

改进的 E-learning 社区自组织算法

金晶, 陈清华, 罗恒

(上海交通大学计算机科学与工程系, 上海 200240)

摘要: 为了解决 E-learning 过程中缺乏对学习者的针对性指导的难题, 该文提出一种构建 E-learning 社区的改进算法。该算法采用 P2P 的架构, 通过改进的 Hebbian 学习法则不断加强具有类似兴趣的学习者之间的联系, 从而自动调整社区结构, 最终将具有类似兴趣的学习者组织在一起。实验结果证明, 与传统的 P2P 架构的 Hebbian 学习算法相比, 该算法达到了更好的社区建设质量和更高的社区建设速度。

关键词: Hebbian 学习; P2P 架构; E-learning 社区

Improved Self-organizing Algorithm in Constructing E-learning Communities

JIN Jing, CHEN Qing-hua, LUO Heng

(Department of Computer Science and Engineering, Shanghai Jiaotong University, Shanghai 200240)

【Abstract】 To solve the problem that learners are unable to get personalized instruction, an improved algorithm is proposed for constructing self-organized communities. This algorithm adopts P2P architecture, and strengthens connections between learners with similar interests based on improved Hebbian learning law. In this way, community structure is adjusted automatically, organizing learners with similar interests together eventually. Experimental results show that compared with traditional Hebbian-based P2P algorithm, this algorithm achieves better construction quality and higher construction speed.

【Key words】 Hebbian learning; P2P architecture; E-learning communities

1 概述

随着网络和多媒体技术的飞速发展, 使用 E-learning 的学习者面临着越来越严重的“信息过载”问题。解决这个问题一个有效方法就是过滤, 比如基于内容的过滤(CBF)^[1]和协同过滤(CF)^[2]。然而, 由于这些方法有一个中心化的架构并且扩展性不够好, 因此无法很好地适应 E-learning 的分布式开放环境。

文献[3]提出了一种基于文献[4]学习法则的 P2P^[5]推荐算法。这种算法根据学习者的反馈调整他们之间的关系, 从而通过几次调整将兴趣相近的学习者组织在一起。但是, 在实际应用中, 文献[3]的算法需要较长的调整时间, 且对于大规模的用户性能下降较快。本文在文献[3]算法的基础上提出一种改进的推荐算法。实验证明, 本文的算法在调整速率和扩展性方面的性能较原算法有所提高。

2 P2P 架构

一个 P2P 系统指的是为了完成一个特定的任务而通过分布式联合式的方式组织在一起共同运作的一组程序或系统^[6]。P2P 系统这种分布式以及自治的特征与 E-learning 的环境十分类似。因此, 改进算法采用 P2P 的架构来构建 E-learning 社区。

本文将每一个学习者终端抽象为一个自治的智能体。每一个智能体在功能上与其他智能体等同, 并且可以不经第三方而与其他智能体通信。因此, 相比于那些中心化架构的算法, 某些智能体非法的退出或者瘫痪对于整个系统的影响大大减小了。

每一个智能体拥有一些学习资源以及对这些资源的评分(分值范围为 0~1)。评分越高, 该智能体对这个资源的评价就

越高。如果 2 个智能体对于某个学习资源有着类似的评分, 那么假设它们对这个学习资源具有类似的学习兴趣。本文对学习者的分类就是基于他们兴趣的相似程度。

图 1 为算法的架构示意图。图中的智能体可以不通过中央服务器而与其他智能体通信, 它们构成了一个 P2P 网络。

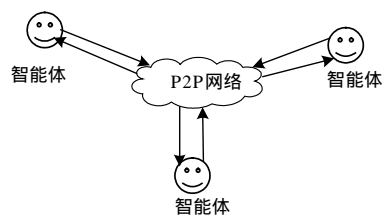


图 1 P2P 的系统架构

类似于 P2P 系统, 每一个智能体仅仅与一部分智能体直接相连, 并且只能从这些智能体处获得信息。这些直接与一个智能体相连的智能体称为该智能体的邻居。需要指出的是, 邻居的关系是非对称的。也就是说, A 是 B 的邻居不代表 B 是 A 的邻居。

算法定义了智能体的 2 种行为: 查询和推荐。某个智能体可能希望通过询问社区中其他智能体的意见来决定选择何种学习资源, 它将向它的邻居发出询问。这种行为称为“查询”。收到智能体查询请求之后, 这些邻居如果已给这个资源评分, 则将评分直接返回, 如果没有, 则将查询转发给它的

基金项目: 国家自然科学基金资助项目(60372078)

作者简介: 金晶(1983-), 女, 硕士研究生, 主研方向: 自组织学习; 陈清华, 硕士研究生; 罗恒, 博士研究生

收稿日期: 2007-11-15 **E-mail:** jingjin@sytu.edu.cn

邻居。在预先定义的转发次数之内，如果有一个智能体对资源已经评分，则将评分沿着转发的路径返回给最初发出查询的智能体。这个过程就叫做“推荐”。为了保证查询和推荐能够在有限时间内完成，当转发次数超过规定值时，便停止转发查询。

3 社区建设

社区建设包括 2 个关键步骤：信任权值调整和邻居调整。第 1 步修改代表智能体之间兴趣相似度的信任权值，而第 2 步根据修改后的信任权值改变智能体之间的关系。

3.1 信任权值调整

算法采用信任权值表示 2 个相邻智能体之间的相似度。具有相似兴趣的相邻智能体具有较高的信任权值。自组织社区通过调整智能体之间的信任权值逐渐建设起来。信任权值的调整算法必须使 2 个智能体对同一个资源评分相近时信任权值增大而评分差异较大时权值减小。这一点是通过应用 Hebbian 学习法则实现的。

令 a_i 表示智能体 i 。令 $w_{i,j}$ 为 a_i 和邻居 a_j 的信任权值。初始化时，每个智能体被随机地分配若干邻居并为它们之间的信任权值赋予随机值。

当 a_i 对资源 r 进行查询时，为了防止网络拥塞，算法仅对一部分邻居发出询问请求。这里，每一个邻居被赋予 1 个概率值，智能体根据邻居被选中的概率随机地选择此次查询将要查询的智能体。对 a_i 而言，选择邻居 a_j 的概率记为 p_j^i ，其定义如下：

$$p_j^i = \frac{w_{i,j}}{\sum_j w_{i,j}} \cdot M, p_j^i = \begin{cases} p_j^i & \text{if } p_j^i < 1 \\ 1 & \text{if } p_j^i > 1 \end{cases} \quad (1)$$

其中， M 为一次查询中询问的邻居最大数目。

经过计算， a_i 将查询发给 t 个邻居。根据前面对查询和推荐行为的定义， a_i 将得到小于等于 t 个评分值。当 a_i 读完查询的资源时，它也会对这个资源给予评分。令 r_k^i 为 a_i 对资源 k 的评分。信任权值 $w_{i,j}$ 将作如下调整：

$$\Delta w_{i,j} = \eta \cdot r_k^i r_k^j \quad (2)$$

其中， η 代表学习速率。

学习速率 η 在 Hebbian 学习中具有重要的意义。与传统 Hebbian 学习不同的是，这里的 η 是动态的，因为不同的评分相近程度需要不同幅度的权值调整。

基于 Hebbian 学习法则，在相似的评分之后信任权值增加而相反的评分之后信任权值减少。这里，定义门限 T_2 确定权值调整的方向。此外，期望差异很大的评分结果引起更大幅度的权值减少，而相似度很高的评分引起更大幅度的权值增加。因此，提出如下参数：

$$p_1 = \begin{cases} 1 + |1 - |r_k^j - r_k^i|| & \text{if } |r_k^j - r_k^i| < T_1 \\ 1 & \text{if } T_1 < |r_k^j - r_k^i| < T_2 \\ -1 & \text{if } T_2 < |r_k^j - r_k^i| < T_3 \\ -1 - |r_k^j - r_k^i| & \text{if } |r_k^j - r_k^i| > T_3 \end{cases} \quad (3)$$

此外，当 2 个评分都特别高或者低，即 2 个学习者都强烈喜欢或者厌恶这个学习资源时，应该对这种情况予以强调。

$$p_2 = 1 + |r_k^j + r_k^i - 1| \quad (4)$$

最后，参数 β 用以控制整体的学习速率，以保证学习过程的收敛性。

由此：

$$\eta = \beta \cdot p_1 \cdot p_2 \quad (5)$$

信任权值通过下式调整：

$$w_{i,j} = (1 - \gamma)w_{i,j} + \Delta w_{i,j} \quad (6)$$

其中， γ 表示衰减速率。

3.2 三列表结构的邻居调整

为了更好地适应 E-learning 的复杂环境，在 F. Yang 算法的基础上增加了一个潜在陌生人列表。也就是说，将从邻居中淘汰下来的那些智能体放入潜在陌生人列表，而不是直接将其划分为陌生人。这样，每个智能体将有 3 个列表：邻居列表，潜在邻居列表和潜在陌生人列表。

在一次查询行为中，有些非邻居的智能体可能返回与查询智能体相似的评分，这些智能体极有可能在将来成为查询智能体的邻居，因此，定义潜在邻居列表存放这些智能体。每次查询之后，将这些潜在的邻居加入到潜在邻居列表。如果列表已满，则将邻居和潜在邻居列表中信任权值最低的那个智能体移除。

在有些情况下，邻居对于某些资源的评分可能与查询智能体差异很大，但是它们实际上有着相同的兴趣。将这些邻居直接移除可能会导致系统的不稳定性。因此，定义潜在陌生人列表来存放这些过去的邻居。这个列表充当了陌生人与邻居之间的一个缓冲。图 2 为这种三列表结构的示意图。更多的细节见文献[3]。

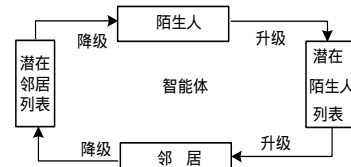


图 2 智能体的三列表结构

4 实验

笔者在上海交通大学网络教育学院进行了算法的实验。

首先在 JADE 平台上实现了算法，然后让学生使用系统进行学习。学生通过系统的客户端加入 JADE 模拟的社区，并在社区内进行交流。模拟社区的客户端程序分为几个部分。左边的面板显示当前学生拥有的资源(已评分和未评分)，左下角是该学生的所有邻居，右侧面板是当前选中资源的详细情况，学生可以通过右下角的下拉列表对资源进行评分。

将所有学习者设定为一个二维平面，每个学习者都有一个坐标，坐标之间的欧式距离表征了学习者之间信任权值的大小。这样，便可以用图形化的方法表现 E-learning 社区的自组织过程。

在初始状态下，用户的邻居由系统随机生成，用户随机散布于平面之中，用户与邻居之间的信任权值比较低，联系不够紧密。经过一段时间的学习，用户的邻居进行了调整，用户与邻居之间的联系加强，信任权值也随之上升。

在学习的过程中，后台程序记录了相关的数据。经过一段时间，将记录下来的数据用 Matlab 进行处理。为了比较算法的性能，对 F. Yang 的算法进行了相同的实验。

算法的性能通过 2 个指标来衡量：评分的平均标准差以及社区稳定前的最少查询次数。前者衡量建立的社区的质量，后者衡量自组织的速度。

2 种算法的比较结果如图 3 和图 4 所示。在图 3 中，当有 500 个用户时，标准差迅速地减小，并且在大约 3 000 次查询后低于预定的门限。从图 4 中可以清楚地看到，本文的算法在用户数目增加时最少查询次数增加得比较缓慢。从这些结果可以看出，改进的算法比传统算法性能更好。

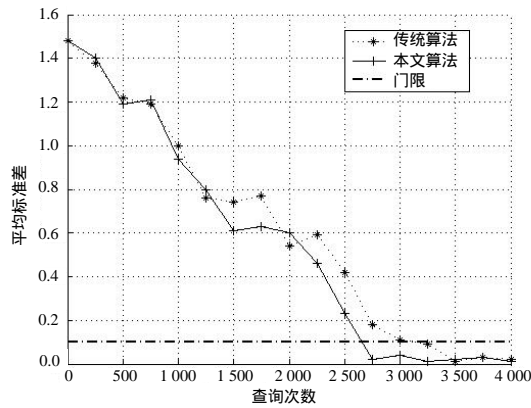


图3 标准差比较

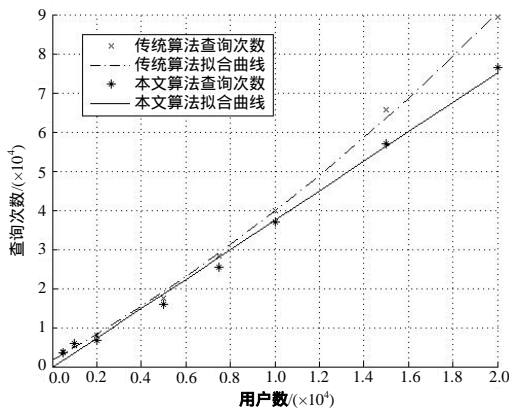


图4 查询次数比较

(上接第 253 页)

参考文献

[1] Shen J P, Lipasti M H. 现代处理器设计——超标量处理器基础[M]. 张承义, 译. 北京: 电子工业出版社, 2004.
 [2] Su A P, Chen Robert. Applying ESL in a Dual-core SoC Platform Designing[C]//Proc. of IEEE International SoC Conference. [S. 1.]:

(上接第 257 页)

口部分。主函数管理协议转换,而协议转换则连接着串口与 CAN 总线部分。其中协议转换程序完成 RS232 到 CAN ,CAN 到 RS232 的数据转换,通过串口将一帧固定长度的数据发送给微处理器,微处理器将数据发往 CAN 通道,CAN 通过自接收,将收到的数据发往串口。

CAN 接口通信程序分为 3 部分:CAN 初始化,数据发送,数据接收^[5]。CAN 初始化主要是设置 CAN 的通信参数。发送数据程序把数据存储区中待发送的数据取出,组成信息帧,并将主机的 ID 地址填入帧头,然后将信息帧发送到 CAN 控制器的发送缓冲区。在接收到主机的发送请求后,发送程序启动发送命令。信息从 CAN 控制器发送到总线是由 CAN 控制器自动完成的。信息从 CAN 总线到 CAN 控制器的接收缓冲区也是由 CAN 控制器自动完成的。接收程序只需从接收缓冲区读取信息,并将其存储在数据存储区。

串口程序设计包括串口初始化、发送和接收以及中断程序。在该系统中,下位机的串口和 CAN 都工作在查询状态。主循环程序工作在查询模式,当 CAN 总线有数据时,微处理器将其发往串口,当串口有数据时,微处理器将其发往

5 结束语

本文提出了一种基于 Hebbian 学习法则和 P2P 架构的算法解决 E-learning 学习社区自组织建设问题,并实现了一个系统验证算法的有效性。结果表明,改进算法较之于传统算法在稳定性和可扩展性方面均有更好的表现。此外,由于算法具有 P2P 的系统架构,因此适用于 E-learning 这种开放的分布式环境。

参考文献

[1] Belkin N J, Croft W B. Information Filtering and Information Retrieval: Two Sides of the Same Coin[J]. Communications of the ACM, 1992, 35(2): 29-38.
 [2] Sarwar B, Karypis G, Konstan J, et al. Item-based Collaborative Filtering Recommendation Algorithms[C]//Proceedings of the 10th International Conference on World Wide Web. [S. 1.]: IEEE Press, 2001: 285-295.
 [3] Yang Fan, Shen Ruimin, Han Peng. A Dynamic Self-organizing E-learner[C]//Proc. of the 16th Australian Joint Conference on Artificial Intelligence. Perth, Australia: [s. n.], 2003: 590-600.
 [4] Hebb D O. The Organization of Behavior[M]. New York, USA: Wiley, 1949.
 [5] Cohen E. Search and Replication in Unstructured Peer-to-Peer Networks[C]//Proc. of the 16th International Conference on Super Computing. New York, USA: ACM Press, 2002: 84-95.
 [6] Xu Linhao, Dai Chenyun, Cai Wenyuan, et al. Towards Adaptive Probabilistic Search in Unstructured P2P Systems[C]//Proc. of the 6th Asia Pacific Web Conference. Hangzhou, China: [s. n.], 2004: 258-268.

IEEE Press, 2006.

[3] 何 虎, 孙义和. 面向寄存器的流水线处理器建模及验证方法[J]. 半导体学报, 2003, 24(1): 98-102.
 [4] ARM Ltd.. MaxSim Developer's Guide[Z]. 2006.
 [5] Texas Instrument Incorporated. DSP Reference Sets[Z]. 2001.

CAN 总线。

5 结束语

该设计在实验室实验阶段通过对部件以及系统功能的简单测试,可以达到设计要求,在后续设计阶段把程序模块化,各方面进行优化可以使之产品化,有一定的研究价值。另外,限于 RS232 总线的性能瓶颈 CAN 总线利用率不高,每秒钟小于 300 帧。

参考文献

[1] 邬宽明. CAN 总线原理和应用系统设计[M]. 北京: 北京航空航天大学出版社, 1996.
 [2] 电狂. RS-232-C 详解(一)[Z]. (2006-10-20). <http://bbs.gongkong.com/detail.asp>.
 [3] 周立功. 带 CAN 控制器的单片 8 位微控制器——P8XC591[M]. 北京: 北京航空航天大学出版社, 2005.
 [4] PCA82C250 Data Sheet[Z]. Philips Semiconductors, 1994.
 [5] 吴慧英. 基于 CAN 总线的工业测控系统应用研究[D]. 北京: 北京航空航天大学, 1993-03.

