

其中, $w = \{w_1, w_2, w_3\}$ 为 3 种算法融合的权重, 使用逻辑回归得到权重值. 本文将该问题看做一个分类问题, 类别为用户是否下载了某一个应用. 我们在用户下载记录中抽取部分数据作为正例, 将用户浏览了而没有下载的应用找到, 剔除用户安装了的应用, 抽取部分应用数据作为负例. 为了增加推荐结果的多样性, 我们在正例选择时剔除热门应用. Y 用来表示用户是否会下载该应用, $Y=1$ 表示用户下载了某 app, 也即用户对某一个 app 感兴趣; $Y=0$ 表示用户浏览了某 app 却没有下载, 也就是表示用户对该 app 没有兴趣. h_w 表示预测函数, 则对于 m 个样本, 最终的损失函数 $J(w)$ 见公式(14).

$$h_w(v_i) = \frac{1}{1 + e^{-s_v^{hybrid}}} \quad (13)$$

$$J(w) = -\frac{1}{m} \sum_{i=1}^m [y^i \log h_w(v_i) + (1 - y^i) \log(1 - h_w(v_i))] \quad (14)$$

为了使损失函数最小, 采用梯度下降法求解即可得到算法融合的权重. 在梯度下降求解后, 得到 $w = \{w_1, w_2, w_3\}$ 以及 w_0 , 可以计算每个用户的推荐候选集中不同 app 的最终得分, 即 s_v^{hybrid} . 根据该值选取 Top-10 的应用作为用户最终的推荐列表. 对于模型的评判, 我们使用 AUC 值来度量分类模型的好坏, 进而判断融合权重的好坏.

综合推荐候选集的生成和个性化推荐列表生成过程, 将混合推荐算法 Hybrid_Rec 的主要步骤表示如下.

算法. Hybrid_Rec.

输入: D_1 : 用户下载行为数据集;

D_2 : 逻辑回归训练数据集;

输出: 每个用户 top 10 推荐列表.

主要步骤:

1. $result_1 = Item_CF(D_1)$;
2. $result_2 = LDA_MF(D_1)$;
3. $result_3 = LDA_CF(D_1)$;
4. Model $H = regression(D_2)$
5. For each user
6. $R_{item} = \text{Top 50 apps selected based on } result_1$;
7. $R_{LDAMF} = \text{Top 50 apps selected based on } result_2$;
8. $R_{LDACF} = \text{Top 50 apps selected based on } result_3$;
9. For each app $v \in R_{item} \cup R_{LDAMF} \cup R_{LDACF}$
10. Calculate s_v^{hybrid} based on model H
11. Select Top-10 apps with high s_v^{hybrid}

4 实验评估与分析

本文的数据来源于某互联网公司手机助手平台中应用推荐的实际数据, 该数据包括用户下载、安装应用的日志信息以及应用的标签信息. 手机助手的应用有 100 万左右, 用户数据有十几亿. 我们选取 100 万活跃用户作为实验对象, 为用户做个性化推荐, 给出 Top-10 的推荐列表, 实验均在分布式平台 Hadoop 上运行.

在 100 万活跃用户中, 我们将用户在 2015 年 5 月的下载行为作为一个数据集, 2015 年 5 月 25 日~5 月 31 日的下载行为作为一周数据以及 2015 年 6 月 1 日的下载行为作为一天数据. 使用 2015 年 6 月 1 日~6 月 3 日这 3 天的下载数据作为评测数据, 在评测时, 我们使用相同的结果每天评测一次, 将 3 天的评测结果的均值作为最终评测结果. 见表 2, 100 万活跃用户一个月平均下载为 20 个应用, 一周平均下载 7 个应用, 一天平均下载 3 个应用.

分析中发现, 有的用户一天下载应用数超过 1 000 个应用. 我们将这种用户判定为刷量用户, 将刷量用户剔除, 不做考虑.

Table 3 User download data

表 3 用户下载数据概况

数据集	用户数	平均下载量	下载的不同应用数
一月数据	749 191	20	199 597
一周数据	509 080	7	115 356
一天数据	173 873	3	44 752

4.1 实验评测指标

4.1.1 推荐准确性评测

在推荐准确性评测方面,我们采用两方面的指标.

- 面向用户的命中准确率,用 P_{hit_user} 表示.将推荐结果与用户一天中实际下载结果进行对比,推荐的应用中只要有用户下载的应用,则认为推荐命中了该用户, N_{hit}^u 为推荐结果命中的用户数量, P_{hit_user} 则为命中的用户占这一天下载用户的比例.公式(15)为 P_{hit_user} 的计算方法.

$$P_{hit_user} = \frac{N_{hit}^u}{N_{download}^u} \quad (15)$$

其中, $N_{download}^u$ 表示一天内所有下载了应用的用户数量.

- 基于应用来考虑的, P_{hit_app} 为推荐列表中被用户下载的应用数目占有用户实际下载应用的比例,公式(16)为 P_{hit_app} 的计算方法.

$$P_{hit_app} = \frac{N_{hit}^v}{N_{download}^v} \quad (16)$$

其中, N_{hit}^v 表示推荐列表中被用户下载的 app 数目,即推荐准确的 app 数目; $N_{download}^v$ 表示一天中用户下载的所有 app 数目.因为用户下载应用的随机性加上有些用户在评测当天没有下载行为,所以在评测数据选择方面,我们以 3 天的数据(2015 年 6 月 1 日~6 月 3 日)分别评测各种算法的推荐结果,使用 3 日评测结果的均值作为最后的准确率结果.

4.1.2 推荐多样性评测

高准确率一直是推荐系统追求的标准,但是推荐准确率高而多样性低的推荐结果并不利于推荐系统的长期发展.推荐多样性的评测包括多个方面:一是推荐系统发掘长尾应用的能力,即对于整个推荐结果,评测能够展示的不同应用数目以及在推荐结果中推荐准确的不同应用数目;二是推荐列表的多样性,该多样性描述了推荐列表中物品两两之间的不相似程度.

对于推荐系统发掘长尾应用的能力,我们使用信息熵来评测.我们认为:如果所有的应用都能出现在推荐列表中,而且出现的次数差不多,那么推荐系统就有很强的发掘长尾应用能力.信息熵能够很好地表示推荐系统发掘长尾应用的两个方面:一是在推荐次数差不多的情况下,推荐列表中出现的不同应用数越多信息熵越大;二是在推荐应用数一定的情况下,推荐次数的分布越均匀信息熵越大.我们沿用文献[13]中物品流行度的概念,将推荐列表中不同应用出现次数称为该应用在推荐结果中的流行度,则信息熵计算见公式(17).

$$H = -\sum_{i=1}^n p(i) \log p(i) \quad (17)$$

其中, $p(i)$ 由应用 i 的流行度除以所有应用流行度之和计算.应用的流行度分布越平均,则信息熵越大,说明推荐系统覆盖率越高.所以信息熵越大,表示推荐系统发掘长尾能力越好.另外,为了将推荐的准确性也一定程度地在多样性度量中加以体现,除了计算推荐结果中能够展示的所有不同应用的熵,记为 $H(all\ app)$ 之外,我们还计算了推荐列表中推荐准确的不同应用的熵,记为 $H(hit\ app)$.

针对用户推荐列表的多样性,我们采用公式(18)计算每个用户 u 推荐列表 $R(u)$ 中两两 app 之间的不相似程度,公式(19)求得所有用户推荐列表不相似程度的均值.

$$Diversity(R(u)) = 1 - \frac{\sum_{i,j \in R(u), i \neq j} sim(i,j)}{\frac{1}{2} |R(u)| (|R(u)| - 1)} \tag{18}$$

$$Diversity = \frac{1}{|U|} \sum_{u \in U} Diversity(R(u)) \tag{19}$$

4.2 算法LDA_CF和LDA_MF

在 LDA_CF 中,我们选取 $K=100$,即 100 个主题.对于主题模型中的先验分布 α, β, γ 的取值,根据已有其他学者的实践经验^[15],令 $\alpha=50/K, \beta=\gamma=0.01$.我们得到 app 标签信息中的 100 个主题,每个主题都由 topic-word 概率分布来描述.选取概率最大的、最能表示该主题的前 5 个词来表示该主题.由于空间限制,表 4 展示了得到的部分主题.其中,#1 表示主题 1,“音乐(0.314)”表示词“音乐”以 0.314 的概率描述该主题.

Table 4 Topics and representative words by Linked-LDA

表 4 Linked-LDA 得到的 app 主题及代表词

编号	主题代表词
#1	音乐(0.314) 铃声(0.124) 下载(0.054) mp3(0.038) 乐器(0.013)
#2	有声读物(0.235) 现代言情(0.036) 恐怖悬疑(0.031) 教育学习(0.023) 儿童读物(0.021)
#3	消除(0.202) 休闲益智(0.16) 三消(0.068) 泡泡龙(0.021) 对对碰(0.017)
#4	办公理财(0.192) 效率办公(0.102) 记事(0.092) 理财(0.053) 笔记(0.033)
#5	理财(0.12) 股票投资(0.112) 投资(0.053) 记账(0.048) 股票(0.039)

在得到 app 的主题分布后,我们可以用该主题分布作为特征来表示 app.即每个 app 由 100 维向量表示,每一维表示该 app 在主题上的概率.对于用户,我们使用用户下载的 app 主题分布的平均值归一化处理后的结果作为用户的特征分布.我们认为:用户一周下载行为表示用户的短期兴趣,用户一个月的下载行为是用户的长期兴趣.得到用户兴趣表示后,采用 cosin 相似度计算用户兴趣和 app 主题分布相似性,将相似度高的 Top-10 的 app 作为推荐结果.

为了比较利用应用的标签和下载用户信息以及利用应用的描述信息进行主题建模的区别,我们将 LDA_CF 中的 LDA 部分替换为对应用的描述文本进行话题建模,相应的算法称为 LDA_CF(text).表 5 为 LDA_CF 推荐算法以及 LDA_CF(text)分别采用一周下载行为和一个月下载行为表示用户兴趣的推荐结果.采用一周数据和一个月数据能够给出推荐的用户数分别为 738 764(即 98.6%的用户)和 495 892(即 97.40%的用户).显然,利用一个月的信息可以推荐的用户数多于利用一周的.但是对比表中结果,我们能够看到,LDA_CF 算法在一周数据上推荐命中的用户的比例比一个月数据高.因为一周行为数据描述的是用户短期兴趣,在应用推荐中,用户一周兴趣比一个月兴趣稳定.另外,LDA_CF 比 LDA_CF(text)的性能显然要好,主要原因是应用的描述信息噪音太大,包含很多与 app 功能不相关的信息.不过,单独使用 LDA_CF 作为推荐准确率并不高.我们将该算法作为推荐候选集生成算法,使用用户一周行为数据作为 LDA_CF 的输入,选取 Top-50 结果加入推荐候选集.

Table 5 Recommendation results of LDA_CF

表 5 LDA_CF 推荐算法效果

测试指标	一个月数据		一周数据	
	LDA_CF	LDA_CF(text)	LDA_CF	LDA_CF(text)
P_{hit_user}	0.046 8	0.015 1	0.056 6	0.017 6
P_{hit_app}	0.017 4	0.006 9	0.023 3	0.008 7

在 LDA_MF 模型中,根据已有其他学者的实践经验^[14],令 $\alpha=50/K, \beta=0.01, K=100$.在 LDA 中使用的主题数目和矩阵分解阶段的隐含特征维度是一样的,即,LDA_MF 最后训练出来的用户隐含特征和 app 的隐含特征均为 100 维.LDA_MF 是对 CTR^[1]的改进,CTR 是将 LDA 与矩阵分解 MF 相结合,但是该模型只对 item(即 app)部分使用 LDA,用户部分并没有使用,其参数与 LDA_MF 设置相同.两个算法推荐准确度的比较见表 6.在推荐候选集生成时,我们选择 LDA_MF 推荐结果的 Top-50 加入推荐候选集中用于混合推荐.

在准确率方面,LDA_MF 和 CTR 推荐结果 P_{hit_users} ,即推荐准确的用户,占当天下载所有用户数的比例差不多.CTR 推荐准确率略高于 LDA_MF,但 LDA_MF 推荐结果推荐准确的不同 app 数目要多于 CTR.LDA_MF 将用户的兴趣纳入模型中,在推荐过程中,不仅仅是根据用户的下载行为学习用户和 app 的隐含特征,还将用户和 app 的语义层面也纳入考虑范围,对于行为过少的 app 信息起到补充作用.在给用户提供推荐时,用户和 app 的隐含特征均包含利用 LDA 模型学习的语义信息,所以 LDA_MF 能够将一部分和用户兴趣关联大的非热门物品推荐给用户.与 CTR 相比,LDA_MF 能够在基本保证准确率的基础上,一定程度上提高推荐结果的多样性.

Table 6 Results comparison of LDA_MF and CTR

表 6 LDA_MF 和 CTR 推荐算法对比

测试指标	LDA_MF	CTR
P_{hit_user}	0.088	0.089
P_{hit_app}	0.033	0.023

4.3 混合算法Hybrid_Rec推荐结果

在混合推荐中,为了提升推荐结果的多样性,降低热门应用的推荐.我们在用户下载的应用中剔除了下载量较大的应用,在剩余的用户下载行为中,随机抽样 20 000 个用户下载组合作为正例.而对于负例的选择,我们将用户浏览了而没有下载的应用找到,剔除集合中用户安装了的应用,随机抽样 20 000 个用户-应用组合作为负例.我们将选取的样本集合,随机选取 80%作为训练集,20%作为测试集合.对于所有样本,计算样本在 3 种算法被推荐的概率,分别作为样本的特征.逻辑回归训练完成后,得到的模型结果见公式(20),即 Item_CF 融合权重为 0.162 3,LDA_CF 算法融合权重为 0.114 8,LDA_MF 算法融合权重为 0.722 9.

$$s_v^{hybrid} = 0.0545 + 0.1623 \times s_v^{item} + 0.1148 \times s_v^{LDACF} + 0.7229 \times s_v^{LDAMF} \quad (20)$$

我们在测试样本上计算模型的 AUC 值,结果为 0.71.模型分类效果较好,说明混合算法的权重可以采用.所以计算最终用户推荐候选集合中每个 APP v 的最终得分 s_v^{hybrid} ,选择得分最高的 Top-10 作为用户最终的推荐列表.

4.4 推荐结果测评

在实验结果评测阶段,我们主要评测本文所提出的 Hybrid_Rec 混合推荐方法和传统方法在推荐准确性以及推荐多样性两方面的表现.我们对比的方法主要有:

- MF:Hu 等人提出的矩阵分解推荐算法^[6],针对于隐式行为的推荐;
- Item_CF:使用对数似然相似度度量的基于物品的协同过滤算法;
- Diverse_Rec:将 Item_CF,LDA_CF 以及 LDA_MF 推荐结果中的 Top-10 的应用作为推荐候选集合,根据 app 下载量逆序排序,将候选集合相对长尾的 Top-10 的应用推荐给用户,属于增加推荐结果多样性的方法.其中,根据下载量逆序排序,重排后推荐给用户的思路类似于文献[6];
- Most Popular:将一个月中下载量最大的 2 000 个应用随机选取 10 个推荐给用户,作为用户推荐列表;
- LDA_MF:本文提出的推荐候选集生成方法之一,将 LDA 与 MF 相结合;
- LDA_CF:本文提出的基于 Linked-LDA 和基于内容的协同过滤算法的推荐方法;
- Hybrid_Rec:本文所提出的混合推荐方法.

我们选择 2015 年 6 月 1 日~6 月 3 日这 3 天数据作为测评,评测用户的推荐列表中是否有用户实际下载过的应用.我们将每天的数据作为评测集合,3 天分别评测,取 3 天评测的平均值作为最后的结果.

在推荐准确率方面,我们主要评测两个指标: P_{hit_user} 和 P_{hit_app} .在推荐多样性评测方面,我们主要比较局部多样性以及全局多样性.局部多样性使用 $D(\text{diversity})$ 表示;而在全局多样性方面,主要比较推荐结果能够展示的所有不同应用的熵 $H(\text{all app})$ 以及推荐列表推荐准确的不同应用的熵 $H(\text{hit app})$.

表 7 展示了不同算法推荐准确性评测结果.从表中结果可知:MF,LDA_MF 以及 Hybrid_Rec 推荐算法的推荐准确率较高,其中,Hybrid_Rec 为推荐准确率最高的算法;混合推荐算法能够推荐命中更多的用户,而且算法

在所有用户推荐列表中推荐准确的 app 比例也较高,排列第二。

接着,进一步比较了推荐准确率最高的 Top-3 的算法:MF,LDA_MF 以及 Hybrid_Rec.另外,为了衡量 item_CF 在混合模型中的作用,实现了不包含该算法的推荐算法,称为 Hybrid2.表 8 对比了这 4 种算法推荐准确率以及多样性的各个指标,可以发现:Hybrid_Rec 推荐准确的用户比例最高,推荐准确的 app 比例较 LDA_MF 次之,但也相差不大.在多样性方面,不论全局多样性指标 $H(\text{hit app})$ 和 $H(\text{all app})$ 抑或是局部多样性指标 $D(\text{diversity})$,Hybrid_Rec 效果都比其他两种算法好.比较 Hybrid2 与 Hybrid_rec 可以发现,不包括 item_CF 的混合算法在全局多样性方面降低得多一些.可见,item_CF 对于提升全局多样性有一定的贡献。

Table 7 Recommendation accuracy

表 7 推荐准确率

方法	P_{hit_user}	P_{hit_app}
MF	0.088 1	0.032 2
LDA_MF	0.087 8	0.033 0
Item_CF	0.025 7	0.009 5
LDA_CF	0.056 6	0.023 3
Most Popular	0.007 2	0.002 1
Diversity_Rec	0.014 3	0.005 1
Hybrid_Rec	0.101 6	0.032 7

Table 8 Performance comparison of top-3 algorithm

表 8 推荐性能 top-3 算法测评对比

方法	P_{hit_user}	P_{hit_app}	$H(\text{hit app})$	$H(\text{all app})$	$D(\text{diversity})$
MF	0.088	0.032	2.172	1.799	0.821
LDA_MF	0.088	0.033	2.201	2.011	0.824
Hybrid2	0.093	0.033	2.243	2.793	0.834
Hybrid_Rec	0.102	0.033	2.309	2.958	0.838

综合不同推荐结果,包括准确性和多样性两方面.混合推荐算法结合了多种算法的优点,准确性最高,能够给更多的用户提供准确的推荐列表,在全局上也能够准确推荐更多的 app.在多样性方面,混合推荐方法能够给用户推荐多样性最为丰富的列表,也就是对于单个用户,混合推荐算法提供的推荐列表物品不相似性最大,用户有更多的选择范围.而在全局多样性方面,混合推荐算法也能够展示较多的应用。

5 结 论

本文研究移动应用的个性化推荐问题.与已有研究大多关注短期推荐准确度不同,本文研究如何提升推荐结果的多样性.我们发现,不同的算法在准确度和多样性方面表现不同.因此,我们主要研究如何将多种算法进行融合.为此:首先,为了进一步改进矩阵分解算法,本文提出将用户的主题模型和应用的主题模型与矩阵分解方法相结合的模型 LDA_MF 算法.为了解决用户和应用缺乏描述信息的问题,我们提出结合主题模型和基于物品的协同过滤算法的推荐算法 LDA_CF.最后,为了在保证准确度的情况下提升多样性,本文提出了将 LDA_MF, LDA_CF 以及 Item_CF 通过逻辑回归进行融合的混合算法 Hybrid_Rec.在真实大数据集上评测证实,Hybrid_Rec 能够得到准确率高而且多样性较为丰富的结果.推荐多样性问题值得深入研究,未来可以进一步研究对用户行为序列进行挖掘,将用户的长期和短期兴趣进行结合的多样化推荐方法.本文提出的混合算法融合的都是常见的推荐算法,在这些算法结果的基础上,通过逻辑回归加以综合,逻辑回归模型无需经常更新,可以设定一定的间隔进行线下更新.如何提高常用的推荐算法如 MF 的效率以便能够在线使用,近年来已有相关研究.未来可以进一步研究如何提升基于话题模型的推荐算法的效率以及如何融合高效率的算法于混合模型等问题,以期同时提升准确度、多样性和推荐效率。

References:

- [1] Koren Y. Factorization meets the neighborhood: A multifaceted collaborative filtering model. In: Proc. of the 14th ACM SIGKDD Int'l Conf. on Knowledge Discovery and Data Mining. ACM Press, 2008. 426-434. [doi: 10.1145/1401890.1401944]

- [2] Salakhutdinov R, Mnih A. Bayesian probabilistic matrix factorization using Markov chain Monte Carlo. In: Proc. of the 25th Int'l Conf. on Machine Learning. ACM Press, 2008. 880–887. [doi: 10.1145/1390156.1390267]
- [3] Rendle S, Freudenthaler C. Improving pairwise learning for item recommendation from implicit feedback. In: Proc. of the 7th ACM Int'l Conf. on Web Search and Data Mining. ACM Press, 2014. 273–282. [doi: 10.1145/2556195.2556248]
- [4] Kabbur S, Ning X, Karypis G. Fism: Factored item similarity models for top- n recommender systems. In: Proc. of the 19th ACM SIGKDD Int'l Conf. on Knowledge Discovery and Data Mining. ACM Press, 2013. 659–666. [doi: 10.1145/2487575.2487589]
- [5] Wang C, Blei DM. Collaborative topic modeling for recommending scientific articles. In: Proc. of the 17th ACM SIGKDD Int'l Conf. on Knowledge Discovery and Data Mining. ACM Press, 2011. 448–456. [doi: 10.1145/2020408.2020480]
- [6] Adomavicius G, Kwon YO. Improving aggregate recommendation diversity using ranking-based techniques. IEEE Trans. on Knowledge and Data Engineering, 2012,24(5):896–911. [doi: 10.1109/TKDE.2011.15]
- [7] Niemann K, Wolpers M. A new collaborative filtering approach for increasing the aggregate diversity of recommender systems. In: Proc. of the 19th ACM SIGKDD Int'l Conf. on Knowledge Discovery and Data Mining. ACM Press, 2013. 955–963. [doi: 10.1145/2487575.2487656]
- [8] Blei DM, Ng AY, Jordan MI. Latent dirichlet allocation. Journal of Machine Learning Research, 2003,3:993–1022.
- [9] Koren Y, Bell R, Volinsky C. Matrix factorization techniques for recommender systems. Computer, 2009,8:30–37. [doi: 10.1109/MC.2009.263]
- [10] Takane Y, Young FW, De Leeuw J. Nonmetric individual differences multidimensional scaling: An alternating least squares method with optimal scaling features. Psychometrika, 1977,42(1):7–67. [doi: 10.1007/BF02293745]
- [11] Erosheva E, Fienberg S, Lafferty J. Mixed-Membership models of scientific publications. Proc. of the National Academy of Sciences, 2004,101:5220–5227. [doi: 10.1073/pnas.0307760101]
- [12] Nallapati RM, Ahmed A, Xing EP, Cohen WW. Joint latent topic models for text and citations. In: Proc. of the 14th ACM SIGKDD Int'l Conf. on Knowledge Discovery and Data Mining. ACM Press, 2008. 542–550. [doi: 10.1145/1401890.1401957]
- [13] Casella G, George EI. Explaining the Gibbs sampler. The American Statistician, 1992,46(3):167–174.
- [14] Xiang L. Recommender System Practice. Beijing: Posts & Telecom Press, 2012 (in Chinese).
- [15] Griffiths TL, Steyvers M. Finding scientific topics. Proc. of the National Academy of Sciences, 2004,101(Suppl. 1):5228–5235. [doi: 10.1073/pnas.0307752101]
- [16] Hu Y, Koren Y, Volinsky C. Collaborative filtering for implicit feedback datasets. In: Proc. of the 8th IEEE Int'l Conf. on Data Mining (ICDM 2008). 2008. 263–272. [doi: 10.1109/ICDM.2008.22]

附中文参考文献:

- [14] 项亮. 推荐系统实践. 北京: 人民邮电出版社, 2012.



黄璐(1990—),女,广东饶平人,硕士生,主要研究领域为数据挖掘,主体模型,推荐系统.



林川杰(1992—),男,硕士生,主要研究领域为深度学习,数据挖掘.



何军(1962—),男,博士,教授,博士生导师,CCF 高级会员,主要研究领域为数据挖掘,社交网络分析,推荐系统.



刘红岩(1968—),女,博士,教授,博士生导师,主要研究领域为大数据分析,商务智能,数据挖掘.



杜小勇(1963—),男,博士,教授,博士生导师,CCF 会士,主要研究领域为数据库系统,大数据管理,智能信息检索,知识工程.