

赞同 38

分享

阿里2024：利用社交推荐，让低活用户也能看到喜欢的内容



SmartMindAI

专注搜索、广告、推荐、大模型和人工智能最新技术，欢迎关注我

已关注

38 人赞同了该文章

Introduction

随着社交媒体的兴起，社会推荐的概念越来越受到关注，它借助社交网络优化推荐策略，特别在数据稀疏问题上有所改善。然而，对于交际行为少的“冷启动”用户，精确推荐仍是个挑战。现存模型主要基于原始社交图，但忽视了其社交关系质量对其推荐效果的影响。实研究揭示，实际社交关系存在质量问题（低价值）和数量不足的问题，这加剧了对不活跃用户的推荐难题。

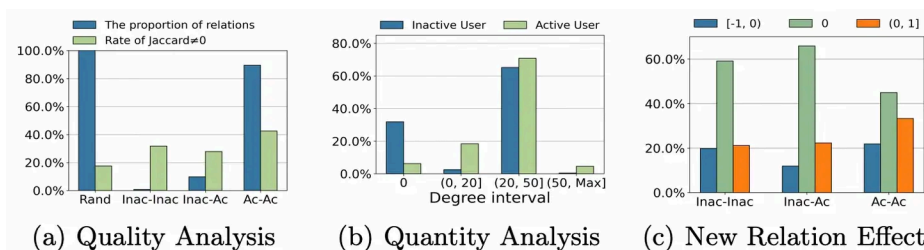


Fig. 1. The industrial observations on social relations based on Taobao.

本文主要贡献如下：

- 首次探究社交关系对社会推荐影响，我们发现在不活跃用户（低互动）群体中，社交网络的质量问题严重：低质量关系不能提供有价值信息，且数量也相对较少。这直接导致他们在获取用户和物品信息时受限，增加了社会推荐的挑战。
- 我们提出LSIR（Learning Social Graph for Inactive User Recommendation），这是一种创新方法，将图结构学习策略融入社会推荐以解决社交关系对冷启动用户推荐的挑战。通过优化用户间的社交图，LSIR通过剔除无效链接并增强有益连接，显著提高了推荐的准确性和有效性，提升了对不活跃用户的推荐服务。
- 我们运用LSIR在公开和工业数据集上对它的有效性及其适应性进行了实证测试，LSIR在处理不活跃用户推荐问题上表现出色，提升幅度达到129.58%，超越了现有最先进方法。

Industrial Observations on Social Relation

在我们的研究中，我们聚焦于社交网络对不活跃用户推荐的影响，以阿里巴巴*淘宝电商平台的数据为依据，样本涵盖了一亿多用户，其中约30%是不活跃用户，活跃度定义为过去两个月内购行为少于10条。图揭示了原始社交关系质量的普遍低下：

- 对于不活跃用户与活跃用户（Inac-Ac）的交互，以及不活跃用户之间的互动（Inac-Inac），非零的购买商品Jaccard相似度比率（衡量兴趣共享程度）均不超过50%，如图的蓝色条形图所示，这表明用户间购买偏好关联度弱。

- 这种低质量的社交网络限制了不活跃用户从他人那里获取的有效信息，对推荐系统⁺的准确性构成挑战。因此，我们提出LSIR，旨在通过优化社交图结构来提升对这类用户的推荐效果。

Preliminary

在社会推荐中，我们定义用户集合 $U = \{u_1, u_2, \dots, u_m\}$ 包含 m 个用户，每个用户 u_i 具有自己的特征向量⁺ $x_{u_i} \in \mathbb{R}^{D_u}$ 表示在原始特征空间⁺中的属性。

项目集合 $I = \{i_1, i_2, \dots, i_n\}$ 由 n 个项目组成，也有相应的特征向量 $x_{i_j} \in \mathbb{R}^{D_i}$

用户 u 对项目 i 的购买行为用集合 $I(u)$ 表示，而用户 u 与项目 i 互动的用户集合用 $U(i)$ 表示。

在社交网络中 N_u 表示用户 u 在原始社交图中的邻居集合，即与之有直接或间接联系的其他用户。为了处理不活跃用户，我们引入了LSIR (Learning Social Graph for Inactive User Recommendation) 模型，它通过分析和优化 N_u 来提升对这类用户个性化的推荐。通过这种方式，LSIR试图弥合社交连接不足带来的信息缺口。

在特定的交易时间窗口下，我们区分用户两类：不活跃用户 u_- 和活跃用户 u_+ 。对于不活跃用户，我们依据其购买行为，如果购物记录的项目数量 $|I(u)|$ 少于预设阈值 ϵ (根据业务规则设置)，则将其归为此类。活跃用户 u_+ 则是指那些购买行为满足条件的用户。这样，用户集合 U 就包含了这两个子集，即 $U = \{u_+, u_-\}$ 。

在LSIR模型中，我们将关注如何针对这两类用户提供不同的推荐策略，利用社交网络的信息优化对不活跃用户的推荐，弥补他们社交连接相对较少带来的信息缺乏。

通过改进社交网络结构，LSIR模型进行增强型社交推荐。对于用户 u ，无论是活跃的 u_+ 还是不活跃的 u_- ，其购买商品的可能性计算公式是：

$$P(i|u) = \frac{1}{|N(u)| + \alpha}$$

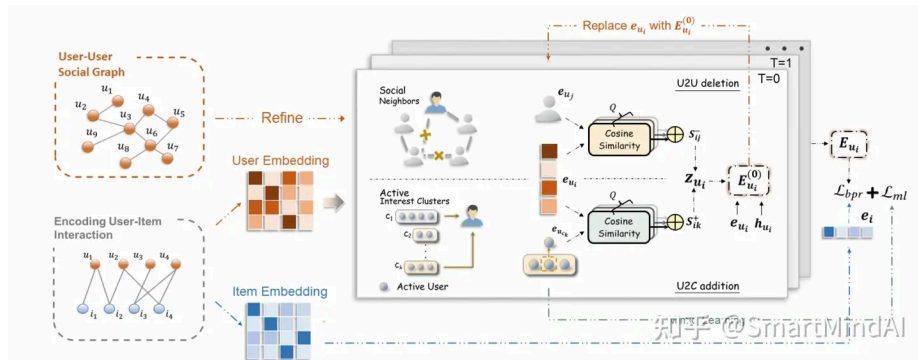
其中： $-N(u)$ 涵盖了用户 u 的所有社交联系，包括活跃和不活跃的邻居。 $-\alpha$ 是个调整系数，代表从活跃用户那里获得的额外信息价值，以提升推荐的准确性。

公式强调了社交网络中用户自身行为与邻居潜在兴趣的结合，通过优化社交关系，能更精准地进行个性化推荐，弥补不活跃用户因社交连接不足带来的信息缺失。

$$\hat{r}_{u_i} = \mathcal{F}(i \in I|u, I(u), N'_u)$$

本研究聚焦于提升对不活跃用户 (u_-) 的社交推荐，通过深化挖掘他们在原始社交关系中的潜在关联，构建了细化社交邻近集合 N'_u 。推荐器 \mathcal{F} 以此为基础，预测这类用户潜在的兴趣。

The Proposed Model



我们详述了名为LSIR的模型，其设计旨在优化针对不活跃用户的社交推荐。模型结构包括：用户-物品交互的特征建模、对用户-用户社交关系的精细处理（通过 N'_u ），以及模仿学习三个核心步骤。通过这些，LSIR通过深化理解不活跃用户的社交网络，提升推荐的有效性和针对性。

首先，我们通过用户-物品交互来刻画用户和项目的特性，将**协同过滤⁺**的影响力融入到嵌入中。由于用户和**项目特征⁺**的结构不同，我们通过投影操作将它们映射到一个共享的潜在**向量空间⁺**。对于用户 u ，我们利用**多层感知器⁺**（MLP，参数 θ_u ）将其原始特征向量 x_u 转化为该空间中的嵌入表示：

$$h_u = \text{MLP}(x_u; \theta_u)$$

这个过程去除了特征的异质性，有利于发现用户和项目之间潜在的关联。

$$h_u = W_U^{(1)} \sigma \left(W_U^{(0)} x_u + b_U^{(0)} \right) + b_U^{(1)}$$

其中，用户嵌入 h_u 由多层感知器（MLP，参数 θ_u ）计算得出，它将用户 u 的原始特征 x_u 转换到一个 d 维的共性空间。PReLU**激活函数⁺**用于增强非线性表达。项目嵌入 h_i 遵循相似的计算路径，通过MLP以 θ_i 作为参数对项目特征 x_i 进行处理，生成在相同空间内的表示 h_i 。这种统一的表示使得用户和项目能够在这个空间中比较和分析潜在的兴趣关联，有利于更精准的社交推荐。

$$h_i = W_I^{(1)} \sigma \left(W_I^{(0)} x_i + b_I^{(0)} \right) + b_I^{(1)}$$

我们通过整合用户实际购买的项目和社交网络中的项目，来增强用户和项目的嵌入。尽管考虑了潜在的转换和非线性影响，我们采取了简单**加权求和⁺**的策略，以保持模型的清晰性。用户嵌入 h_u 通过累加已购项目 $\beta_i \cdot x_i$ 的特征和邻居用户 $\gamma_v \cdot h_v$ 的嵌入得到，而项目嵌入 h_i 直接使用 $\beta_i \cdot x_i$ 。这种做法既保持了模型的简洁性，又在某种程度上确保了信息的多样性和推荐的精确性。

$$e_u^{(k)} = \sum_{i \in I(u)} \frac{e_i^{(k-1)}}{\sqrt{|I(u)|} \sqrt{|I(i)|}}, \quad e_i^{(k)} = \sum_{u \in U(i)} \frac{e_u^{(k-1)}}{\sqrt{|U(i)|} \sqrt{|I(u)|}}$$

在第 k 层，我们分别得到用户 $e_u^{(k)}$ 和项目 $e_i^{(k)}$ 的嵌入，它们起始于底层的 h_u 和 h_i 。初始化时 $e_u^{(0)} = h_u$ ， $e_i^{(0)} = h_i$ 。通过经过 K 轮传播迭代，我们通过‘Combine’函数将所有层次的嵌入融合，以生成用户和项目的最终交互表示：

$$e_u = \text{Combine}(\{e_u^{(k)}\}_{k=1}^K), \quad \text{and} \quad e_i = \text{Combine}(\{e_i^{(k)}\}_{k=1}^K)$$

此过程强化了基于交互信息的用户和项目个体特征，以优化推荐。‘Combine’函数的设计决定了如何处理和整合这些多层次的信息，以提升推荐质量。

$$e_u = \frac{1}{K} \sum_{k=0}^K e_u^{(k)}, \quad e_i = \frac{1}{K} \sum_{k=0}^K e_i^{(k)}$$

Graph Structure Learning on Social Graph

在编码用户-物品交互后，对不活跃用户 u_- ，我们通过社交结构优化来弥补其消费历史信息不足。首先，进行U2U删除，去除无效的社交关系，通过计算用户 u_i 和邻居用户 j 的嵌入相似性 S_{ij-} ，保留有效边，其中

$$S_{ij-} = \text{similarity}(e_i, e_j) \times w_{ij}$$

w_{ij} 是边权重。接着，引入U2C添加，考虑用户对中心节点的社交行为，通过计算用户与中心节点的相似度来扩展其社交网络，以获得更全面的用户视角。这样，通过这两个阶段的组合，构建了一个更精细的社交图，以提升对不活跃用户的个性化推荐。

$$S_{ij-} = \cos(W_n \cdot e_{u_i}, W_n \cdot e_{u_j})$$

在这个上下文中， W_n 是一个动态学习的投影器，增强了测量的适应性。为处理用户间多元相似性，我们对公式进行了扩展，形成了多头版本的 $S_{ij-}^{(h)}$ ：

$$S_{ij-}^{(h)} = \text{similarity}_h(e_i, e_j) \times w_{ij}^{(h)} \quad \text{for } h = 1, 2, \dots, H$$

还通过不同头的相似性来深化对用户理解。

$$s_{ij-}^{(q)} = \cos(\mathbf{W}_n^{(q)} \cdot \mathbf{e}_{u_i}, \mathbf{W}_n^{(q)} \cdot \mathbf{e}_{u_j}), \quad S_{ij-} = \frac{1}{Q} \sum_{q=1}^Q s_{ij-}^{(q)}$$

给定用户 u_i 的邻居集合 $\mathcal{N}(u_i)$ ，删除率计算公式为：

$$\text{DeleteRatio}_i = \tanh\left(\sum_{j \in \mathcal{N}(u_i)} S_{ij-}^{(h)}\right)$$

其中 $S_{ij-}^{(h)}$ 是第 h 个头的相似度指标，通过 \tanh 函数对各个邻居的相似度分数加权求和。这个设计考虑了用户直接的购买行为以及社交网络中潜在的关联性，从而综合评估并调整对不活跃用户的推荐策略。

$$p_{del}(|I(u_i)|) = \frac{\exp(|I(u_i)|/r_1) - \exp(-|I(u_i)|/r_1)}{\exp(|I(u_i)|/r_1) + \exp(-|I(u_i)|/r_1)}$$

在U2C添加阶段，我们强化了对中心节点⁺的考量，弥补了仅依赖U2U删除的不足。首先，通过计算用户 u_i 与中心节点的相似度 Similarity_{iu} 来获取额外信息。然后，基于此相似度和删除率 DeleteRatio_i ，我们筛选出与中心节点关联最紧密的邻居，即 TopNeighbors_{iu} ，数量为 $1 - p_{del}(|I(u_i)|)$ ，其中 $|I(u_i)|$ 是用户 u_i 的互动次数或其他相关指标。这样 $N_{u_i}^C$ 成为了用户 u_i 的新社交网络，用于更精确地为不活跃用户 u_- 定制推荐。

[U2C Addition]

受观察3启示，我们意识到直接将所有不活跃用户连接到活跃用户的策略效率不高且可能导致过密的社交图。为此，我们提出U2C（用户到簇）添加策略。

首先，通过挖掘活跃兴趣簇的过程：

1. **聚类⁺活动用户**：对活跃用户进行聚类，根据共同兴趣或行为特征将他们分组。
2. **选择簇内代表**：在每个簇中，选择最具影响力的活跃用户作为连接的中心，通常是意见领袖或活跃度高的用户。
3. **连接到簇**：对不活跃用户，根据其兴趣匹配程度或与活跃兴趣簇的相关性，决定连接与否。

这样，我们既能保持社交网络的稀疏性，减少存储需求，又能利用活跃用户的信息帮助不活跃用户发现潜在社交网络，同时保证推荐的有效性和可靠性。

- 通过对物品的特征进行聚类，我们获得了 \mathcal{C} 这个由 l 个子集组成的集合，每个子集 c_i 具体表示为

$$i_1^{c_i}, i_2^{c_i}, \dots, i_{|c_i|}^{c_i}$$

这些代表了不同的兴趣类别。

- 对于活跃用户 u_+ ，我们通过计算其兴趣集合 $I(u_+)$ 与聚类中每个类别 c_i 的Jaccard相似度 $\mathcal{J}(I(u_+), c_i)$ 来确定其最相关兴趣领域。Jaccard相似度是衡量交集与并集比例的指标，反映了兴趣的重合程度。我们通过遍历所有类别，找到与 $I(u_+)$ 最相似的那个，即定义为最佳匹配簇 $\mathcal{C}(u_+)$ ：

$$\mathcal{C}(u_+) = \max_{c_i} \mathcal{J}(I(u_+), c_i)$$

这个簇指向了用户 u_+ 最热衷或者最感兴趣的内容类别。在推荐时，我们依据这个信息，向用户 u_+ 推荐与 $\mathcal{C}(u_+)$ 相匹配的项目，以提供个性化推荐服务。

$$\mathcal{C}(u_+) = \max_{c_i} \mathcal{J}(I(u_+), c_i)$$

- 对于兴趣簇 c_i ，我们构造集合 $\mathcal{U}(c_i)$ ，包括所有属于 c_i 的活跃用户，记为

$$\mathcal{U}(c_i) = \{u_+ | \mathcal{C}(u_+) = c_i\}$$

在该集合中，我们找出与 c_i 最相似的用户，定义为用户 u_{c_i} ，通过优化Jaccard相似度：

u_{c_i} 是 c_i 中的代表性用户，其兴趣高度一致。在推荐策略中，优先考虑向 u_{c_i} 推荐与 c_i 相关的内容，以提升推荐的准确性和个性化体验。

- 通过收集并集中 $U_C = \{u_{c_1}, u_{c_2}, \dots, u_{c_l}\}$

这些代表用户，我们形成了每个兴趣簇的实际触点。借助这些用户的行为和喜好数据，我们可以深入理解和预测他们对各自簇内物品的反应。这样 U_C 成为了连接兴趣类别和用户行为的桥梁，从而为每个簇提供个性化的推荐服务。

在这个公式中 S_{ik+} 表示用户 u_i 和兴趣簇 u_{c_k} 之间在所有 h 个头上的交互得分，通过 Tanh 函数计算得到：

$$S_{ik+} = \tanh\left(\sum_{h=1}^H S_{ij-}^{(h)} \cdot S_{ck-}^{(h)}\right)$$

其中 H 是头的数量 $S_{ij-}^{(h)}$ 和 $S_{ck-}^{(h)}$ 是用户头 h 下用户 i 和簇 k 的相似度分数。这种方法考虑了用户不仅在直接兴趣上的相似，还考察他们在不同社交维度下的互动，从而更精确地判断连接这两者间的概率。

$$s_{ik+}^{(q)} = \cos(W_n^{(q)} \cdot e_{u_i}, W_c^{(q)} \cdot e_{u_{c_k}}), \quad S_{ik+} = \frac{1}{Q} \sum_{q=1}^Q s_{ik+}^{(q)}$$

公式 $p_{add,i}$ 通过 `adaptively_adjust` 函数，动态调整了向用户 u_i 添加锚用户 (u_{c_k}) 的比例。这个过程考虑了用户 u_i 与簇内所有用户 ($U(c_i)$) 基于权重矩阵 $W_c^{(q)}$ 的相似度，既利用了直接的相似性，又融入了社交网络中潜在的关联信息。这样的策略确保了推荐更具个性化，能针对不活跃用户提供有针对性的补充建议。

$$p_{add}(|I(u_i)|) = \frac{1}{1 + \exp(|I(u_i)|/r_2)}$$

通过函数 `fuse` z_{u_i} 是通过整合用户 i 在社交网络 $N_{u_i}^U$ 和兴趣网络 $N_{u_i}^C$ 的信息生成的。这种综合表示融合了用户的社交行为和兴趣属性，为后续的个性化推荐或聚类分析⁺提供了全面的用户画像。

$$z_{u_i} = \alpha \left(\sum_{j \in N_{u_i}^U} S_{ij-} \cdot e_{u_j} + \sum_{k \in N_{u_i}^C} S_{ik+} \cdot e_{u_{c_k}} \right) + (1 - \alpha) e_{u_i}$$

通过 `skip-connection`，初始用户表示 $E_u^{(0)}$ 结合了原始特征 E_{ori} ，来自层次化 GNN (LGCN) 的第二层信息 $E_{LGCN}^{(2)}$ ，以及用户间交互的直接信息 E_{UU} 。这个过程允许原始特征不被忽视，同时利用社交网络结构和交互的多层信息，生成一个既全面又复杂，能体现用户多样性的初始嵌入，为后续的优化和学习提供起点。

$$E_u^{(0)} = W_f \cdot [h_u || e_u || z_u] + b_f$$

在这个公式中， L_{BPR} 是基于 BPR (Bayesian Personalized Ranking) 的目标函数⁺。 \mathcal{D} 是用于训练的数据集，其中 (u, i, j) 表示用户 u 对项目 (i, j) 的实际反馈（1 代表喜欢，0 代表不喜欢）。sigmoid 函数 $\sigma(x)$ 用来对得分进行非线性转换，使输出在 0 到 1 之间。

模型通过计算用户 u 对项目 j 的评分 $-E_u^T E_j$ (负向交互得分，因为 BPR 通常处理正面评价的学习)，然后与用户真实反馈 y_{ui} 比较，来评估预测的合理性。如果用户实际喜欢项目，目标是提高预测，反之则降低。损失函数⁺通过优化来最大化用户对喜欢项目 (正向样本) 的预测，同时最小化对不喜欢项目 (负向样本) 的预测，以此实现个性化排序。通过最小化 L_{BPR} ，模型能够学习用户的真实兴趣，同时考虑到社交影响，为用户推荐具有社交价值的相关项目。

$$L_{bpr} = \sum_{(u, i, i') \in \mathcal{O}} -\log \sigma(\hat{r}_{ui} - \hat{r}_{ui'}) + \lambda ||\Theta||^2$$

在 LSIR (Social Influence-based Item Recommendation, 基于社交影响的物品推荐) 模型中，参数集合 Θ 控制着模型的行为。预测得分 $\hat{r}_{ui} = E_u^T e_i$

是通过计算用户 u 对项目 i 的特征向量乘积得出的。sigmoid 函数 σ 用于将得分转化为概率。

喜欢项目（正样本）的预测，同时最小化对不喜欢项目的预测，以实现个性化的排序。

通过这种方式，LSIR不仅能捕捉到用户个体的偏好，还能考虑社交环境对用户行为的影响，从而提供基于社交影响的推荐服务。

Mimic Learning

在处理未活跃用户匹配问题时，尽管我们优化了用户-用户社交图，但U2C部分仍面临挑战。原有方法仅用来自 \mathbf{e}_u 作为相似性指标，这限制了对未活跃用户潜在聚类的理解。为解决这一不足，我们引入模仿学习，生成伪未活跃用户嵌入 $\hat{\mathbf{e}}_{u_+}$ ，目标是使LSIR能准确匹配。借鉴MixUp在监督学习中的策略，我们设计了Inactive Mixture方法。在每次批次训练中，对每个活跃用户 u_+ ，我们随机选择 η 个未活跃用户并按权重 β 与之混合，形成混合用户表示 $\tilde{\mathbf{u}}$ ：

$$\tilde{\mathbf{u}} = \beta \cdot \mathbf{u}_+ + (1 - \beta) \cdot \mathbf{u}_{inactive}$$

通过这种混合，LSIR不仅能学习主动用户的特性，还能间接接触并学习未活跃用户的特征，从而提升对聚类的识别和匹配能力。

$$\hat{\mathbf{e}}_{u_+} = \beta \mathbf{e}_{u_+} + (1 - \beta) \mathbf{e}_{u_-}$$

在生成伪未活跃用户嵌入 $\hat{\mathbf{e}}_{u_+}$ 后，我们引导编码器使其靠近目标聚类 $\mathcal{C}(u_+)$ 的中心。为此，我们使用对比式InfoNCE损失来优化模型。该损失定义如下：

$$L_{InfoNCE} = -\log \frac{\exp(\hat{\mathbf{e}}_{u_+}^\top \mathbf{z}_+)}{\sum_{k=1}^K \exp(\hat{\mathbf{e}}_{u_+}^\top \mathbf{z}_k) + \sum_{k=1, k \neq u_+}^{|\mathcal{C}(u_+)|} \exp(\hat{\mathbf{e}}_{u_+}^\top \mathbf{z}_k)}$$

其中 \mathbf{z}_+ 是 u_+ 所在聚类的代表 \mathbf{z}_k 代表其他用户 K 是聚类数目。InfoNCE旨在使 $\hat{\mathbf{e}}_{u_+}$ 与同类用户更相似，同时与非同类用户相异，从而精确识别其实际所属的聚类。通过优化这个损失，LSIR能更好地理解 and 定位与 u_+ 相关联的聚类。

$$\mathcal{L}_{ml} = -\sum_{u_+} \log \frac{\exp(\cos(\hat{\mathbf{e}}_{u_+}, \mathbf{e}_{uc_j})/\tau)}{\sum_{k \neq j} \exp(\cos(\hat{\mathbf{e}}_{u_+}, \mathbf{e}_{uc_k})/\tau)}$$

在给定的情境中，当 $\mathcal{C}(u_+) = c_j$ 时，我们通过调整 $\hat{\mathbf{e}}_{u_+}$ 来使其靠近聚类 c_j ，同时避免与其他聚类混淆。这通过模仿损失实现，它促使伪未活跃用户嵌入尽量逼近 c_j 且远离其他聚类。模仿损失结合了BPR (Bayesian Personalized Ranking) 损失，后者关注个性化排序的学习。

整体损失函数为：

$$L_{total} = L_{BPR} + \lambda L_{InfoNCE}$$

这里 L_{BPR} 是用于优化用户兴趣预测的，而 λ 是一个权重参数，它平衡了模仿损失和BPR损失的相对重要性。 L_{total} 的目标是提升推荐的准确性，同时确保模型能准确地识别不同用户的社交影响模式，从而提供更精准的社交影响下的推荐。

$$\mathcal{L}_\Omega = \mathcal{L}_{bpr} + \xi \cdot \mathcal{L}_{ml}$$

利用 ξ 作为组合系数，我们利用反向传播和SGD等优化工具对LSIR模型进行迭代优化。通过不断减小损失函数 L_{total} ，该过程旨在最小化误差，以提升用户行为预测的准确性以及增强聚类识别。这样，模型不仅能更好地理解用户偏好，还能区分不同社交影响情境，从而提供更精确的推荐。

Experimental Setup

实验数据集：我们在Flickr和Yelp两个公开数据集，以及AliData 1和AliData 2两个实际工业数据集上进行实验。

Dataset	#User	#Inactive User	#User-User	#Item	#User-Item	Inactive Cutoff
Flickr	8,358	2,386	187,273	82,120	327,815	≤ 2
Yelp	17,237	5,445	143,765	38,342	204,448	≤ 3
AliData I	40,428	12,071	323,608	115,502	1,016,122	≤ 7
AliData II	103,440	23,335	984,555	2,417,818	12,095,268	≤ 20

AliData 1针对特定商品，用户行为稀疏且集中，AliData 2则包含更多商品种类和多样客户，交互更频繁。为了保证研究的公正性，我们详细列出了每个数据集在处理未活跃用户冷启动时采用的标准。

Overall Recommendation Performance(RQ1)

在这个部分，实验旨在证明LSIR在推荐任务中的效能。我们通过记录未活跃用户和整体用户接受Top-10/20推荐的情况，并在表格中详细报告了针对活跃用户NDCG@10的具体数值。通过这些数据，我们能够评估LSIR在提升用户相关性和兴趣匹配方面的表现。基于这些结果，我们可以得出结论，即LSIR确实能够在社会推荐系统中展现出优越的性能。

Datasets	Metric	Top-K	BPR	LightGCN	SimGCL	SBPR	DiffNet++	S ² -MHCN	SEPT	MeLU	TaNP	LSIR	Improv.
Flickr	NDCG	10	0.0503	0.0797	0.0648	0.0659	<u>0.1251</u>	0.0990	0.0455	0.0660	0.0781	0.2872	129.58%
		20	0.0630	0.1042	0.0801	0.0808	<u>0.1470</u>	0.1137	0.0561	0.0994	0.1068	0.3253	121.29%
	HR	10	0.0731	0.1350	0.0951	0.0997	<u>0.1752</u>	0.1413	0.0675	0.1154	0.1391	0.3927	124.14%
		20	0.1112	0.2094	0.1444	0.1465	<u>0.2462</u>	0.2000	0.1028	0.2045	0.2176	0.5164	109.75%
	PR	10	0.0146	0.0270	0.0190	0.0199	<u>0.0350</u>	0.0283	0.0135	0.0231	0.0273	0.0785	124.29%
		20	0.0111	0.0209	0.0144	0.0147	<u>0.0246</u>	0.0200	0.0103	0.0204	0.0217	0.0516	109.76%
Yelp	NDCG	10	0.1437	0.1636	0.1847	0.1678	<u>0.2261</u>	0.1845	0.1480	0.1485	0.1562	0.2669	18.05%
		20	0.1725	0.1945	0.2152	0.2002	<u>0.2605</u>	0.2151	0.1834	0.1969	0.2126	0.3038	16.62%
	HR	10	0.2185	0.2439	0.2709	0.2525	<u>0.3288</u>	0.2742	0.2274	0.2492	0.2642	0.3804	15.69%
		20	0.3103	0.3409	0.3720	0.3555	<u>0.4495</u>	0.3754	0.3357	0.3971	0.4085	0.4997	11.17%
	PR	10	0.0437	0.0487	0.0542	0.0505	<u>0.0658</u>	0.0548	0.0455	0.0498	0.0531	0.0761	15.65%
		20	0.0310	0.0304	0.0372	0.0356	<u>0.0450</u>	0.0375	0.0336	0.0397	0.0408	0.0500	11.11%
AliData I	NDCG	10	0.1634	0.1629	0.1815	0.1644	0.1793	<u>0.1817</u>	0.1650	0.1648	0.1696	0.2018	11.06%
		20	0.1817	0.1766	0.1931	0.1812	0.1990	<u>0.2019</u>	0.1798	0.1784	0.1810	0.2214	9.66%
	HR	10	0.2201	0.2129	0.2224	0.2169	<u>0.2515</u>	0.2509	0.2122	0.2219	0.2223	0.2748	9.26%
		20	0.2859	0.2622	0.2638	0.2791	<u>0.3223</u>	0.3227	0.2655	0.2660	0.2724	0.3530	9.39%
	PR	10	0.0330	0.0301	0.0333	0.0325	<u>0.0377</u>	0.0376	0.0318	0.0317	0.0324	0.0412	9.28%
		20	0.0214	0.0187	0.0198	0.0209	<u>0.0241</u>	0.0242	0.0199	0.0202	0.0204	0.0265	9.50%
AliData II	NDCG	10	0.1126	<u>0.1812</u>	0.1757	0.1327	-	-	0.1796	0.1777	0.1808	0.2060	13.69%
		20	0.1251	<u>0.1903</u>	0.1821	0.1414	-	-	0.1897	0.1840	0.1866	0.2208	16.03%
	HR	10	0.1191	0.1638	0.1554	0.1368	-	-	0.1688	0.1610	0.1612	0.2092	23.93%
		20	0.1575	0.1955	0.1787	0.1664	-	-	<u>0.2037</u>	0.1812	0.1855	0.2663	30.73%
	PR	10	0.0329	0.0452	0.0430	0.0378	-	-	0.0467	0.0455	0.0444	0.0541	24.27%
		20	0.0218	0.0270	0.0247	0.0230	-	-	<u>0.0282</u>	0.0260	0.0262	0.0368	30.50%

- 对未活跃用户的行为处理上，LSIR显示出明显优势。具体体现在Flickr数据集上，LSIR的NDCG指标增长了129.58%，而在AliData 2中，HR (Hit Rate) 提升了30.73%。这强有力地证实了LSIR的有效性和独特性。尤其值得注意的是，即使在处理大型的AliData 2时，LSIR仍能保持良好的性能，而DiffNet++和S²-MHCN则面临挑战，这突显了LSIR在处理大规模数据和高并发+情况下的优越性，体现其强大的可扩展性+。

Table 3. The NDCG@10 performances on active and overall users got from social recommenders. (bold: best; underline: runner-up; '-': Out-Of-Memory on 32GB GPU)

Datasets	Flickr		Yelp		AliData I		AliData II	
Methods	Active	Overall	Active	Overall	Active	Overall	Active	Overall
SBPR	0.0728	0.0709	0.1687	0.1685	0.2128	0.1984	0.2964	0.2595
DiffNet++	<u>0.1011</u>	<u>0.1080</u>	<u>0.2193</u>	<u>0.2213</u>	0.2138	0.2035	-	-
S ² -MHCN	0.0918	0.0939	0.1991	0.1950	0.2274	0.2138	-	-
SEPT	0.0593	0.0554	0.1701	0.1638	0.2362	<u>0.2150</u>	0.3415	0.3050
LSIR	0.1010	0.1542	0.2317	0.2418	<u>0.2281</u>	0.2203	0.3335	<u>0.3048</u>

- 在Flickr的实验结果中，LSIR的表现尤为显著。这种现象可解释为用户平均社交邻居数量的差异。根据表中的数据，Flickr的数值 (22.41) 远高于其他平台 (如AliData2的9.52、Yelp的8.34和AliData1的8.00)，这与模型性能提升的顺序相吻合。这意味着在更丰富的社交网络环境下，LSIR能更好地利用这些连接，从而增强了其推荐效果。因此，较高的社交邻域数量为LSIR提供了更强的性能提升空间。
- 冷启动推荐器在所有测试条件下表现相对较弱，原因有两个。首先，它们缺乏利用社交网络信息的能力，因为它们是基于MAML而非GNN架构，没有充分利用消息传递的优势。其次，它们无法从用户间的社交关系中提取有价值的知识，仅依据用户直接的交互数据来做推荐，这是它们表现不佳的一个主要原因。
- LSIR在活跃用户和一般用户群体的表现都保持了竞争力。图表展示于下，它显示在学术和工业数据集的各类用户场景中，无论是针对活跃用户还是总体用户，LSIR都能提供最优或接近最优的表现。这说明，我们的改进措施并未损害对活跃和全部用户的推荐效果。实际上，LSIR不仅在处理

原文《Learning Social Graph for Inactive User Recommendation》

编辑于 2024-05-31 07:20 · IP 属地北京

推荐系统 社交推荐 新用户



理性发言，友善互动

2 条评论

默认 最新



arachis

有开源代码吗？

07-17 · 北京

回复 喜欢



我是一根虫

低活跃用户其实对于展示层面没有那么大的价值🐛

05-29 · 广东

回复 喜欢

推荐阅读

首次公开 | 淘系技术总监马廐谈淘系用户增长

导读：近年来，关于用户流量的瓶颈让很多企业感到焦虑不安，互联网用户整体增速放缓，用户规模趋于饱和。同时，竞争个体成倍增长，流量资源争夺越发激烈，流量成本日趋高涨。企业面对巨大的...

阿里云云栖... 发表于程序员进修...

用户超长行为历史下CTR预测 (阿里 at KDD2019

Practice on Long Sequential User Behavior Modeling for Click-Through Rate Prediction (阿里 at KDD2019 5月 通过系统设计解决用户超长行为历史下CTR建模与在线预测性能瓶颈，效果...

cutef... 发表于推荐系统分...



阿里妈妈牟娜：定向广告新一代点击率预估主模型——深度...

DataF... 发表于人工智能算...



阿里榜单&常见标识

柒007