通过相似度支持度优化基于 K 近邻的协同过滤算法

罗 辛 欧阳元新 熊 璋 袁 满
(北京航空航天大学计算机学院 北京 100191)

摘 要 个性化推荐系统能基于用户个人兴趣为用户提供定制信息.此类系统通常使用协同过滤技术实现,其中一种广泛使用的经典模型是基于用户评分相似度的 k 近邻模型.使用 k 近邻模型需要预先计算出用户或者项目的 k 个最近邻居,k 值过大时会导致计算量过大而影响推荐产生的实时性,而 k 值过小则会导致推荐精度下降.为解决此问题,该文中提出了一种新的最近邻度量——相似度支持度.基于相似度支持度,该文提出了数种能够在保持推荐精度和密度的前提下维持合理规模的 k 近邻的策略.在真实大规模数据集上的实验结果表明,相比传统算法,该文提出的策略能够在保证推荐精度的前提下大幅降低计算复杂度.

关键词 个性化推荐;协同过滤;相似度支持度;k 近邻;近邻关系模型
中图法分类号 TP391 **DOI 号**: 10.3724/SP.J.1016.2010.01437

The Effect of Similarity Support in *K*-Nearest-Neighborhood Based Collaborative Filtering

LUO Xin OUYANG Yuan-Xin XIONG Zhang YUAN Man
(*School of Computer Science, Beihang University, Beijing* 100191)

Abstract Recommender systems which can provide people with personalized suggestions usually rely on Collaborative Filtering (CF). A classical approach to CF is based on k-nearest-neighborhood (kNN) model, where the most important task is constructing the kNN sets for involved users or items. However, when constructing kNN sets, there is a dilemma to decide the value of k—A too small value will lead to poor recommendation performance, whereas a too large one will result in unacceptable computational complexity. In this work the authors first empirically validated that the suitable value of k in kNN based CF was affected by the number of the totally involved entities, and then focused on improving the quality of the kNN sets in kNN based CF for providing high recommendation performance as well as maintaining suitable kNN set size. To achieve this objective, the authors propose a novel kNN metric named Similarity Support (SS). By taking SS into consideration during the kNN building process, the authors design a series of strategies for optimizing kNN based CF. The empirical studies on public large, real datasets show that due to the improvement on the quality of kNN set brought by SS, CF adjusted by the new strategies turned out to be superior to kNN based CF in term of both recommendation performance and computational complexity.

Keywords recommender system; collaborative filtering; similarity support; k-nearest neighborhood; neighborhood based model

---

收稿日期:2010-06-11.本课题得到软件开发环境国家重点实验室探索性自选课题与中央高校基本科研业务费专项资金(YWF-10-02-012)资助.**罗 辛**,男,1983 年生,博士研究生,研究方向主要包括数据挖掘与数据活化.E-mail: luoxin21@gmail.com.**欧阳元新**,女,1978 年生,博士,讲师,研究兴趣集中于数据挖掘和工作流管理.**熊 璋**,男,1956 年生,教授,博士生导师,目前的研究兴趣集中于智慧城市和数据活化.**袁 满**,男,1987 年生,博士研究生,研究方向主要集中在智慧城市方向.

1 引 言

个性化推荐系统能基于用户个人兴趣为用户提供定制信息.此类系统通常使用协同过滤技术实现[1].在基于协同过滤的推荐系统中,用户对于相关项目的兴趣以一个用户-项目评分矩阵表示,其中较高的评分对应较强的用户兴趣.故而协同过滤推荐的问题可被看作矩阵缺失值估计问题:根据用户-项目评分矩阵中已有的评分值,对未知的评分值进行估计.根据在协同过滤领域内的最近研究成果,构造协同过滤推荐系统时最常用的两类模型是近邻关系模型[2-5]和隐向量模型[6-8].近邻关系模型通过构建用户与用户之间,或者项目与项目之间的关联,从而建立起相应实体的近邻关系;进行推荐时,推荐系统根据当前用户的近邻已做出的评分,或当前项目的近邻已获得的评分来进行预测.与近邻关系模型不同,隐向量模型使用矩阵因式分解技术来对评分矩阵进行分析:其将用户和项目映射至相同维数的隐向量空间,并根据已有的评分训练相应的隐向量;进行推荐时,推荐系统使用相应隐向量对的内积作为对未知评分的预测.相较于近邻关系模型,隐向量模型能够更充分地描述数据的多方面特性.然而,近邻关系模型具备更高的灵活性,更易与其它模型整合,并且其推荐结果也更加直观、易于理解[5].因此,在实际应用中,大部分推荐系统采用基于近邻关系模型的协同过滤技术.

在基于近邻关系的协同过滤模型中,一种被广泛采用的经典模型是 k 近邻模型,其工作原理是利用评分相似度来构造用户或者项目的 k 近邻集合[1-5],再使用 k 近邻集合进行推荐.使用 k 近邻模型需要预先计算出用户或者项目的 k 个最近邻居,k 值过大时会导致计算量过大而影响推荐产生的实时性,而 k 值过小则会导致推荐精度下降.本文立足于优化基于 k 近邻的协同过滤推荐模型,提出一种新的 k 近邻度量——相似度支持度,并基于该度量,提出了一系列对 k 近邻模型的优化策略.本文的主要贡献包括:

(1)假设在基于 k 近邻的协同过滤推荐系统中,k 的最优值取决于该推荐系统内的项目总数;并对该假设进行了实验验证.

(2)提出一种新的 k 近邻度量——相似度支持度,并基于此度量提出了一系列对 k 近邻模型的优化策略,能够在优化 k 近邻规模的同时提高推荐质量.

(3)在大规模真实数据集上,对本文提出的优化策略进行了实验验证.

本文第 2 节讨论相关工作,并提出问题定义;第 3 节介绍本文的策略;第 4 节给出实验结果及分析;最后在第 5 节中做出结论.

2 相关工作和问题定义

协同过滤推荐问题的定义如下:给定用户集 U 和项目集 I,则用户对于项目的兴趣可以表示为一个 $|U|\times|I|$ 的矩阵 $\boldsymbol{R}$,在该矩阵中,每一行向量表示一特定用户的评分集合,每一列向量表示一特定项目的被评分集合,每一元素 $r_{ui}\in\boldsymbol{R}$ 表示用户 u 对于项目 i 的评分(通常评分越高,代表用户对该项目兴趣越强).一般情况下,$\boldsymbol{R}$ 中已知评分的数量远远小于未知评分的数量.给定评分集 $T\subset\boldsymbol{R}$($|T|\ll|\boldsymbol{R}|$)作为训练集,根据 T 中已知的评分,构造一个推荐系统,该系统需要能以最小的累积误差来对 $\boldsymbol{R}$ 中未知的评分进行预测.推荐系统的累积误差将在验证集 V 上进行验证.为避免过度拟合,验证集 V 中的数据不能够用于推荐系统的训练过程,即 $V\subset\boldsymbol{R}$ 且 $V\cap T=\varnothing$.

实现协同过滤推荐系统时,一类常用模型是近邻关系模型.在近邻关系模型中,对于未知评分进行预测的前置条件是对用户与用户之间,或者项目与项目之间的关系进行建模.根据所建模的关系种类,近邻关系模型可进一步细分为基于用户的近邻关系模型和基于项目的近邻关系模型两类.由于在实际应用中,项目数量更加稳定,并往往远低于用户数量,因此,对项目之间的关系进行建模是更为常用的方法[2-5].

k 近邻模型是一种被广泛采用的经典近邻关系模型[1-5].在 k 近邻模型中,项目间的关系使用评分相似度 rs 表示.评分相似度通常使用余弦相似度、Pearson 相关相似度和修正余弦相似度进行度量[1-2],其中 Pearson 相关相似度运用最为普遍.在 k 近邻模型中,对于指定项目 i,系统将项目 i 与系统中其他项目 $j\in(R-i)$ 之间的评分相似度按照从高至低的顺序进行排序,并记录下与项目 i 具备最高评分相似度的 k 个项目,该 k 个项目就是项目 i 的 k 近邻集合,记为 $kNN(i)$.当需要为用户 u 进行推荐时,对未知评分 r_{ui} 的预测,是在用户 u 在 $kNN(i)$ 上的已知评分和 $kNN(i)$ 中已经被用户 u 进行过评分

的项目与项目 i 的评分相似度这两者的基础上做出的，表示为

$$\hat{r}_{ui}=\frac{\sum_{j\in kNN(i)\cap R(u)} rs_{ij}\cdot r_{uj}}{\sum_{j\in kNN(i)\cap R(u)} rs_{ij}} \tag{1}$$

其中 $\hat{r}_{ui}$ 表示推荐系统对于未知评分 r_{ui} 的预测值，$R(u)$ 表示用户 u 的已知评分集合，rs_{ij} 表示项目 i 和项目 j 之间的评分相似度. 使用 Pearson 相关相似度计算 rs_{ij} 时，计算方法如下

$$rs_{ij}=\frac{\sum_{u\in R(i)\cap R(j)}(r_{ui}-\overline{r_i})\cdot(r_{uj}-\overline{r_j})}{\sqrt{\sum_{u\in R(i)\cap R(j)}(r_{ui}-\overline{r_i})^2}\cdot\sqrt{\sum_{u\in R(i)\cap R(j)}(r_{uj}-\overline{r_j})^2}} \tag{2}$$

其中 $R(i)$ 和 $R(j)$ 分别表示已知的对项目 i 和项目 j 的评分集合，$\overline{r_i}$ 和 $\overline{r_j}$ 分别表示在项目 i 和项目 j 上的平均评分值.

基于式(1)的未知评分预测规则可以通过预先移除全局平均评分和相应实体相对于全局均值的观察偏差来进行改进[5]，表示为

$$\hat{r}_{ui}=\mu+b_u+b_i+\frac{\sum_{j\in kNN(i)\cap R(u)} rs_{ij}\left[r_{uj}-(\mu+b_u+b_j)\right]}{\sum_{j\in kNN(i)\cap R(u)} rs_{ij}} \tag{3}$$

其中 μ 表示训练集的全局平均评分，b_u 表示用户 u 相对于 μ 的观察偏差，b_i 表示项目 i 相对于 μ 的观察偏差. 参数 μ、b_u 和 b_i 可以用如下方式进行估计：

$$\begin{aligned}\mu&=\sum_{(u,i)\in T} r_{ui}/|T|,\\ b_i&=\sum_{(u,i)\in T}(r_{ui}-\mu)/(\beta_1+|R(i)|),\\ b_u&=\sum_{(u,i)\in T}(r_{ui}-\mu-b_i)/(\beta_2+|R(u)|)\end{aligned} \tag{4}$$

其中 β_1 和 β_2 是使用交叉验证方法确定的常量. 令 $b_{ui}=\mu+b_u+b_i$，$R_i^k(u)=kNN(i)\cap R(u)$，代入式(3)，可将其简化为

$$\hat{r}_{ui}=b_{ui}+\frac{\sum_{j\in R_i^k(u)} rs_{ij}\cdot(r_{uj}-b_{uj})}{\sum_{j\in R_i^k(u)} rs_{ij}} \tag{5}$$

推荐规则(5)具有相当的合理性：对于与当前用户给予较高评分的项目具备高评分相似度的项目，系统将会给出较高的用户评分预测值. 然而，在为每个项目构造 k 近邻集合时，却面临着两难抉择：k 值过小，会导致推荐系统的性能急剧降低；k 值过大，则会导致在进行推荐时计算量过大而影响到推荐产生的实时性. 根据现有研究成果. k 的最优值取值应在[30,60]区间内，如果 k 的取值超出该区间的范围，推荐系统的性能将很难继续提高[2-5]. 然而我们在实践中发现，当推荐系统中的项目总数逐渐上升时，上述 k 的最优值区间将会逐渐失准. 根据上述情况，本文提出假设如下.

假设 1.　在基于 k 近邻模型的协同过滤推荐系统中，k 的最优值依赖于在推荐系统内相互间存在评分相似度的项目总数.

对于该假设的实验验证将在下一节中给出.

此外，现有的 k 近邻模型仅根据评分相似度来构建各个项目的 k 近邻集合. 在实际应用中，当两个项目间评分重叠率非常低时，往往会导致这两个项目间具备非常高的评分相似度，从而使这两个项目互相属于对方的 k 近邻集合. Herlocker 等的研究表明，这些基于极少样本计算出的最近邻往往会导致十分荒谬的推荐结果[9].

k 近邻模型的上述两点缺陷事实上是由同一个原因引起的，即项目的 k 近邻集合仅由评分相似度决定，具备很大的片面性. 从这点出发，我们提出了一种新的 k 近邻度量——相似度支持度，其定义如下.

定义 1.　相似度支持度 ss，是评分相似度的支持样本数量. 项目 i 和 j 间的相似度支持度就是同时对项目 i 和 j 进行评分的用户数量，表示为 $ss_{ij}=|U_{ij}|$，其中 U_{ij} 表示同时对项目 i 和 j 进行评分的用户集.

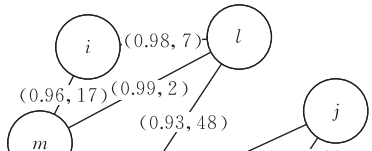

图 1　使用(rs,ss)二元组表示项目间的关系

事实上，如果我们使用无向图来表示项目间的关系，那么图上每一条边的权值可以表示为一个由评分相似度和相似度支持度组成的二元组，如图 1 所示. 在构造各个项目的 k 近邻集合时，若两个项目之间的评分相似度较高，说明这两个项目被评分集的匹配度较高；若两个项目之间的相似度支持度较高，则说明这两个项目代表的用户兴趣点重叠度较大. 如果我们能够整合评分相似度和相似度支持度，在构造项目的 k 近邻集合的过程中同时使用这两种

度量,将很有希望提高 k 近邻集合的质量,从而提高推荐系统的性能.

近年来,国外也有研究人员针对优化 k 近邻模型中 k 近邻集质量的问题进行了研究. Herlocker 等在文献[9]中,使用两个用户共同评分项目的数量除以事先约定的阈值得到一个相似度缩放系数,并使用该系数对用户间的评分相似度进行修正. Bell 等在文献[4]以及 Koren 在文献[5]中,都提出了对具备较少支持样本数量的评分相似度进行缩减控制的规则. 上述方法与本文提出的优化策略相比,主要存在以下不足:(1)没有提出对支持度相似度的明确定义;(2)都需要预先定义一个阈值,该阈值只能通过交叉验证方法来确定,将导致计算更加复杂;(3)只能对少数缺少样本支持的评分相似度进行调整,不能反映整个数据集上相似度支持度的分布情况. 与现有方法相比,本文提出的优化策略不存在上述弊端. 在下一节中,我们将对本文提出的优化策略进行详细阐述.

3 基于相似度支持度的 k 近邻优化策略

在本节中,我们将首先对前文提出的假设 1 进行实验验证,然后对本文提出的策略进行阐述.

3.1 假设 1 的实验验证

相关研究认为,在基于 k 近邻模型的推荐系统中,建立项目的 k 近邻集合时,k 的最优取值应在[30,60]区间内;如果 k 超出该区间范围,推荐精度将几乎不会提高[2-4]. 然而我们在实践中发现,当推荐系统中相互之间存在评分相似度关系的项目数量逐渐上升时,该论断将会逐渐失准. 因此,本文提出了假设 1,并设计了以下实验进行验证.

实验数据集. 实验在 MovieLens10K 数据集(以下简称 ML10K)上进行. 该数据集由明尼苏达大学 GroupLens 研究小组通过 MovieLens 网站收集,包含了 943 个用户对 1682 个项目的 10000 条评分信息. 所有的评分值分布在[0,5]区间内,越高的评分值代表越强的用户兴趣.

实验评判度量. 实验使用平均绝对误差 *MAE* 和推荐覆盖度 *Coverage* 作为推荐系统性能的评判度量. 其中 *MAE* 是被广泛采用的用于评判推荐系统预测精度的度量[10]. 在计算推荐系统的 *MAE* 之前,首先需要计算用户平均绝对误差 *MAUE*. *MAUE* 的计算方法如下式所示

$$MAUE_u = \sum_{i \in IP(u) \cap IR(u)} |\hat{r}_{ui} - r_{ui}| / |IP(u) \cap IR(u)| \tag{6}$$

其中 $IP(u)$ 是推荐系统为用户 u 推荐的项目集,$IR(u)$ 是用户 u 在测试数据集上进行评分的项目集. 计算出每个用户的 *MAUE* 后,就可以计算出该推荐系统的 *MAE*,如下式

$$MAE = \sum_{u \in U} MAUE_u / |U| \tag{7}$$

MAE 越低,代表推荐系统的预测精度越高.

而 *Coverage* 是一项被广泛使用的用以评价推荐系统推荐覆盖度的评判度量,指的是推荐系统为用户推荐的项目集对用户兴趣的覆盖范围[10],其计算方式为

$$Coverage = \sum_{u \in U} |IP(u) \cap IR(u)| / \sum_{u \in U} |IR(u)| \tag{8}$$

Coverage 越高,代表推荐系统对用户兴趣的覆盖能力越强.

实验设置. 为了检验 k 的最优取值随着项目数量的增多是否会有相应的变化,实验分为 3 个阶段,每个阶段分别从 ML10K 数据集中随机选取 100 个项目、500 个项目和 1000 个项目以及这些项目相应的评分作为实验数据集. 这 3 个子数据集被分别记为 I100、I500 和 I1000. 在每个子数据集上,实验按照 90%～10%的比例构造训练-测试数据. 在实验中,项目间的相似度使用式(2)中的 Pearson 相关相似度进行度量. 实验在基于经过改进的 k 近邻模型的推荐系统上进行,该推荐系统将使用式(5)对未知评分进行预测.

实验结果. 图 2、图 3 和图 4 分别给出了在 I100、I500 和 I1000 这 3 个数据集上 k 值对于 *MAE* 的影响. 可以看到,在 I100 数据集上,由于当 k 值大于 60 时,推荐系统的 *MAE* 几乎不会再继续降低,

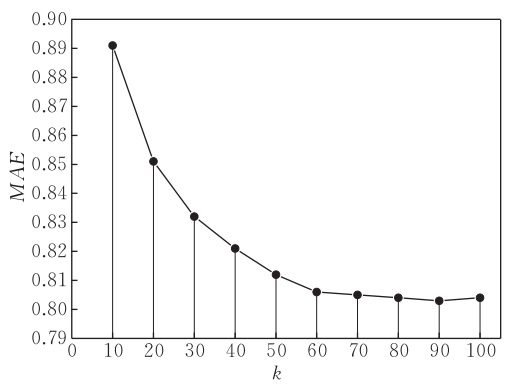

图 2 在 I100 数据集上 k 值对 *MAE* 的影响

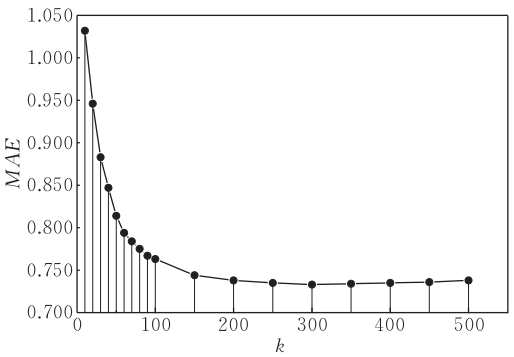

图 3　在 I500 数据集上 k 值对 MAE 的影响

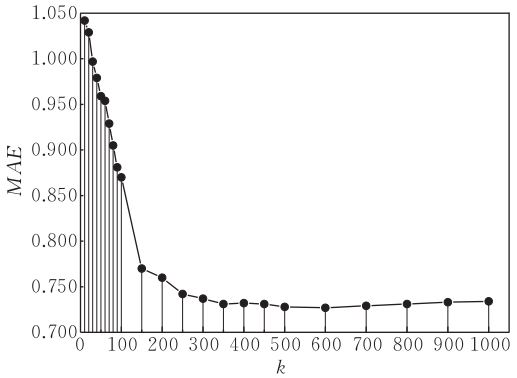

图 4　在 I1000 数据集上 k 值对 MAE 的影响

故而在 I100 数据集上 k 的最优值是 60，此时，现有研究建议的 k 最优值取值区间[30,60]仍然适用；然而在 I500 数据集上，k 的最优取值为 200；在 I1000 数据集上，k 的最优取值则为 350，均不在[30,60]区间内.

图 5、图 6 和图 7 分别给出了在 I100、I500 和 I1000 这 3 个数据集上 k 值对于 $Coverage$ 的影响. 在 I100 数据集上，当 k 值大于 70 时，$Coverage$ 将停止上升；而在 I500 和 I1000 数据集上，$Coverage$ 将分别在 $k=200$ 和 $k=300$ 时达到峰值. 就 $Coverage$ 而言，无论是在 I100、I500 还是 I1000 数据集上，k 的最优值均不在[30,60]区间内.

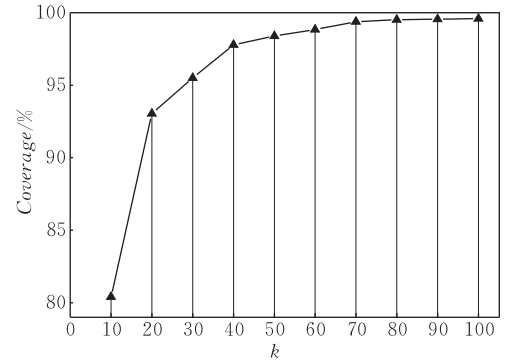

图 5　在 I100 数据集上 k 值对 $Coverage$ 的影响

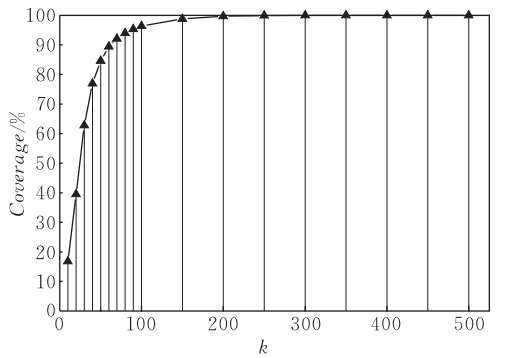

图 6　在 I500 数据集上 k 值对 $Coverage$ 的影响

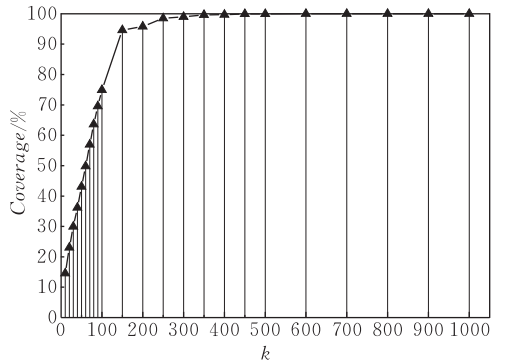

图 7　在 I1000 数据集上 k 值对 $Coverage$ 的影响

基于以上实验结果，我们可以得出以下结论：

(1) 使用 MAE 和 $Coverage$ 这两项评判度量对基于 k 近邻模型的推荐系统进行性能评判时，k 的最优取值将很有可能位于[30,60]的取值区间之外；

(2) 在基于 k 近邻的推荐系统中，当项目数量上升时，k 的最优取值也有上升的趋势.

综合上述结论，我们可以判定假设 1 是正确的.

3.2　使用相似度支持度改进 k 近邻模型

从定义 1 中可以发现，相似度支持度实际上代表了项目间评分相似度的可信度，若相似度支持度较大，说明评分相似度支持样本数量较多，可信度较高；反之，则说明评分相似度支持数量较少，可信度较低. 若能够在建立项目 k 近邻的过程中，能够同时从项目间评分相似度和相似度支持度两方面进行全面的考虑，将很有可能提高项目 k 近邻的质量，从而最终提高推荐系统的性能. 从这点出发，本文设计了以下几种整合应用评分相似度和相似度支持度，来构造项目 k 近邻集的策略：

SSR(相似度支持度排名)策略. SSR 策略完全使用相似度支持度来构造项目的 k 近邻集合. 经过 SSR 策略调整的 k 近邻模型将会把当前项目的近邻按照与当前项目间的相似度支持度的高低进行排序，并选取与当前项目具备最高的相似度支持度的

前 k 个近邻构成当前项目的 k 近邻集.在使用 SSR 策略的 k 近邻模型中,相似度支持度对未知评分预测过程没有影响:在对未知评分进行预测时,推荐系统仍然只使用评分相似度来产生预测.

SSR 策略实际上就是利用相似度支持度对项目间评分相似度的可信度的代表性,对 k 近邻模型进行改进.这与现实生活中的人际关系十分相似:基于寥寥可数的几次接触,人们往往很难对他人给出合理、可信的评价;反之,如果两人之间频繁联系,相互熟悉,则他们对彼此的评价将更具备代表性.根据相似度支持度来选取当前项目的 k 近邻集也是基于类似的原理:与当前项目具备较高的相似度支持度的近邻就是与当前项目"联系"频繁的对象,因而这两个项目之间的评分相似度应具备更强的代表性,所蕴含的信息也更多.

ISW(相似度支持度加权)策略. ISW 策略在构造项目的 k 近邻集合时,将会根据当前项目与其近邻间的相似度支持度,为当前项目的每个近邻分配一个相似度支持度权重 sw,其计算公式为

$$sw_{ij} = \frac{ss_{ij} - \min(ss)}{\max(ss) - \min(ss)} \tag{9}$$

其中 sw_{ij} 表示项目 i 和项目 j 间的相似度支持度权重,$\min(ss)$ 和 $\max(ss)$ 分别表示在训练集上相似度支持度的最小值和最大值.计算出的相似度支持度权重将会用于对评分相似度进行调整,如下式所示

$$rs'_{ij} = rs_{ij} \cdot sw_{ij} \tag{10}$$

调整后的相似度支持度将会被用以构造当前项目的 k 近邻集合,并在式(5)中替换原始评分相似度进行未知评分预测.ISW 策略将会以如下方式构造项目 k 近邻集:综合考虑当前项目与其近邻间的评分相似度和相似度支持度,并优先选择在这两种度量上同时具备较高数值的近邻.

由于不同项目间的相似度支持度差异很大,故而使用 ISW 策略调整后的评分相似度可能会存在数量级上的差别.但由式(5)可知,评分相似度的数值变化会因为除法运算而抵消,因此,使用 ISW 策略不会导致推荐系统对于未知评分的预测结果间存在数量级的差异.事实上,ISW 策略将会极大弱化当前项目的近邻集中具备较低相似度支持度的近邻项目对未知评分预测的影响,反之亦然.

GW(相似度支持度高斯加权)策略. GW 策略根据当前项目与其近邻间的相似度支持度,为当前项目的每个近邻分配一个高斯权重,用以调整其与当前项目间的相似度支持度.使用 GW 策略来构造项目的 k 近邻集时,首先需要使用高斯分布 $N(\mu, \sigma^2)$ 来拟合项目间的相似度支持度.其中参数 μ 和 σ^2 可以使用极大似然估计法进行估计,如下式所示

$$\hat{\mu} = \overline{ss} = \frac{1}{N}\sum ss_{ij},\ \hat{\sigma}^2 = \frac{1}{N}\sum (ss_{ij} - \hat{\mu})^2 \tag{11}$$

其中 N 表示在给定训练数据集上训练得到的相似度支持度的数量.对项目间的相似度支持度进行拟合后,就可以对每个评分相似度根据其相似度支持度赋予一个支持度高斯权重,该权重由拟合得到的高斯分布的概率分布函数决定,如下式所示

$$gw_{ij} = F(ss_{ij} \mid \hat{u}, \hat{\sigma}^2) = \int_{-\infty}^{ss_{ij}} f(ss \mid \hat{u}, \hat{\sigma}^2)\mathrm{d}ss \tag{12}$$

其中 gw_{ij} 表示项目 i 和项目 j 间的相似度支持度权重,$F(ss_{ij} \mid \hat{u}, \hat{\sigma}^2)$ 表示高斯概率分布函数,$f(ss \mid \hat{u}, \hat{\sigma}^2)$ 则表示高斯概率密度函数.gw_{ij} 的值实际上代表在当前训练数据集上,项目 i 和 j 之间的相似度支持度大于等于其它项目间相似度支持度的概率.最后,类似于 ISW 策略,我们将评分相似度与对应的高斯权重相乘,从而对评分相似度进行调整

$$rs'_{ij} = rs_{ij} \cdot gw_{ij} \tag{13}$$

调整后的相似度支持度将会被用以构造项目的 k 近邻集合和评分预测.

GW 策略将会以如下方式构造项目 k 近邻集:综合考虑当前项目与其近邻间的评分相似度和相似度支持度,并优先选择在这两种度量上同时具备较高数值的近邻;大部分相似度支持度值在均值附近的近邻,其高斯权值均位于中间值 0.5 左右,因而这些近邻在当前项目的近邻集内的相对排序不会受到太大影响.

4 实验结果及分析

实验数据集. 为了检验本文提出的优化策略在不同数据集上的效果,我们分别在两个数据集上进行了实验.第一个数据集是 3.1 节中用以验证假设 1 的 ML10K 数据集的全集.第 2 个数据集取自 Netflix 数据集,该数据集是 Netflix 公司举办 Netflix 竞赛时所使用的数据集,其中包含了 Netflix 公司随机挑选的 48 万名匿名客户对 1 万 7 千个项目超过 1 亿条的评分数据.该数据集中的所有评分值分布在区间[1, 5]内.我们的实验在 Netflix 数据集中前 1000 个项目的被评分数据上进行.该子数据集总共

包含约 40 万名用户在这 1000 个项目上超过 5 百万条的评分数据，简称为 NF5M 数据集.

实验评判度量. 实验使用 3.1 节中提到的 *MAE* 和 *Coverage* 作为推荐系统性能的评判度量.

实验设置. 在每个实验数据集上，实验按照 90%～10%的比例构造训练-测试数据. 在实验中，项目间的相似度使用式(2)所示的 Pearson 相关相似度进行度量. 实验使用基于经过改进的 k 近邻模型的推荐系统作为基准参照方法，该推荐系统根据式(5)对未知评分进行预测. 然后实验分别使用 SSR 策略、ISW 策略和 GW 策略对基准参照方法进行优化，并与基准参照方法对比，以验证这 3 种优化策略的实际效果.

实验结果. 图 8 和图 9 给出了在 ML10K 数据集上的实验结果，图 10 和图 11 则给出了在 NF5M 数据集上的实验结果. 在所有图例中，CF 表示基准参照方法；SSR-CF、ISW-CF 和 GW-CF 则分别表示使用 SSR 策略、ISW 策略和 GW 策略对基准参照方法进行改进后的推荐系统.

首先我们看在 ML10K 数据集上的实验结果. 图 8 给出了实验中各推荐系统在 ML10K 上的 *MAE* 对比情况，而图 9 则给出了 *Coverage* 对比情况. 如图 8 和图 9 所示，使用基于相似度支持度的优化策略能够显著地提升推荐系统的性能. 与基准参照方法相比，使用 SSR 策略、ISW 策略和 GW 策略改进后的推荐系统都能达到更低的 *MAE* 值. 更重要的是，使用优化策略改进后，推荐系统的收敛速度要远远快于改进前. 在本文提出的 3 种优化策略中，效果最好的是 GW 策略：在 $k=200$ 时，使用 GW 策略优化的推荐系统就能获得比参照基准方法在 $k=1000$ 时更高的推荐精度和非常接近的推荐覆盖度，如表 1 所示. 这意味着 GW 策略在提高推荐质量的同时还降低了 80%的计算量.

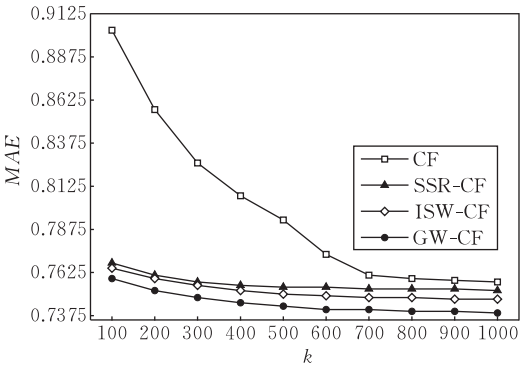

图 8　实验中各推荐系统在 ML10K 数据集上的 *MAE* 比较

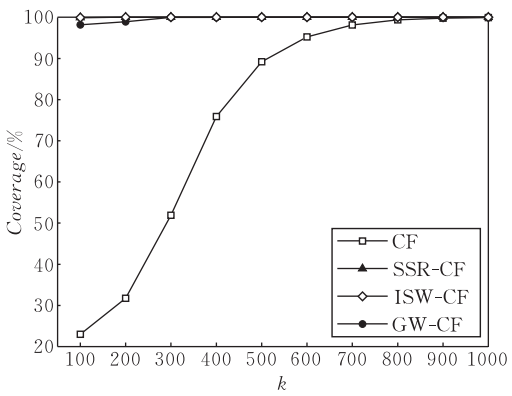

图 9　实验中各推荐系统在 ML10K 数据集上的 *Coverage* 比较

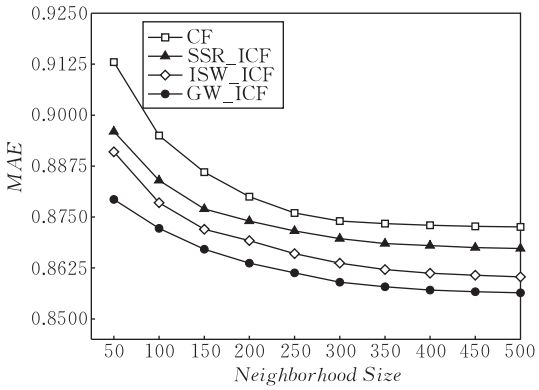

图 10　实验中各推荐系统在 NF5M 数据集上的 *MAE* 比较

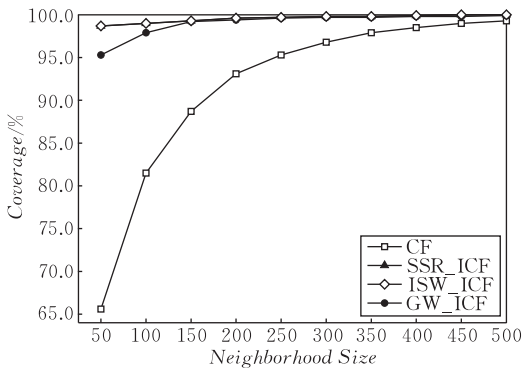

图 11　实验中各推荐系统在 NF5M 数据集上的 *Coverage* 比较

表 1　实验中各推荐系统在 ML10K 数据集上的性能比较

推荐系统	评判度量		
	MAE	*Coverage*/%	k 值
CF	0.7573	100.0	1000
SSR-CF	0.7569	100.0	300
ISW-CF	0.7554	100.0	300
GW-CF	0.7521	98.9	200

图 10 和图 11 分别给出了实验中各推荐系统在 NF5M 数据集上 *MAE* 和 *Coverage* 的对比情况. NF5M 上的实验结果和 ML10K 上十分相似，GW_GF 仍然有最佳的表现，在 $k=100$ 时，使用 GW 策略优化的推荐系统就能获得比参照基准方法在 $k=500$ 时更高的推荐精度和非常接近的推荐覆盖度，

如表 2 所示.

表 2 实验中各推荐系统在 NF5M 数据集上的性能比较

推荐系统	评判度量		
	MAE	*Coverage*/%	k 值
CF	0.8726	99.3	500
SSR-CF	0.8716	99.7	250
ISW-CF	0.8720	99.3	150
GW-CF	0.8722	97.9	100

在实验过程中,我们注意到,使用 ISW 策略和 GW 策略优化的推荐算法,除了收敛速度明显加快外,预测精度也有较为显著的提高.这是由于这两种策略均使用了经过修正的评分相似度进行未知评分预测.从实验结果来看,GW 策略对评分相似度的修正方式能产生更好的推荐效果.

复杂度分析.基于实验结果,我们发现使用基于相似度支持度的优化策略对基于 k 近邻模型的推荐算法进行优化,在进行推荐时,不但可以显著地降低计算复杂度,还可以提高预测精度.然而,引入相似度支持度将会提高推荐系统训练过程中的计算复杂度.在基于 k 近邻的推荐系统中,训练的时间复杂度和空间复杂度均为 $O(n^2)$.引入相似度支持度后,会在以下两个方面增加训练时的计算复杂度.

对相似度支持度的存储,额外需要大小为 $O(n^2)$的存储空间,因此引入 SS 后,推荐系统在训练时的空间复杂度为 $O(2n^2)$.

使用 ISW 策略和 GW 策略时,由于系统需要根据相应的相似度支持度对每个评分相似度进行修正,就需要在全部的评分相似度上迭代一次,因而推荐系统在训练时的时间复杂度为 $O(2n^2)$.

5 结 论

在基于 k 近邻模型的协同过滤推荐系统中,一项最为关键的任务是为每个项目构造用以进行未知评分预测的 k 近邻集.在这个过程中,面临着对 k 值的两难选择:如果 k 值过大,会导致系统在进行推荐时计算量过大,从而影响推荐产生的实时性;如果 k 值过小,则会导致推荐系统的性能急剧下降.以往研究认为 k 的最优取值区间应在[30,60] 区间内,如果 k 值超出该区间的范围,推荐系统的性能将会很难再继续提高.但我们在实践中发现 k 的最优值事实上与推荐系统中的项目数量密切相关,当推荐系统中的项目数量上升时,为了获得较好的推荐效果,k 的取值也需要有相应增加.为了能够在维持推荐系统性能的同时维持合理的 k 近邻集规模,本文提出了一个新的 k 近邻度量——相似度支持度.基于对相似度支持度和评分相似度的综合考虑,本文提出了 SSR、ISW 和 GW 三种对基于 k 近邻模型的推荐系统的优化策略.在大规模真实数据集上的实验结果表明,使用本文提出的优化策略优化后的基于 k 近邻模型的推荐系统,其进行推荐时的计算复杂度显著降低,并且能提供更好的推荐效果.

参 考 文 献

[1] Adomavicius G, Tuzhilin A. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. IEEE Transactions on Knowledge and Data Engineering, 2005, 17(6): 734-749

[2] Sarwar B, Karypis G, Konstan J, Reidl J. Item-based collaborative filtering recommendation algorithms//Proceedings of the 10th International Conference on World Wide Web. Hong Kong, China, 2001: 285-295

[3] Deshpande M, Karypis G. Item-based top-n recommendation algorithms. ACM Transactions on Information Systems, 2004, 22(1): 143-177

[4] Bell R M, Koren Y. Improved neighborhood-based collaborative filtering//Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. California, 2007: 7-14

[5] Koren Y. Factor in the Neighbors: Scalable and accurate collaborative filtering. ACM Transactions on Knowledge Discovery from Data, 2009, 4(1): 1-24

[6] Kurucz M, Benczúr A A, Csalogány K. Methods for large scale SVD with missing values//KDD Cup Workshop at Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. California, 2007: 31-38

[7] Paterek A. Improving regularized singular value decomposition for collaborative filtering//KDD Cup Workshop at Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. California, 2007: 39-42

[8] Takács G, Pilászy I, Németh B, Tikky D. Investigation of various matrix factorization methods for large recommender systems//Proceedings of the 2nd KDD Workshop on Large-Scale Recommender Systems and the Netflix Prize Competition, 2008: 1-8

[9] Herlocker J, Konstan J, Riedl J. An empirical analysis of design choices in neighborhood-based collaborative filtering algorithms. Information Retrieval, 2002, 5(4): 287-310

[10] Herlocker J, Konstan J, Terveen L, Riedl J. Evaluating collaborative filtering recommender systems. ACM Transactions on Information Systems, 2004, 22(1): 5-53

LUO Xin, born in 1983, Ph. D. candidate. His research interests focus on data mining and data vitalization.

OUYANG Yuan-Xin, born in 1978, Ph. D., lecturer. Her research interests include data mining and workflow management.

XIONG Zhang, born in 1956, professor, Ph. D. supervisor. His current research mainly concentrates on smarter city and data vitalization.

YUAN Man, born in 1987, Ph. D. candidate. His research interests focus on smarter city.

Background

Modern online retainers suffer from the problem of how to help their customers picking up personal favorites from mountains of choices. Recommender systems that can recommend products according to individual tastes have emerged to address this problem since the early—1990s. Because recommender systems can automatically match customers with their potential favorites, they can offer an additional dimension to customer experience, and greatly improve customer satisfaction. So this thriving subfield of data mining is becoming more and more attractive to online merchants. *k*-Nearest-Neighborhood (*k*NN) based Collaborative Filtering (CF) is a widely used approach to implementing personalized recommender systems. However, there is a dilemma to decide the value of k in *k*NN based CF: a too small value of k will lead to poor recommendation performance, whereas a too large one will cause unacceptable computational complexity. Early research advocates that in *k*NN based CF, a suitable value of k should lie in the range of [30, 60]; and if k grows exceeding this range, the performance of the recommender will hardly improve. Pioneering researchers also find that the credibility of the Rating Similarity (RS) (the metric deciding the *k*NN sets of the involved entities) usually relies on the count of the supporting observations: RS based on few observations is always proved to be highly responsible for terrible recommendations. This work firstly empirically validated that the suitable value of k in *k*NN based CF was affected by the number of the totally involved entities, and then focused on improving the quality of the *k*NN sets in *k*NN based CF for providing high recommendation performance as well as maintaining suitable *k*NN set size. To achieve this objective, a novel *k*NN metric named Similarity Support (SS) was proposed. By combing RS and SS during the *k*NN building process, the authors designed a series of strategies for optimizing the *k*NN based CF. Due to the improvement on the quality of *k*NN set brought by SS, CF adjusted by these strategies turned out to be superior to original *k*NN based CF in term of both recommendation performance and computational complexity.

This work is part of the Exploratory Research Topic of the State Key Laboratory of Software Development Environment of P. R. China, and is fully supported by Fundamental Research Funds for the Central Universities of P. R. China under grant YWF-10-02-012.