

Recommend Diversity Survey

1. 定义

推荐多样性(Diversity)是指尽可能多的将每种项目推荐给用户，而不局限于少数种类。多样性往往是衡量推荐系统的一个重要指标。

推荐系统的多样性与准确性(Accuracy)是一对相互矛盾的关系。若提高多样性必然会导致准确性的损失。如，总是推荐热门的项目给用户，虽然准确度很高，但是其多样性很差，若提高多样性，就需要推荐给用户相对不那么热门的项目，这会损失部分准确性。事实证明，仅仅通过提高准确率已经不能足以反映用户的潜在兴趣，多样性越来越受到关注，它可以推荐许多人们意想不到的项目或者是能激发人们兴趣的项目。因此很多研究是在多样性与准确性之间寻找一个平衡点(tradeoff)。

2. 传统推荐系统存在的问题

长尾效应(long-tail effect)

“超级明星”效应(superstar effect)

3. 推荐系统多样性分类与测量方法

系统级别的多样性(Aggregate Diversity)是指从系统整体角度出发，以减缓“长尾效应”为目标，提升不同种类项目被推荐的频率。[1]给出了常见的测量方法：

1. absolute long-tail metrics (某个固定排名的项目被推荐的次数)
2. relative long-tail metrics (低于某个排名的项目被推荐的次数)
3. the slope of the log-linear relationship (项目排名和推荐次数往往成幂律分布或指数函数分布，因此可以通过拟合并测量相关系数决定)

在系统级别下，Item-Based 的协同过滤算法的多样性远远优于 User-Based 方法，因为 User-Based 方法往往倾向于推荐热门的物品，而 Item-Based 方法有较好的新颖性，擅长推荐长尾中的物品（只要有那么一些用户，他同时购买了两个冷门物品，这两个冷门物品之间就有较高的相似度）。所以尽管 Item-Based 方法精度略小于 User-Based，但是如果考虑到多样性，Item-Based 方法好很多。

用户级别的多样性(Individual Diversity)是指从用户个人角度出发，尽可能多的推荐不同种类的项目，即推荐列表中项目两两之间的相似度，通常的测量方法是计算在推荐中，每个类型被覆盖到的百分比。该方法使用 User-Based 协同过滤算法要优于 Item-Based 算法，因为 Item-Based 算法往往与用户的历史记录相似。

4. 提升多样性方法

通常分为两类方法：一种是在设计推荐系统时就考虑到多样性，并作为 loss 函数的一部分；另外一种 Re-Rank 方法，对设计好的推荐系统的输出结果重新排序，以增加多样性。

4.1. 在设计推荐系统时考虑推荐多样性

这一类文章往往是提出一个改进的推荐系统，多样性是其中的一个评价指标。例如[4]中的推荐系统使用了监督学习实现，改善了基于矩阵分解(Matrix Factorization, MF)算法的推荐系统的多样性。其 Ground-Truth 的设计使用了贪心的策略，首先由推荐系统产生 c 个待选的项目， $C = \{i_1, i_2, i_3, \dots, i_c\}$ ，假设最终需要从中选出 $l (l \leq c)$ 个推荐给用户，那么在选择 Ground-Truth 时项目集 y_u 时应满足：

$$\arg \max_{y_u \in C^k} \frac{2 \times f(y_u) \times g(y_u)}{f(y_u) + g(y_u)}. \quad (\text{F-measure})$$

其中 $f(y_u)$ 为准确度， $g(y_u)$ 为多样度。多样度测量采用的是第 3 节中所说的用户级别的多样性(Individual Diversity)测量，值等于任意两个项目的不相似度的平均值，即

$$g(y) = \frac{\sum_{i,j \in y, i \neq j} d(i,j)}{\frac{1}{2}l(l-1)},$$

$d(i,j)$ 为项目 i 和项目 j 的不相似度，定义如下：

$$d(i,j) = -\cos(\mathbf{i}, \mathbf{j}) = -\frac{\mathbf{i}^\top \mathbf{j}}{\|\mathbf{i}\| \times \|\mathbf{j}\|}.$$

然后作者设计了参数化的矩阵分解(Parameterized Matrix Factorization)算法：增加了参数 $\sigma = \{\sigma_1, \sigma_2, \dots, \sigma_k\}$ 来表示矩阵分解不同的解，将传统的矩阵分解修改为： $\mathbf{R}_{m \times n} \approx \mathbf{U}_{k \times m}^\top \Sigma_{k \times k} \mathbf{I}_{k \times n}$ 其中 Σ 为参数 σ 的对角矩阵。具体参数向量 σ 的值可以通过学习得到。

该系统的 Loss 函数使用两部分表示，其一为参数化矩阵分解的 loss 函数：

$$\arg \min_{\mathbf{U}, \mathbf{I}} \sum_{\forall r_{u,i} \in \mathbb{R}_0} (r_{u,i} - \mathbf{u}^\top \Sigma \mathbf{i})^2 + \lambda (\|\mathbf{U}\|_F^2 + \|\mathbf{I}\|_F^2),$$

其中 $r_{u,i}$ 为针对用户 u ，项目 i 的相关度(rating)值。

另外一部分是总的学习 loss 函数，该 loss 函数充分考虑了多样性和准确性的权衡：

$$\begin{aligned} & \arg \min_{\mathbf{w}, \sigma} \frac{1}{2} (\|\mathbf{w}\|^2 + \|\sigma\|^2) + C\xi \\ & \text{s.t. } \forall \mathbf{u} \in \mathbf{U}, \forall y \in \mathcal{Y} \setminus y_u : \\ & \mathbf{w}^\top \sum_{u \in U} [\Psi(\mathbf{u}, \mathbf{I}_u, \sigma, y_u) - \Psi(\mathbf{u}, \mathbf{I}_u, \sigma, y)] \geq \sum_{u \in U} \Delta(y_u, y) - m\xi \end{aligned} \quad (5)$$

其中 \mathbf{w} 为学习参数， σ 为前边定义参数向量， C 为权衡的值。 Ψ 为联合特征(Joint Features)矩阵

$$\Psi(\mathbf{u}, \mathbf{I}_u, \sigma, y) = \begin{bmatrix} z_1 \sum_{\forall i \in y, j \in \mathbf{I}_u \setminus y} (\mathbf{u}^\top \Sigma \mathbf{i} - \mathbf{u}^\top \Sigma \mathbf{j}) \\ z_2 \sum_{\forall i, j \in y, i \neq j} \frac{\mathbf{i}^\top \mathbf{j}}{\|\mathbf{i}\| \|\mathbf{j}\|} \\ \|\mathbf{u}\|^2 \\ z_3 \sum_{\forall i \in y} \|\mathbf{i}\|^2 \\ \mathbf{u} \\ z_4 \sum_{\forall i \in y} \mathbf{i} \end{bmatrix},$$

z_1, z_2, z_3, z_4 为归一化系数， Σ 为参数 σ 的对角矩阵。

Δ 为错误函数，也分为两部分，一部分是关于准确率的，一部分是关于多样性的：

$$\begin{aligned} err_{acc}(y_u, y) &= f(y_u; \mathbf{u}, \mathbf{I}_u) - f(y; \mathbf{u}, \mathbf{I}_u) \\ err_{div}(y_u, y) &= g(y_u) - g(y). \end{aligned}$$

其中 $f(y_u)$ 为准确度， $g(y_u)$ 为多样性。

通过 F-measure 确定最终的错误函数：

$$\Delta(y_u, y) = \frac{2 \times err_{acc}(y_u, y) \times err_{div}(y_u, y)}{err_{acc}(y_u, y) + err_{div}(y_u, y)}.$$

4.2. Re-Rank 方法

Re-Rank[3]方法是一种启发式的方法，它使用参数 T_R 来表示准确度与多样性的权衡， $T_R \in [T_H, T_{max}]$ ，其中 T_{max} 为 Rating 的最大值（文中定义为 5）， T_H 为设定的阈值（文中定义为 3.5）。当 T_R 接近于 T_{max} 时表示准确性较高，当 T_R 接近于 T_H 时多样性较高。Re-Rank 定义了新的项目排名方法，对于用户 i 和参数 T_R ，有 $rank_x(i, T_R)$ ：

$$rank_x(i, T_R) = \begin{cases} rank_x(i), & \text{if } R^*(u, i) \in [T_R, T_{max}], \\ \alpha_u + rank_{Standard}(i), & \text{if } R^*(u, i) \in [T_H, T_R), \end{cases}$$

where $I_u^*(T_R) = \{i \in I | R^*(u, i) \geq T_R\}$, $\alpha_u = \max_{i \in I_u^*(T_R)} rank_x(i)$.

当 T_R 接近于 T_{max} 时，倾向于使用 $rank_{standard}$ 的排序方法， $rank_{standard} = R^*(u, i)^{-1}$ ，即标准的排序方法确定的项目排名，其准确度高。

当 T_R 接近于 T_H 时，倾向于使用 $rank_x$ 的排序方法，文中定义 $rank_x(i) = R^*(u, i)$ ，即反向预测 Rating 值，其多样性高。

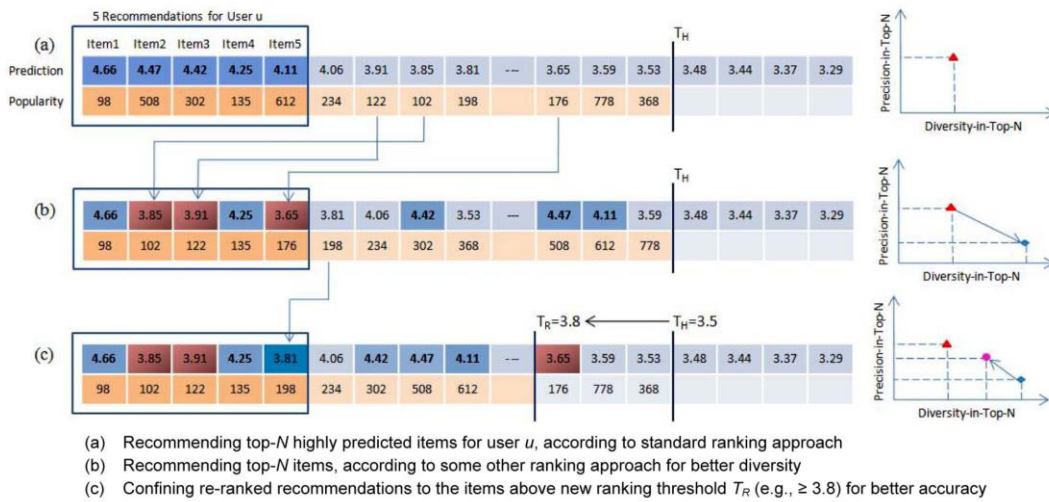


图 4-1 Re-Rank 方法改进多样性的过程

如图 4-1 所示为该 Re-Rank 方法改进多样性的过程，分为三步：第一步使用传统的排序

方法，对所有物品的 rating 值进行排序，并取前 N 个作为推荐项目。第二步使用提高多样化的排序方法，如 $rank_x(i) = R^*(u, i)$ ，进行排序。第三步，调整 T_R 的值，以改善准确率，限制推荐的项目的 Rating 都满足 $R^*(u, i) \geq T_R$ 。如图中的 Rating=3.65 比 $T_R=3.8$ 小，所以被抛弃，其后一位的项目数顺势补充为第五个推荐项目。

对于每个用户，算法复杂度为 $O(n \log n)$ ，即排序算法的复杂度。总的复杂度为 $O(mn \log n)$ ，其中 m 为用户个数， n 为项目数。

4.3. 二分图匹配方法

[2]提出了使用最大流和二分图匹配的方法提升多样性。该方法是一种改善的 Re-Rank 方法，相对于 4.2 节提出的方法，在牺牲部分效率地情况下，显著地提高了系统整体的推荐多样性(Aggregate Diversity)。

该方法的建模如图 4-2(a)所示，图的左侧每个节点代表用户，右侧每个节点代表项目，其中如果用户与项目“有关联”则他们之间连一条边，最大流量为 1。其中“有关联”可以表示成 $R^*(u, i) \geq T_H$ ， $R^*(u, i)$ 是推荐系统预测出的用户 u 和项目 i 的关联系数(Rating)， T_H 为设定的推荐阈值。设立源点到每个用户节点的边，最大流量是 N 。每个项目到汇点的边，最大流量的 1。然后使用最大流算法求解。为了提升效率，可以使用更加高效的二分图匹配算法 –Hopcroft-Karp 算法，可以在 $O(mn\sqrt{m})$ 的复杂度内求解，其中 m 是用户数目， n 是项目数目。

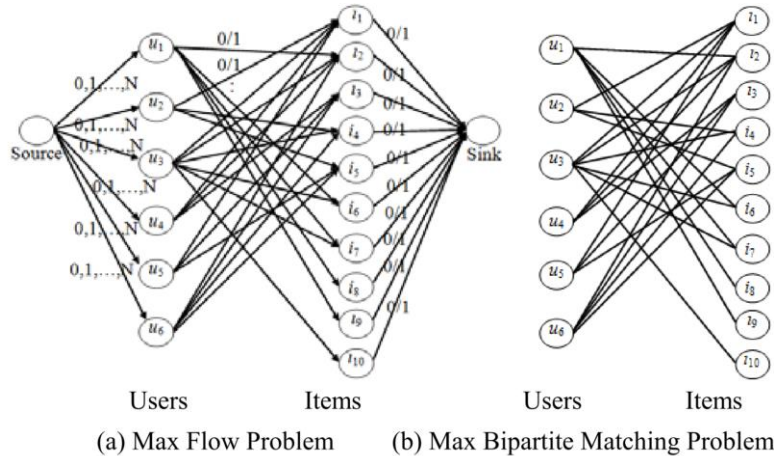


图 4-2 最大流方法的推荐模型

因为有可能给用户推荐的项目数目少于 N ，其他的部分可以通过传统的推荐方法补足，因为使用二分图算法求解出的解已经是最优的，所以补充的部分不会增加推荐的多样性。总的时间复杂度为 $O(mn \log n + mn\sqrt{m})$ ，其中 m 是用户数目， n 是项目数目。

REFERENCE

- [1] Brynjolfsson, E., Hu, Y., & Smith, M. D. (2010). Research commentary—long tails vs. superstars: The effect of information technology on product variety and sales concentration patterns. *Information*

Systems Research, 21(4), 736-747.

- [2] Adomavicius, G., & Kwon, Y. (2011, October). Maximizing aggregate recommendation diversity: A graph-theoretic approach. In *Proc. of the 1st International Workshop on Novelty and Diversity in Recommender Systems (DiveRS 2011)* (pp. 3-10).
- [3] Adomavicius, G., & Kwon, Y. (2012). Improving aggregate recommendation diversity using ranking-based techniques. *IEEE Transactions on Knowledge and Data Engineering*, 24(5), 896-911.
- [4] Cheng, P., Wang, S., Ma, J., Sun, J., & Xiong, H. (2017, April). Learning to Recommend Accurate and Diverse Items. In *Proceedings of the 26th International Conference on World Wide Web* (pp. 183-192). International World Wide Web Conferences Steering Committee.