

MOOC 中基于二分图推荐的同伴互评系统优化^{*}

何升¹, 邓伟林², 肖体斌³

(1. 广东轻工职业技术学院 计算机系, 广州 510300; 2. 中山大学 信息科学与技术学院, 广州 510006; 3. 广东工业大学 机电工程学院, 广州 510006)

摘要: 针对 MOOC 课程平台中作业反馈不及时的问题, 就同伴互评系统的准确性与可靠性优化进行了研究。为了达到为作业推荐合适评阅人的目的, 在综合考虑作业评阅人的评阅意愿、评阅能力和评阅双方作业相似度等多种因素的基础上, 建立了作业评阅人推荐模型; 引入二分图匹配理论来求解作业评阅人的任务均衡问题, 设计了与之相适应的最优均衡匹配算法。实验表明, 算法在评阅人的工作量均衡和作业评阅人的准确推荐等方面均取得了较好的效果, 优化提高了同伴互评系统的准确性和可靠性。采用经互评算法优化的同伴互评系统, 可以改善 MOOC 平台的整体满意度。

关键词: 大规模开放在线课程; 二分图; 同伴互评; 推荐; 均衡分配

中图分类号: TP391; TP182 文献标志码: A 文章编号: 1001-3695(2016)05-1399-04

doi: 10.3969/j.issn.1001-3695.2016.05.027

Peer review system optimization based on bipartite graph recommendation in MOOC

He Sheng¹, Deng Weilin², Xiao Tabin³

(1. Dept. of Computer, Guangdong Industry Technical College, Guangzhou 510300, China; 2. School of Information Science & Technology, Sun Yat-sen University, Guangzhou 510006, China; 3. School of Electro-mechanical Engineering, Guangdong University of Technology, Guangzhou 510006, China)

Abstract: On the question of homework correcting not in time in MOOC, this paper researched how to improve the accuracy and reliability of peer review system. In order to recommend right reviewer for every homework, integrated into account the willingness and ability of reviewer as well as homework similarity and other factors. It built the model of reviewer recommendation. Secondly, it designed an optimal balanced matching algorithm with modified bipartite graph matching method to meet the need of reviewer work balancing and recommending right reviewer for every homework. Experimental results show that, results about reviewer work balancing and reviewer recommendation accurately are satisfactory. It can improve the overall satisfaction of MOOC platform through apply peer algorithm to optimize peer reviewer system.

Key words: massive open online course (MOOC); bipartite graph; peer reviewer; recommendation; equalized assignment

2012年,一种新的网络在线学习平台——大规模开放在线课程(massive open online course, MOOC)由于其免费、不限人数、随时随地接入等特点而在全球兴起,国内外一些著名高校将许多优质课程搬上 Coursera、EDX 等 MOOC 平台,吸引了世界各地成千上万的学习者注册参与学习。2013年后,更多高校基于 EDX 的开源代码,搭建并二次开发了自己的 MOOC 课程平台。MOOC 如火如荼的发展,在方便学习的同时,却因学习群体规模巨大而严重增加了教师的负担。例如斯坦福大学开设的《人工智能》课程有 16 万多名学习者注册参与^[1],学习者众多而带来庞大的作业评阅任务,单靠教师已无法完成。同伴互评是解决此问题的一种有效作业评阅方式,它鼓励学习者参与教师的作业批阅活动,既可减轻教师负担,又可加强学习的交互与反馈^[2]。现在许多机构在大力研究同伴互评系统,互评效果优化已经成为 MOOC 的研究重点。

1 相关工作

当前的同伴互评系统由于功能设置简单,出现“学习者不

愿评价、随意评价”等现象。文献[3]针对随意评价问题,提出了 WebPA 算法用来减少互评中随意评价对最终评分的影响,但是其参数设置缺少理论支持和不可解释。文献[4]研究了影响评分准确度可靠度的因素,构建了三种同伴互评的概率统计模型;虽然其参数取值有理论模型的支持,但是参数权重依据学生的历史评分偏差来计算,没有考虑学习者的评价意愿与评价能力对最终评分的影响,只能在一定程度上提高互评系统的性能。文献[5]研究了学术会议系统的论文分配问题,对本文有很多启发和帮助;论文评审分配和作业互评都是开放环境下不同人员意见的融合与匹配,但在论文评审分配中作者与专家的意见是不对等的,专家意见起支配作用;而在互评系统中作业提交者与评阅人双方是对等的,双方之间的匹配受到两方的注册信息和学习情况等多种因素影响。文献[6]针对信息服务体验质量与评价问题,提出了基于隐马尔可夫的用户体验质量模型和基于随机模型的评价方法。文献[7]提出了基于二分图匹配的网络服务选择模型与算法。本文从文献[6,7]中得到了灵感启发,研究了作业评阅人推荐模型的构建方法和

收稿日期: 2015-04-15; 修回日期: 2015-06-03 基金项目: 国家自然科学基金资助项目(61033010); 国家科技重大专项基金资助项目(2008ZX10005-013); 广东省教育厅“十二五”规划项目信息技术专项基金资助项目(13JXN051)

作者简介: 何升(1971-),男,湖北红安人,副教授,硕士,主要研究方向为智能算法、智能信息处理(Guangdvoica@163.com); 邓伟林(1980-),男,广东韶关人,讲师,博士,主要研究方向为计算智能、智能控制; 肖体斌(1976-),男,湖南株洲人,教授,博士,主要研究方向为智能控制。

互评算法的设计。

提高互评效果的关键是为每份作业推荐合适的评阅人。本文综合考虑作业提交者与评阅人双方在知识水平、学习情况上的匹配,以及评阅人的评阅意愿与能力是否满足作业批阅的要求。为了计算这些影响双方匹配因素的权重,提出了评阅人推荐模型。为了从整体上提高互评的效果,将二分图匹配理论用于评阅人推荐问题,设计了适用于求解互评系统评阅人推荐问题的最优均衡匹配算法,从使评阅人工作量均衡和作业评阅人的准确推荐两个方面来获取整体上的最优匹配,从而使得互评系统达到较好的学习者满意度。

2 评阅人推荐模型

2.1 基本模型

一般互评系统中每份作业有 3~5 位评阅人,最终评分为各位评阅人的评分综合,同时每位评阅人被分配 3~5 份作业,评阅人按评分标准打分。作业与评阅人之间的分配不是简单的随机分配,而是双方属性关系最大满足的分配。因此,作业评阅人分配问题可转换为有约束的二分图最优匹配问题。二分图匹配理论用于求解推荐问题可以减少数据稀疏、算法可扩展性差等问题的影响,是比较新颖的推荐方法^[8]。评阅人推荐二分图匹配问题的形式化表示如图 1、2 所示。

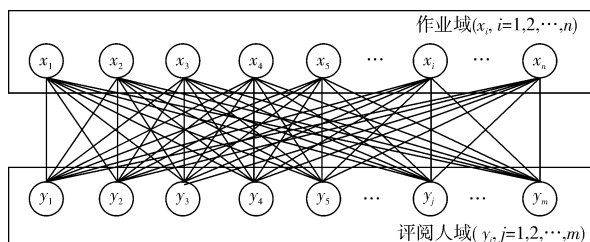


图 1 评阅人推荐之前的完全匹配

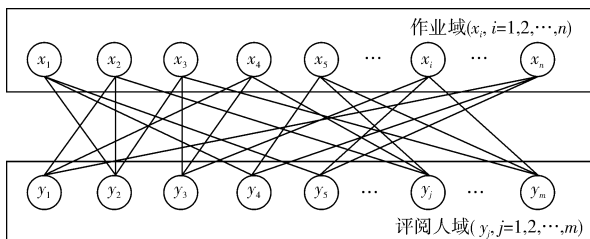


图 2 作业与评阅人的一个匹配结果

- 作业数 $n >$ 评阅人数 m (n, m 为自然数);
- 每一份作业由 t 名评阅人批阅 ($1 \leq t < m$, t 为系统设定的自然数);
- 每名评阅人批阅不超过 q 份作业 ($1 \leq q < n$, q 为系统设定的自然数)。

图 1 所示为评阅人推荐之前作业与评阅人之间的完全匹配。图中 x_i 表示第 i 份作业, y_j 表示第 j 名评阅人, 边表示作业与评阅人之间的关联权重。图 2 所示为向每份作业推荐三名评阅人, 每名评阅人共批阅三份作业的最大权匹配结果, 即最优匹配解, 图中各边权重之和为所有匹配解中的最大值。

2.2 评阅人与作业之间的匹配权重

图 1、2 中各条边的权重计算需要其正相关的因素。相关研究指出, 互评效果与作业评阅所耗费的时间、作业提交人、评阅人三方面因素有关^[4]; 知识水平、教育背景和思维方式接近的学生之间容易发生交互学习, 他们参与互评能够提高互评的

准确性^[4,9]; 成绩相差不大的学生之间互评能够使评阅偏见降到最低^[4]。可见: a) 知识水平是影响互评的重要因素, 它包括学习者的学历、专业、是否学过类似课程以及客观题得分等; b) 评阅人的评阅意愿也是权重的重要因素, 它由学习者课程注册时的意愿调查、参评次数、参加课程讨论的次数等决定; c) 学习者的评阅能力是进行互评的必要条件, 它与是否参加过互评训练、历史打分偏差等因素有关; d) 如果学习者双方的作业相似度高, 他们之间的互评意愿更高、互评结果更准确。此外, 据文献[4]的调查统计, 一次批阅所耗费的时间是一个重要因素, 一般一次 20 min 为宜。

综合考虑上述因素, 本文选取评阅人知识水平 (kh)、评阅意愿 (aw)、评阅能力 (ac)、作业相似度 (hs) 和一次批阅的时间耗费 (ts) 五个参数来计算作业与评阅人的匹配权重。先作如下定义:

定义 1 作业域。设集合 $\bar{X} = \{x_1, x_2, \dots, x_n\}$, $n \in R$, 其中 x_i 表示第 i 份作业, 每份作业对评阅人的需求为 $\tilde{x}_i^R = (x_i^{kh}, x_i^{aw}, x_i^{ac}, x_i^{hs}, x_i^{ts})$ 。

定义 2 评阅人域。设集合 $\bar{Y} = \{y_1, y_2, \dots, y_m\}$, $m \in R$, 其中 y_j 表示第 j 位评阅人, 评阅人的评阅能力与意愿状况 $\tilde{y}_j^p = (y_j^{kh}, y_j^{aw}, y_j^{ac}, y_j^{hs}, y_j^{ts})$ 。

定义 3 匹配权重。评阅人 j 满足作业 i 批阅需求的匹配权重。当评阅人的评阅能力满足作业的需求时, 作业与评阅人的匹配权重是上五个参数权重的迭加, 即

$$M_{ij} = \begin{cases} 0 & x_i^{ac} > y_j^{ac} \\ \partial_{kh} \cdot w_{kh} + \partial_{aw} \cdot w_{aw} + \partial_{ac} \cdot w_{ac} + \partial_{hs} \cdot w_{hs} + \partial_{ts} \cdot w_{ts} & x_i^{ac} \leq y_j^{ac} \end{cases} \quad (1)$$

其中: ∂_{kh} 、 ∂_{aw} 、 ∂_{ac} 、 ∂_{hs} 、 ∂_{ts} 为各参数的影响占比, w_{kh} 、 w_{aw} 、 w_{ac} 、 w_{hs} 、 w_{ts} 为各参数的独立权重。

2.2.1 各参数的独立权重计算

前面的模型考虑了五个参数对匹配的影响, 其中每个参数的权重计算需考虑各自的分配原则。经过分析, 确定如下分配原则:

- 知识水平
 - 知识水平相近的学习者可以互评;
 - 知识水平高的人可以评阅知识水平低的人的作业;
 - 拥有相同地域、语言、教育背景的学习者可以互评;
 - 知识水平高低通过专业、学历、是否学过类似课程、客观题得分等来计算)。
- 评阅能力
 - 评阅人不能评阅自己的作业;
 - 评阅人参加并且通过了互评训练;
 - 评阅人历史评分偏差低于一定数值。
- 评阅意愿
 - 学习者课程注册信息满足评阅意愿调查;
 - 评阅人在课程论坛主持讨论的次数接近一定数值;
 - 评阅人被认定随意评价次数满足系统要求值。
- 作业相似度
 - 评阅人自己的作业与待批阅作业的相似度达到一定数值。

(作业相似度的计算采用文献[10]中介绍的方法)。

- 一次作业批阅的时间耗费

一次时间耗费是否接近某个设定值(它的采集借助评阅人的信息调查和系统测定)。

以上分配原则是计算各参数权重的依据, 各原则的取值设

定在 0~5,取值越高表示参数的权重越大;0 表示不匹配,5 表示权重最大必定匹配。

2.2.2 各参数对匹配权重的影响占比计算

由于各参数对互评结果的作用不一样,各参数独立权重对图 1.2 中每条匹配边权重的影响占比的计算方法也不一样,本文采用文献[7]中的方法来分别计算。

a) 区间型参数 ∂_{kh} 、 ∂_{hs} 。知识水平 kh 和作业相似度 hs 相近的学生互评效果较好,两者在一定区间(u_k, l_k)内取值。评阅人 j 对于作业 i 参数 kh 、 hs 的权重影响计算式分别为

$$\partial_{ij}^{kh} = \begin{cases} 1 & y_j^{kh} > u_k^{kh} \\ \frac{y_j^{kh} - l_k^{kh}}{u_k^{kh} - l_k^{kh}} & l_k^{kh} \leq y_j^{kh} \leq u_k^{kh} \\ 0 & y_j^{kh} < l_k^{kh} \end{cases} \quad (2)$$

$$\partial_{ij}^{hs} = \begin{cases} 1 & y_j^{hs} > u_k^{hs} \\ \frac{y_j^{hs} - l_k^{hs}}{u_k^{hs} - l_k^{hs}} & l_k^{hs} \leq y_j^{hs} \leq u_k^{hs} \\ 0 & y_j^{hs} < l_k^{hs} \end{cases} \quad (3)$$

b) 门限值参数 ∂_{ac} 。评阅人的能力 ac 必须大于某个门限值 t 才能参加互评,其计算式为

$$\partial_{ij}^{ac} = \begin{cases} 1 & y_j^{ac} \geq t \\ 0 & y_j^{ac} < t \end{cases} \quad (4)$$

c) 接近型参数 ∂_{aw} 、 ∂_{ts} 。评阅意愿 aw 和一次批阅的时间耗费 ts 参数一般要求与值 t 的偏离度越小越好,因为成绩差距不大的学生之间的互评偏见最小。它们的计算式为

$$\partial_{ij}^{aw} = \begin{cases} 1 & |y_j^{aw} - t| \leq \zeta \\ \frac{\zeta}{|y_j^{aw} - t|} & |y_j^{aw} - t| > \zeta \end{cases} \quad (\zeta \text{ 为偏离度阈值}) \quad (5)$$

$$\partial_{ij}^{ts} = \begin{cases} 1 & |y_j^{ts} - t| \leq \tau \\ \frac{\tau}{|y_j^{ts} - t|} & |y_j^{ts} - t| > \tau \end{cases} \quad (\tau \text{ 为偏离度阈值}) \quad (6)$$

2.3 匹配权重的归一化处理

最后,为了系统计算的准确性,消除各参数取值范围的影响,将评阅人 j 满足作业 i 的匹配权重 M_{ij} 进行如下归一化处理:

$$M_{ij}^* = \frac{M_{ij} - M_i^{\min}}{M_i^{\max} - M_i^{\min}} \quad (7)$$

从式(7)可以看出,评阅人 j 满足作业 i 的评阅要求的匹配权重 $M_{ij}^* \in (0, 1)$ 。当 $M_{ij}^* = 1$ 时,作业匹配到了最合适的评阅人;当 $M_{ij}^* = 0$ 时,作业与评阅人之间完全没有匹配关系。

3 评阅人推荐最优均衡分配算法

3.1 问题目标与分析

3.1.1 问题目标

设每份作业被不超过 t ($t < m$) 位评阅人批阅,每位评阅人批阅不超过 q ($q < n$) 份作业,由 2.1 节讨论和定义 3 可知问题目标是:

$$\max \sum_{j=1}^m \sum_{i=1}^n M_{ij}^* \quad \text{s.t. 评阅人约束条件} \sum_{i=1}^n x_i^q \leq y_j^q$$

3.1.2 问题分析

前面 3.1 节已为作业与评阅人推荐构建了二分图最优匹配模型,但是现有的二分图匹配算法不能直接用来进行问题求解^[11],因为:

a) 二分图两边域中的节点数严格相等,而受评阅人评阅能力的限制,一般评阅人数 $m \leq$ 作业数 n 。解决办法是:可以向

评阅人域中添加一些权重为 0 的节点,使两域节点数目相等。

b) 现有二分图匹配算法严格要求一对一匹配,但是互评系统是多对多匹配,每份作业被分配多名评阅人,每位评阅人被分配多份作业,以减少评分偏差。解决办法是:可以考虑先求解二分图最大权匹配解,然后依次将解中一条边的权重置为 0,得到新的最大权匹配解,最后把每次结果叠加而成最终的解。

c) MOOC 课程学习人数巨大,但是基于二分图论的深度广度优先搜索算法时间耗费太大而影响系统性能。解决办法是:采用机器学习中的聚类方法来加快匹配求解过程。由于知识水平接近的学生之间互评效果较好,可以先用 K-means 将学习者在知识水平等字段上聚类,然后根据聚类结果形成不同的学习者的分组,最后分别对各学习者分组同时并行求解最优匹配。

3.2 评阅人推荐最优均衡分配算法

3.2.1 算法原理与思路

根据前面的分析,算法思路是先用 KM 算法(Kuhn-Munkras, 1957)^[12] 求出评阅人推荐模型的最优匹配解 M ,然后依次将 M 中每条边的权重置为 0 (即评阅人 y_j 与作业 x_i 的匹配解除) 而得到一个子图 G_i ,接着运用匈牙利算法^[12] 寻找子图 G_i 的最佳匹配 M' ,再将匹配 M 与 M' 合并得到新的解 M^* , M^* 是在原来解 M 基础上为评阅人 y_j 分配了第二份作业 x_k 。重复此过程,直到 M 中的所有边都被如此处理完后,所有评阅人就都被分配到均衡数量的作业。在此过程中,如果一位评阅人的分配作业数达到设定值后,就不再为他分配作业;如果一份作业所分配的评阅人达到设定值后,就停止为它分配新的评阅人。

可能会出现的情况是: a) 评阅人匹配不到任何作业,此时可将作业的权重乘上一个松弛系数,适当降低匹配要求; b) 评阅人能够匹配所有的作业,即出现几个匹配权重相等的情况,此时可让系统在等权重的作业中随机分配; c) 无法为每个评阅人分配等量的作业,因为评阅人数与作业数的比值有时不能为整数,此时完全的任务均衡是做不到的,可以允许各位评阅人之间的作业数差为 1。

3.2.2 算法流程

算法的具体流程如下:

a) 分别在学习者数据集的知识水平、作业相似度属性上作 K-means 聚类。利用聚类结果将所有作业进行分组。

b) 构建每个分组作业与评阅人二分类(X, Y) 的二分图 G ; G 的节点数为 $\max(n, m)$, 计算图 G 中各边的匹配权重 ω_{ij} 。

c) 对作业节点 x_i 赋值 $L(x_i) = \max(\omega_{ij})$, 评阅人节点 y_j 赋值 $L(y_j) = 0$, $j = 1, 2, \dots, \max(n, m)$ 。

d) 运用 KM 算法求解图 G 的最优匹配 M , 保存 M 、 x_i (作业 i 被指定的评阅人数) 和 y_j (评阅人 j 所分配的作业数)。

e) 令 M 中的 $\omega_{ij} = 0$, 得到子图 G_i ; 否则转步骤 g)。

f) 运用匈牙利算法求 G_i 的最佳匹配 M' , 将 M 与 M' 叠加, 叠加结果保存为 M^* 。

g) 计算并保存 M^* 中 x_i, y_j 的值。若 $x_i < p$ (p 为评阅人阈值) 且 $y_j < q$ (q 为作业阈值), 转步骤 e); 否则转步骤 h)。

h) 系统为剩余作业与评阅人随机匹配, 算法结束。

3.2.3 算法分析

算法组合并修改二分图匹配的经典算法来求解同伴互评的多对多匹配问题,它基于 KM 算法先求解作业评阅人二分图

的最大权匹配解 M , 然后依次将匹配解中每条边权重修改为 0 后得到新子图, 再用匈牙利算法求出每个新子图的完备匹配 M' , 最后所有的完备匹配 M' 都与最大权匹配 M 叠加而得到问题的最终解。每次将匹配解 M 中的一条边权重修改为 0 就得到一个子图, 匈牙利算法必定能求出新子图的完备匹配, 新子图的完备匹配求解过程就是为该评阅人匹配下一份作业的过程。算法迭代过程中检查每位评阅人所匹配的作业数和每份作业所匹配的评阅人数是否达到设定的阈值, 如果达到就不再修改它们的匹配边权值。由于 KM 算法的时间复杂度为 $O(n^3)$, 而每位评阅人需匹配 q 份作业, 所以本文算法的时间复杂度为 $O(qn^3)$ 。此外, 由于在匹配之前进行了学习者聚类分组, 控制了每个互评组的人数规模, 使得在各个互评组上并行运行的最优均衡匹配算法的时间耗费大大减少。

4 实验过程与分析

实验在自行开发的广东轻工职业技术学院 MOOC 课程平台上进行, 它是根据 EDX 的开源项目代码^[13], 满足自身需求二次开发的 MOOC 平台。实验数据是选修《程序设计基础》课程的 516 名学生同伴互评数据, 学生来自全校 12 个专业的两个年级, 个体情况差别较大。

4.1 实验过程

为了加快处理速度, 系统根据学生知识水平差异情况进行层次聚类, 即依次在学生的专业、年级和客观题得分等属性上进行 K-means 聚类, 根据聚类结果将学生分成了 14 个组, 每组人数在 13~86 人。互评系统根据学生的注册信息、学习数据以及互评训练环节, 产生每个组内的评阅人集合, 每组评阅人占比在 84%~93%, 共计 473 名评阅人。然后在 14 个分组上同时并行运行评阅人推荐最优均衡算法, 有 464 份作业与评阅人被匹配, 剩下未被匹配上的作业由系统随机分配评阅人。最后同伴互评系统将学生互评打分与教师的打分进行了分析, 输出了学生互评与教师评价数据比较的统计图表。

4.2 实验结果分析

系统产生的学生互评与教师评分散点图、学生互评与教师评分一致性比较的 Bland-Altman 分析图以及学生各互评组间评分变异情况箱线图分别如图 3~5 所示。

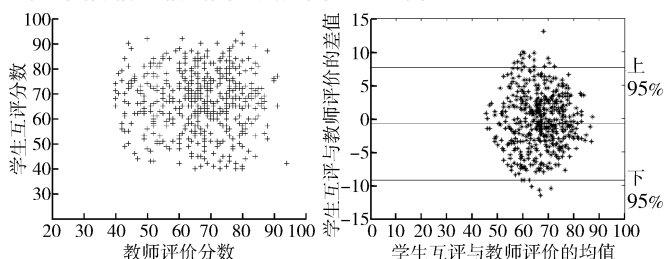


图3 学生互评与教师评分散点图

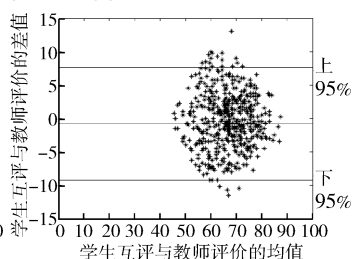


图4 学生互评与教师评价的一致性比较分析图

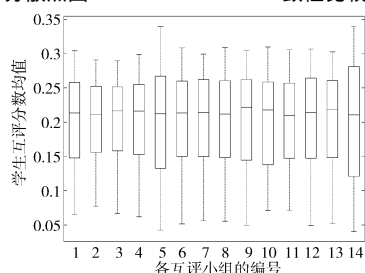


图5 学生各互评组评分变异图

图 3 的散点图中, 学生互评与教师评价分数大呈球形分布, 说明实验中学生互评评分的准确度与教师评分接近; 图 4 的 Bland-Altman 分析图中绝大多数点均落在上、下 95% 的置信区间内, 说明学生互评与教师评价这两种不同评分方法具有较好的一致性, 学生互评评分是可靠的。

图 5 的箱线图是将学生各组互评分数经归一化处理后绘制出来的, 它直观显示了评分数据的分布情况。图中除了个别箱体长度略大以及有些异常点之外, 各组的中位数线非常接近, 这说明各组打分高低虽略有不同, 但数据分布情况都是一样的。组与组之间的学生能力水平虽有差异, 但组内的分数分布却是一样的, 再次证明了水平接近的学生之间的互评效果较好。图中少量异常点的出现, 说明本文的做法尚不能完全消除学生随意评价的现象, 还有待进一步的改进与提高。如果能与文献[4]中的概率统计模型结合起来, 从提高匹配精准度与消除异常评分的影响两个方面着手, 互评效果将会更好。

5 结束语

本文从作业与评阅人之间准确匹配的角度, 提出了一种提高 MOOC 平台同伴互评效果的方法。综合考虑评阅人的评阅能力、评阅意愿以及作业批阅需求等多种因素, 并构建了评阅人推荐问题模型。运用二分图匹配理论来求解作业与评阅人之间的多目标任务均衡匹配问题, 设计了与问题相适应的算法并进行了实验分析与验证, 结果显示本文算法能够从整体上提高同伴互评的效果以及 MOOC 平台的学习满意度。

参考文献:

- [1] Martin F. Will massive open online courses change how we teach? [J]. Communications of the ACM 2012, 55(8): 26-28.
- [2] Kop R, Fournier H, Mak F. A pedagogy of abundance or a pedagogy to support human beings? Participant support on massive open online courses [J]. International Review of Research in Open and Distance Learning 2011, 12(7): 74-93.
- [3] Loddington S, Pond K, Wilkinson N. A case study of the development of WebPA: an online peer-moderated marking tool [J]. British Journal of Educational Technology 2009, 40(2): 329-341.
- [4] Piech C, Huang J, Chen Zhenghao, et al. Tuned models of peer assessment in MOOCs [C]//Proc of the 6th International Conference on Educational Data Mining. 2013.
- [5] Laurent C, Richard Z, Craig B. A framework for optimizing paper matching [C]//Proc of Uncertainty in Artificial Intelligence. 2011.
- [6] 林闯, 胡杰, 孔祥震. 用户体验质量 QoE 的模型与评价方法综述 [J]. 计算机学报 2012, 35(1): 1-15.
- [7] 袁尧, 张玉成, 董雯霞, 等. 基于二分图匹配的多业务流网络选择机制 [J]. 软件学报 2010, 21(6): 1378-1390.
- [8] 印桂生, 崔晓晖, 董红斌, 等. 量子协同的二分图最大权完美匹配求解方法 [J]. 计算机研究与发展 2014, 51(11): 2573-2584.
- [9] 许云红, 王如. MOOC 背景下基于推荐机制的提高同伴互评效果的研究 [J]. 现代远程教育 2014(5): 17-21.
- [10] 华秀丽, 朱巧明, 李培峰. 语义分析与词频统计相结合的中文文本相似度度量方法研究 [J]. 计算机应用研究 2012, 29(3): 833-836.
- [11] 张卫丰, 周毓明, 许蕾, 等. 基于匈牙利匹配算法的钓鱼网页检测方法 [J]. 计算机学报 2010, 33(10): 1963-1975.
- [12] 程兴国. 仿生算法的动态反馈机制及其并行化实现方法研究 [D]. 广州: 华南理工大学 2013.
- [13] Raju B. Optimization of LMS for improving user response time [D]. Bombay: Indian Institute of Technology 2013.