用性。事实上, 特征的不准确表示甚至可能引入噪音, 降低整体性能。举个例子: 如果有一条样本长<年轻, 女性, 学生, 喜欢粉色>, target item是连衣裙, 那么这一条样本都对整个target item具备比较大的作用, 比如说女性整个特征, 跟连衣裙交叉就应该获得比较大的权值。我们接着来看另一条样本: <年轻, 女性, 学生, 喜欢蓝色>, target item是笔记本, 这个时候女性这个特征对于笔记本这个item就不应该赋予很大的权值。

## 方法:

作者改写了原来的FM的公式，可以看出来主要变化在于 $\omega_i$ 和 $V_i$ 加入了 $X$ ,这就意味着每个样本的 $\omega_i$ 和 $V_i$ 都是自己独有的，尽管两条样本都具备相同的特征，比如之前说的女性这个特征，那么他们具备各自的权值。

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

$$\hat{y}_{IFM}(\mathbf{x}) = w_0 + \sum_{i=1}^n w_{\mathbf{x},i} x_i + \sum_{i=1}^n \sum_{j=i+1}^n \langle \mathbf{v}_{\mathbf{x},i}, \mathbf{v}_{\mathbf{x},j} \rangle x_i x_j \quad (3)$$

对输入的特征做embedding并拼接起来，然后经过一个网络，最后Reweighting。

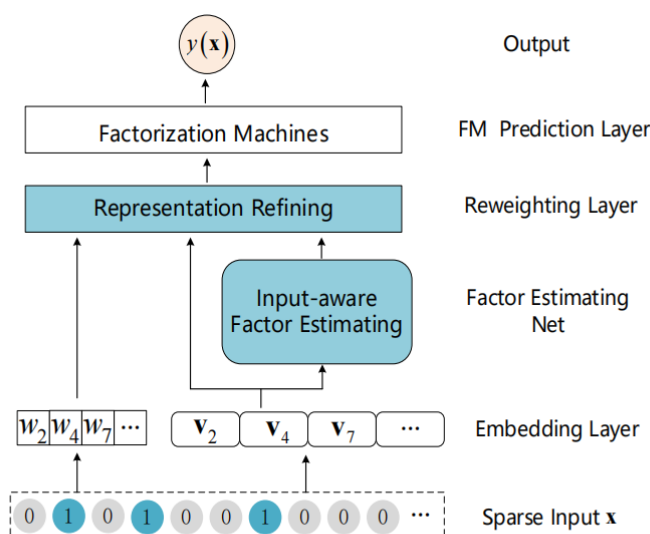


Figure 1: The network architecture of our proposed Input-aware Factorization Machines model

## PRM

Changhua Pei, Yi Zhang, Yongfeng Zhang, Fei Sun, Xiao Lin, Hanxiao Sun, Jian Wu, Peng Jiang, Junfeng Ge, Wenwu Ou, Dan Pei. Personalized re-ranking for recommendation[P]. Recommender Systems, 2019.

## 背景:

排序阶段从标记的数据集中学习排序功能以优化全局性能，从而为每个单独的物品生成一个得分。但是可能不是最佳选择，因为评分功能分别应用于每个物品，并且没有明确考虑物品之间的相互影响以及用户偏好或意图的差异。

## 方法:

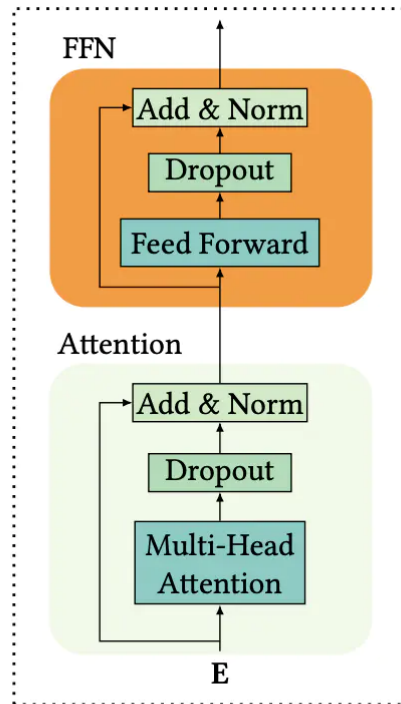
在重排序阶段引入了Transformer结构来对每一个物品对关系进行建模，从而直接优化整个推荐列表，进一步提高性能。

物品对之间的相互影响可以直接从现有LTR模型为请求 $r$ 给出的初始列表 $\mathbf{S}_r = [\mathbf{i}_1, \mathbf{i}_2, \dots, \mathbf{i}_n]$ 中学习。但是，很少有模型考虑用户和项目之间的交互。物品对的相互影响程度因用户而异。本文引入了个性化矩阵PV来学习用户特定的编码功能，该功能可以对物品对之间的个性化相互影响进行建模。

新的loss定义如下：

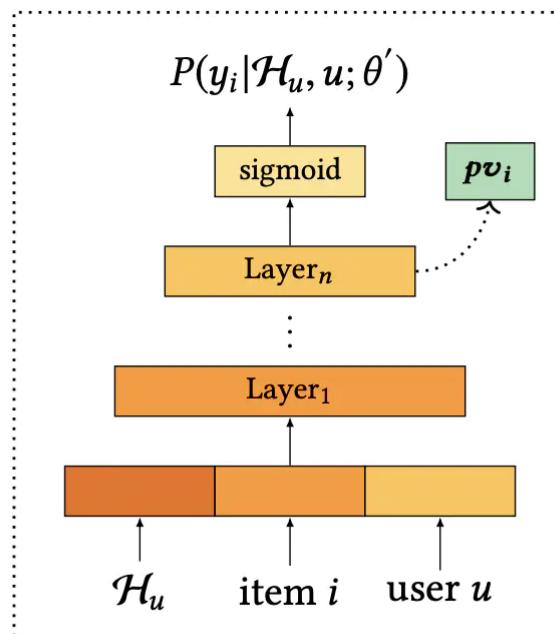
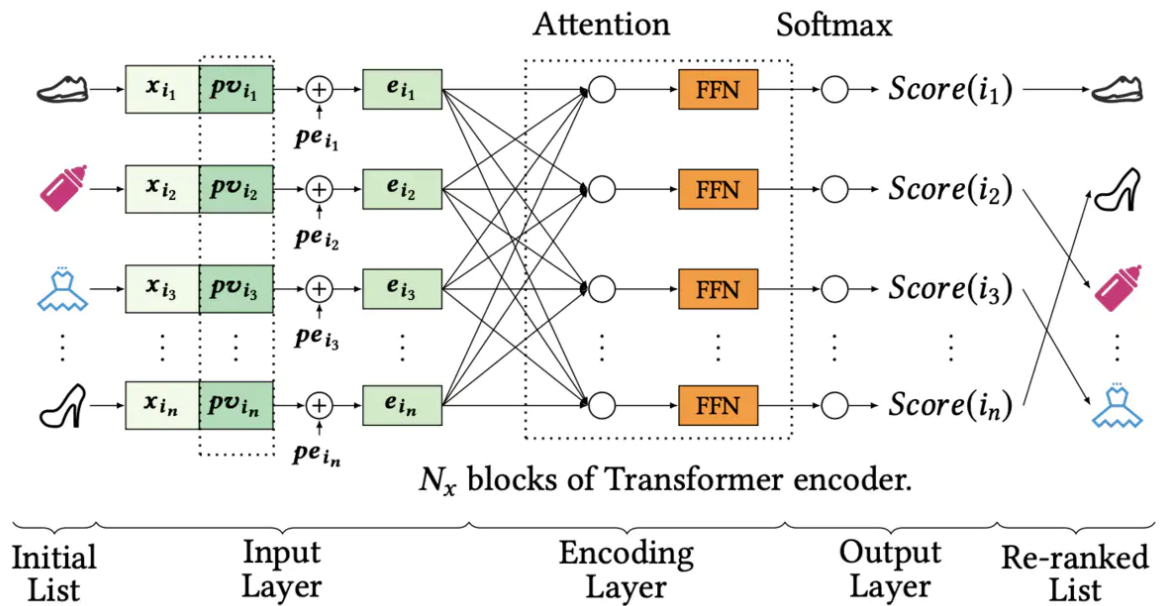
$$\mathcal{L} = \sum_{r \in \mathcal{R}} \ell(\{y_i, P(y_i|X, PV; \hat{\theta}) | i \in \mathcal{S}_r\})$$

比上面的loss多了PV（个性化向量矩阵），其他的是一样的， $\mathcal{S}_r$ 是初始化列表，参数变成来re-rank模型的参数。



(a) One block of Transformer encoder.

个性化部分，用来产生PV向量，在整个模型训练的一开始就要先进行预训练。 $H_u$ 是用户u交互过的所有物品，除此之外，用户的性别，年龄等信息也会作为特征输入。



其实就是在重排的时候加入了用户的个性化内容（用户自己的+用户和物品的互动关系）和物品对之间的关系。

## 总结

## 发展方向

- 处理的数据量在不断增大
- 深度化发展：现在提出的方法更多的会用到神经网络甚至是多层神经网络，推荐系统算法正在向深度化发展。
- 方法之间会有更多的交互
- 特征表示更多元复杂化，发展方向就是如何对：人、物、行为等数据进行向量表示，基于此向量表示再进行与目标之间相关性学习。

## 方法的出发点

- 处理数据稀疏问题：MF、CB-MF
- 隐性数据的应用：IF-MF、UIS-MF、DuLDA-MF、SBPR、BPRDR、TNDBPR
- 物品和用户的个性化：AFM、IFM、PRM