

# 一种自适应的个性化学习序列生成研究

蒋艳荣 韩坚华 吴伟民

(广东工业大学计算机学院 广州 510006)

**摘要** 根据学生的个性特点生成合适的学习序列是智能教学系统实现的关键,也是系统智能性的重要体现。其难点在于既要考虑知识点之间的逻辑联系,又要适应学生个性化的学习特点。提出了基于知识点之间内在关系指导下的学习序列生成算法,讨论了学生学习水平和用户个性的建模和计算以及对学习序列的调整和优化的方法。采用学习窗口作为活动组织的单元,以适应具有不同学习能力的学生,采用学习模式描述知识内容的组织方式及对应学习方式的不同。原型系统的应用表明了所提方法的有效性,系统具有较强的适应性,能够满足学生的个性化学习需求。

**关键词** 学习序列,自适应,个性化,智能教学系统

**中图分类号** TP391 **文献标识码** A

## Adaptive Approach to Personalized Learning Sequence Generation

JIANG Yan-rong HAN Jian-hua WU Wei-min

(School of Computer, Guangdong University of Technology, Guangzhou 510006, China)

**Abstract** To generate suitable learning sequence according to the student's individual characteristic is a key to build an intelligent tutoring system, and is also an exhibition of intelligence for the system. The difficulty is that both the relationship between the knowledge items should be considered, and the personalized learning characteristics of students should be adapted. In this paper, a learning sequence generation algorithm, which is under the guidance of relationships between knowledge, was proposed. The computation of learning level and the modeling of student personality were described. Based on these, the learning sequence can be optimized. And the detailed adjust and optimization methods were discussed in detail. Learning window was proposed to describe the learning contents and to adapt to the students with different learning abilities, and the learning model was used to distinguish the organization of contents and the corresponding learning styles. The application of the prototype indicates the effectiveness of the proposal, and the prototype system is adaptive to meet students' individual learning needs.

**Keywords** Learning sequence, Adaptability, Personality, Intelligent tutoring system

## 1 前言

近年来,具有适应性的智能教育系统(Adaptive and Intelligent Educational System, AIES)成为远程教育研究的一个热点<sup>[1-3]</sup>。与智能教学系统(ITS)<sup>[4]</sup>相比,AIES系统更多地强调适应性,能够根据学生的学习水平、个人偏好等学习特点,调整教学内容和教学方式,以适应每位学生的特点。Iglesias等<sup>[1]</sup>将AIES看成是ITS与自适应超媒体系统(AHS)<sup>[5]</sup>的一个结合,通过引入人工智能技术实现因材施教。

AIES系统实现的一个关键问题是如何根据学生的学习特点生成合适的学习序列,并在环境改变的情况下动态地进行适应性调整。学习序列指定了最适合学生学习需要的知识单元和学习任务(例子、问题、习题等)的序列<sup>[1]</sup>,其生成的难点在于既要兼顾知识点之间的逻辑联系,又要适应学生个性化的学习特点。例如,前提关系就是知识点之间常见的关系,

用以约束在学习某知识点之前,需要学习的先导知识;而对于学生的学习个性来说,不同学生的学习能力、学习水平和个性偏好等不同,带来在学习进度、学习内容的组织、学习方式的调整等多方面相应的改变。因此,如何综合考虑以上因素,生成合适的个性化的学习序列成为实现学习系统智能性和适应性的一个关键。

然而,现有的远程教育系统通常将预设的动作计划(action plan)作为学习序列,遵循预设的计划将教学内容呈现给每一位学生,而忽视了学生之间的差异性<sup>[6]</sup>,如早期的PMS系统<sup>[7]</sup>等,其缺点在于灵活性和适应性不够,且依赖于固有的教学计划。在引入机器学习技术后,有些学者试图通过试探与反馈信息来自动生成学习序列。例如,Iglesias等<sup>[1]</sup>提出采用强化学习对不同的学习序列进行尝试,通过与虚拟学生的交互来获得反馈,进而训练系统以获得最优的学习序列。其缺点在于没有考虑知识点之间的内在关系(如前提关系等),

来稿日期:2012-09-30 返修日期:2013-01-12 本文受国家自然科学基金项目(61142012),广东省科技计划项目(2012B010600014, 2012B010500025)资助。

蒋艳荣(1976—),男,博士,讲师,主要研究方向为机器智能、智能教学系统, E-mail: yriang@gdut.edu.cn; 韩坚华(1955—),女,教授,主要研究方向为人工智能; 吴伟民(1956—),男,教授,主要研究方向为可视化计算。

而这些关系可用于指导学习序列的生成,并且系统的训练成本较高;此外,系统的性能还依赖于对虚拟学生建模的准确性。其它类似的研究,还包括采用 Kohonen 网络<sup>[3]</sup>、贝叶斯网络<sup>[8]</sup>等技术以试图获得最优的学习序列。

已有研究存在的主要问题是:1)侧重于局部的优化技术,未综合考虑与学习相关的各种因素,如知识点之间的关系、用户的学习特点及差异性等;2)对满足学生的个性需求、适合学生学习特点的个性化学习序列生成的研究还远远不足,而这对提高学习系统的智能性和适应性较为关键。为此,本文提出一种采用知识点之间的内在关系作为启发信息,并结合用户个性与学习反馈进行适应性调整的个性化学习序列生成方法,其优点是,学习序列中的知识点之间的次序满足其内在关系的约束,同时又能够适应学生的学习特点。

## 2 知识表示及学习序列生成

### 2.1 知识的表示与建模

**定义 1(知识点)** 学习系统中的知识点可以表示为一个多元组: $K=(ID, name, des, KWS, dif)$ ,其中  $ID$  为知识点的唯一编号; $name$  为知识点的名称; $des$  为知识点的描述,包括该知识点的定义、阐述等,来源于领域知识; $KWS$  为知识的关键词集,是对该知识点的高度概括; $dif \in [0, 1]$  为知识点的难度系数。

一个学习系统中的知识点可以是一门课程的一章、一节、一小节、一个算法等。例如,“二叉树的先序遍历”就是一个知识点, $k=(K6121, \text{“二叉树的先序遍历”}, \text{“二叉树先序遍历的定义是...;先序遍历算法的步骤是...;等”}, \{ \text{“二叉树、先序遍历”}, 0.6)$ 。

学习系统中的知识点不是孤立的,而是相互关联的。我们将知识点之间的关系概括为:subtopic-of, precedence-of, association-of 3 种。对于一个知识点来说,通常可以分解为几个小的子知识点,而子知识点又可以分解为更小的子知识点,依次类推。例如,“二叉树的遍历”可以分解为“先序遍历”、“中序遍历”和“后序遍历”3 个子知识点。将这种关系称为 subtopic-of 关系,记为  $s(K_i, K_j)$ ,表示  $K_i$  是  $K_j$  的子知识点。如果对某知识点  $K$  进行逐步分解,则最终可以得到一棵根结点为  $K$  的树。

**定义 2(知识树, Knowledge Tree)** 从知识点  $K$  分解得到的树,称为知识树,记为  $T(K)$ 。树中的孩子结点是双亲的子知识点,满足 subtopic-of 关系。显然,若  $K_2$  是  $K_1$  的祖先,则也存在这样的关系: $s(K_1, K_2)$ 。为了区分,将孩子双亲关系进一步记为  $sc(K_1, K_2)$ ,表示  $K_1$  是  $K_2$  的孩子。对于树中不能分解的叶子结点,将其称为原子知识点(atom knowledge),它是学习系统中知识的最小单元。

为了满足学习的需要,每个知识点可以与一个或者多个辅助资源相关联,同时一个资源也可以与多个知识点相关联。这些资源包括该知识点的实例、测试和问题等。例如每个知识点有对应的测试,同时某一个测试可以与多个知识点相关联。我们称这种附属性的关系为 association-of。它不参与逻辑层面的学习序列的生成,一般用于学习序列的具体实现(如学习网页的生成)。

前提关系(precedence-of)一般用于描述两个知识点之间

的前提依赖关系,可用于调整学习序列中知识点之间的次序,保证在学习某知识点时已经对其先导知识进行了学习和了解。这种前提依赖关系具有传递性、单反性。

**定义 3(precedence-of 关系)** 若某知识点  $K_1$  可用于定义、描述或解释另一知识点  $K_2$ ,则称  $K_2$  依赖于  $K_1$ ,或  $K_1$  是  $K_2$  的前提,记为  $p(K_1, K_2, q)$ ,其中,  $q \in [0, 1]$  为依赖强度,对于直接前提关系,有  $q=1$ 。在学习系统中,为了获得较好的学习效果,若要学习前提知识点  $K_p$ ,有时需要进一步学习与  $K_p$  相关联的知识点,并且满足以下规则。

**规则 1** 若存在前提关系: $p(K_p, K_d, 1)$ ,且  $s(K_p, K_f)$  为真,则有: $p(K_f, K_d, q_{fd})$ ,其中依赖强度  $q_{fd}=1/2^N$ ,  $N$  为结点  $K_p, K_f$  之间的距离。若  $K_f$  存在其它孩子结点,且  $q_{fd}$  的值大于阈值  $\mu$ ,那么对所有的  $K_i (i=1, \dots, n, K_i \neq K_p)$ ,  $s(K_i, K_f)$ ,且  $K_i$  与  $K_f$  的距离小于阈值  $\rho$ ,则有  $p(K_i, K_d, q_{id})$ ,  $q_{id}=q_{fd}$ 。若结点  $K_i$  存在多个依赖强度值,则取最大的那个。

**规则 2** 若存在前提关系: $p(K_p, K_d, q_{pd})$ ,  $q_{pd}=1$ ,则对所有的  $K_i (i=1, \dots, n)$ ,  $s(K_i, K_p)$ ,且  $K_i$  与  $K_p$  的距离小于阈值  $\rho$ ,则有  $p(K_i, K_d, q_{id})$ ,  $q_{id}=q_{pd}$ 。

以上规则在提炼与知识点  $K_p$  关联度较高的知识点集合时,分为向上和向下两个方向的动作,则在设定一定的阈值  $\mu$  与  $\rho$  时,向上可以将与  $K_p$  一定距离内的祖先结点囊括进来(距离越远的结点,认为其关联度较小),向下可以将相关联的子孙结点包括进来。通过依赖度  $q$  的计算,可以得出这些结点的依赖强度。从这个角度来看,可以将其视为依赖度  $q$  的传播算法。通过设定不同的阈值,可以得到不同范围大小的知识子树,从而适应不同学习能力的用户。

**定义 4(知识关联图, Knowledge Association Graph)** 知识关联图为一个多元组: $KAG=(N_k, N_r, R_s, R_p, R_a)$ ,其中  $N_k$  为知识点的结点集,  $N_r$  为资源结点集,  $R_s$  为 subtopic-of 关系集(边),  $R_p$  为前提关系 precedence-of 的关系集,  $R_a$  为 association-of 关系集。

图 1 是一个数据结构学习系统的 KAG 图的一部分。图中, subtopic-of 关系用实线表示, precedence-of 关系用虚线。为简便起见,忽略了 association-of 关系。图中省略了名称的知识点为: K6111—K6112 为二叉树的顺序和链式存储结构; K6121—K6123 为二叉树的先序、中序和后序遍历; K6131—K6133 为线索二叉树的存储结构、遍历和中序线索化; K6211—K6213 为树的双亲、孩子和孩子兄弟表示法。

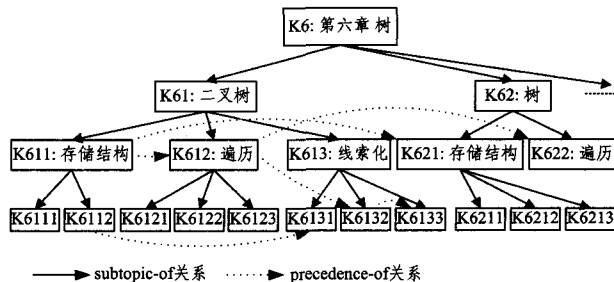


图 1 KAG 示意图

在获得了知识关联图之后,用户可以通过生成的学习序列对其进行学习。在学习过程中,可能需要对其中某个知识点  $K$  及其关联的知识进行学习,那么可以将此紧密关联的集合提取出来,生成一个 KAG 的子图,以满足要求。

**定义 5(学习图, Learning Graph)** 学习图  $LG(K) = (N, R)$  是关于知识点  $K$  的一个定义良好的 KAG 子图, 且满足以下条件: 1)  $N \subseteq N_k \cup N_r, R \subseteq R_i \cup R_p \cup R_a$ ; 2) 对  $LG(K)$  的任意两个不同的知识结点  $K_p, K_d \in N$ , 若有  $p(K_p, K_d, 1)$ , 那么对于  $K_p$  来说, 所有满足规则(1)和规则(2)的结点  $K_i$ , 则必有  $K_i \in N$ ; 3) 对任意一个结点  $K_d \in N_k$ , 若  $r$  为与  $K_d$  相关联的资源, 则必有  $r \in N$ 。

## 2.2 学习序列的生成算法

在有前提关系约束的条件下生成合适的学习序列, 是实现学习系统的一个关键。学习内容的组织依照其内在的逻辑联系通常采用树型结构, 因此采用深度优先遍历比较合适。而对于加入知识点之间的前提关系之后形成的 KAG 图, 通常采用的拓扑排序方法所生成的学习序列数目庞大, 效率低, 难以选择最优的序列, 且序列的知识点之间缺乏逻辑联系。因此, 本文对此进行改进, 将深度优先遍历得到的次序关系作为启发信息, 提出一种深度优先遍历指导下的拓扑排序算法以用于学习序列的生成。

**定义 6(优先关系,  $\leq$ )** 设  $T_i(K)$  与  $T_j(K)$  为两棵知识树, 若  $\exists x \in T_i(K), y \in T_j(K)$ , 且  $p(x, y)$  或  $s(y, x)$  为真, 则有次序优先关系  $T_i(K) \leq T_j(K)$ , 表示知识树  $T_i(K)$  要先于  $T_j(K)$  进行遍历。

算法名称: 学习序列生成算法 KSequenceGen()

入口参数:  $LG(K)$  图

返回: 学习序列 RList

0)  $RList \leftarrow \Phi; LG(K)' \leftarrow LG(K)$ 。

1) 将  $LG(K)'$  图删除 precedence-of 关系, 删除 association-of 关系和资源结点  $N_r$ , 得到知识树森林  $F: (T_1(K), T_2(K), T_3(K), \dots, T_n(K))$ ; 若  $n=1$ , 则采用深度优先遍历知识树  $T(K)$ , 得到一个知识点的序列 KList, 转步骤 3); 否则转下一步。

2) 根据定义 6, 确定知识树之间的次序优先关系, 得到新的知识树序列:  $T_1'(K) \leq T_2'(K) \leq T_3'(K) \leq \dots \leq T_n'(K)$ 。依次对此序列的知识树进行深度优先遍历, 并将得到的知识点序列合并成一个总的知识点序列 KList。

3) 判断 KList 是否为空。若为空, 则转步骤 10); 否则转下一步。

4)  $e \leftarrow$  KList 的队头元素。

5) 判断  $e$  在图  $LG(K)$  中对应结点的入度是否为 0? 若为 0, 则转下一步; 否则转步骤 9)。

6)  $RList.enQueue(e)$ 。

7) 将图  $LG(K)$  中所有  $e$  发出的弧所指的结点入度减 1。

8) 在 KList 中删除  $e$  对应的元素。

9) 在 KList 中是否还有下一个元素? 若有, 将该元素赋给  $e$ , 转步骤 5); 否则转步骤 3)。

10) 返回结果 RList, 算法结束。

该算法可以对学习图  $LG(K)$  获得一个初步的学习序列, 序列中的知识点之间满足前提关系的约束。下面讨论如何针对用户的不同学习水平和个性化特点, 对生成的学习序列进行改进和优化, 以赋予学习系统的自适应性和个性化特点。

## 3 个性化的学习序列生成

学习活动通常是按照一定大小的时间单元来组织的, 例如一节课、一个学时等。学习一门课程的全体知识大致可以用时间单元的数目来进行衡量(如 56 个学时), 其中默认为每个学生的学习效率、掌握知识的程度是一样的。而实际上, 由

于学生的学习能力、学习特点等因素的不同, 其所花费的时间单元数有较大差异。而这个差异的大小, 也反过来反映了学生的学习能力的不同。因此, 按照学生学习能力的不同, 依据一定的时间单元大小, 将学习内容划分为一个由更大尺度的学习单元组成的序列, 有助于提供满足用户个性化学习需要的服务。

学习单元内所包含的知识点需要用户花费一定的精力去学习, 其花费的代价我们称为学习工作量。它与学习单元所包含的知识点数量和难度有关。在学习单元的时间大小一定的条件下, 显然, 包含的知识点数量越多, 知识点的难度越大, 则对应的学习工作量就越大。

**定义 7(学习能力)** 适合用户的学习能力可定义为学习单元内所包含的学习工作量的大小与学习单元的时间大小的比值, 记为  $l_a = eff/t$ , 其中,  $eff$  为学习工作量,  $t$  为学习单元的时间大小。显然, 在确定了学习单元的时间大小和课程的总学习工作量之后, 根据每位学生的学习能力, 可以得出该学生大致的学习时间, 进而生成适合其学习特点的学习序列, 我们将其称为学习窗口序列。

**定义 8(学习窗口)** 一个学习窗口为一个多元组:  $LW = (t, KS, eff)$ , 其中  $t$  为学习窗口的时间大小, 一般是一节课或一个学时;  $KS$  为学习窗口所包含的知识点集合,  $KS \subseteq N_k$ ;  $eff = \sum_{i=1}^N dif_i$  为学习窗口对应的学习工作量, 为简便起见, 将其定义为知识点的难度系数之和, 其中,  $N = |KS|$ ,  $dif$  为知识点的难度系数。在时间大小  $t$  一定的情况下, 我们称学习工作量  $eff$  为学习窗口的容量。

图 2 为一个由学习窗口组成的学习序列, 其中,  $LW_i$  为当前学习窗口,  $LW_i$  前面的窗口序列为已经学习过的, 后面的为还未学习过的。注意, 每个学习窗口的时间大小  $t$  一般相等, 但是所包含的知识点个数  $n_i$  可能不同。显然, 在学习工作量一定的情况下, 知识点的难度系数越大, 则包含的知识点就越少, 反之, 则相反。因此, 这些数值有助于系统精确地控制对应学习网页的生成。

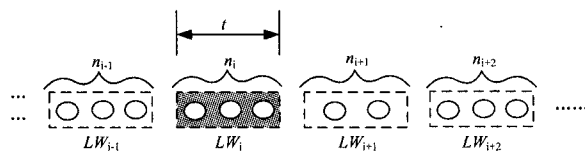


图 2 由学习窗口组成的学习序列示意图

### 3.1 用户学习水平的计算

用户的学习能力只能反映其学习的效率, 而不能反映其对知识的掌握情况。我们将用户的学习能力与用户对知识的掌握结合起来, 用于描述其学习水平。用户的学习水平可以由用户以往的学习和测试情况体现出来。测试反映了用户对学习内容的掌握, 知识点越多, 对应的测试成绩越好, 则学习水平越高。同样地, 学习水平也与用户的学习效率有关, 对于相同分量的学习内容, 学习的时间越短, 则该用户的学习水平越高。

设对于知识点序列  $K_1, K_2, \dots, K_P$ , 用户学习了  $N$  次, 每次所花的时间依次为  $t_1, t_2, \dots, t_N$ ; 用户考试的次数为  $M$ , 每次考试的成绩为  $s_i$ , 难度系数为  $q_i, i=1, \dots, M$ , 则用户的学习水平  $l_i$  可用以下公式计算:

$$l_i = \alpha \sum_{j=1}^M s_{ij} \varphi_{ij} / M + \beta \sum_{j=1}^P \lambda_{ij} / \sum_{j=1}^N t_{ij} \quad (1)$$

式中,  $\lambda$  为知识点的难度系数,  $\alpha, \beta$  为权值, 且  $\alpha + \beta = 1$ 。每次测试的难度系数  $\varphi_i$  由其中试题的难度  $\varphi_{ij}$  决定,  $\varphi_i = \sum_{j=1}^T \varphi_{ij} / T$ ,  $T$  为测试中试题的总数。

我们将用户的学习水平  $l_i$  用于设定和调整学习窗口的容量大小, 以提供适应用户水平的学习内容。在实际应用中, 如果在用户的学习效率足够高的情况下, 发现其对知识的掌握情况不理想, 则可以调低其学习窗口的容量大小, 并给出提示; 反之, 则可以适当加大学习窗口的容量大小。

### 3.2 用户个性建模

在心理学里, 个性常用个性特质和倾向性来描述。个性特质指在心理活动过程中表现出来的比较稳定的成分, 而个性倾向是指人们对客观事物的基本趋向、基本态度和基本看法, 主要包括需要、动机、兴趣等<sup>[9]</sup>。作为个性中的动力结构, 个性倾向对个性的其他因素起支配作用, 并影响其他行为活动<sup>[10]</sup>。本文主要探讨个性在学习系统中的应用, 以及对学习模式的影响, 其中着重考虑了用户偏好的作用。

用户的个性是实现学习系统适应性和个性化的重要因素。例如, 给定多个学习选择项, 系统可以根据用户个性进行抉择。如有的学生喜欢通过练习来促进学习, 那么可以将学习环节的工作量调低, 将练习环节的比重调高; 有的学生喜欢学习完全部的知识点之后再进行测试, 也有的学生喜欢学习完一个或数个知识点之后就先进行测试, 然后再学习下一个知识点等。因此, 不同个性的用户对系统提供的学习内容的组织和呈现、学习和测试的组合方式、学习资源的选择等有不同的要求, 尽量精确地刻画和捕获用户的个性, 对于实现个性化的智能学习具有重要意义。然而在实际应用中, 要求用户显式地输入其个性特点会带来许多不便, 也影响用户体验的提高。因此, 更好的方法是通过对用户的学习和操作记录进行学习, 捕获用户的学习个性, 来逐步提高系统对用户的适应性。

**定义 9(学习个性)** 用户的学习个性可以定义为一个多元组:  $Per = (P, M, F)$ , 其中,  $P = \{P_1, P_2, P_3, \dots, P_n\}$  是用户的偏好集, 每项表示用户对某一方面事物的喜好或厌恶。用户偏好项  $P_i = (name, d\_time, w)$ , 其中  $name$  为偏好项的名称;  $d\_time$  为偏好项的统计时间, 表示为了得到该偏好项, 所计算的多长时间范围的数据;  $w$  是偏好项的强度值, 称为兴趣度。兴趣度值越大, 则用户的兴趣越大; 若兴趣度为负值, 则表示该用户厌恶此方面的事物。  $M$  是学习模式集, 表示学习内容和学习方式的不同模式, 例如知识点与辅助资源的搭配方式、知识学习与测试的组织方式等。  $F$  为映射函数,  $F: P \rightarrow M$ , 将某偏好项映射到对应的学习模式。

**定义 10(学习模式)** 学习模式是学习序列的具体实现, 表示知识内容的具体组织方式及对应的学习方式, 可以用三元组定义:  $M = (O, S, L)$ , 其中,  $O$  为学习内容集,  $O \subset N_k \cup N_r$ , 包括了知识点和附属的资源;  $S = \{s_i \mid s_i = \langle o_i, o_{i+1} \rangle, o_i, o_{i+1} \in O, i = 1, \dots, n-1\}$  为学习内容之间的次序集;  $L$  为学习方式集,  $L = \{l_1, l_2, l_3, \dots, l_n\}$ , 表示不同的学习方式。例如, 先将知识点全部学习完再进行测试, 或者在学习了知识点后及时进行测试, 就是两个典型的学习方式。系统可以根据不

同的学习模式生成对应的学习网页序列。

图 3 是根据用户的学习偏好, 将初始的杂乱的学习序列转换为适合用户个性的学习模式的示意图, 考察的是学习与测试的组织方式。图中的圆圈表示知识内容的学习, 其中蕴含了知识点与辅助资源的不同搭配形式。方块表示对该知识点的测试与练习。不同的学习模式代表了知识学习及其对应测试的不同组织方式。例如, 图中的模式(1)表示在每个知识点学习完之后马上进行测试, 而模式( $n$ )表示用户希望将所有的知识点都学习完之后才进行测试。根据需要, 也可对知识点与辅助资源的组织形式进行考察, 则学习模式体现的是用户对这两者之间的搭配方式的偏好。优化后的学习模式将满足用户的个性化需求。

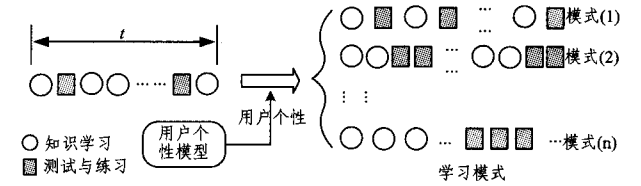


图3 个性化的学习模式的生成

值得注意的是, 偏好集可以根据需要进行扩充或删减。偏好集的调整可以赋予系统对学习方式、学习内容组织等不同方面的适应能力, 只需要设计好合适的偏好项与对应学习模式的关联即可; 而偏好项中兴趣度的计算可以帮助系统精确刻画用户的兴趣强度, 从而针对不同兴趣度的学生提供合适的学习模式。兴趣度的计算主要是通过对以往的学习记录等信息进行获取, 但也可通过问答的方式由用户显式地输入。下面讨论对偏好项“学习与测试的取舍倾向”的兴趣度的计算。

设一段时间  $t$  内用户对某知识内容  $O$  进行了学习, 在学习过程中还包括对知识点的内容进行测试。用户可以选择在学习完某知识点之后马上进行测试, 也可以选择将测试进行推迟。同时一个测试可以只包含一个知识点, 也可以包含多个知识点。由该时间段内的学习记录可以得到对应的兴趣度, 用式(2)进行计算:

$$w_{(t)} = \frac{N}{\sum_{i=1}^N \sum_{j=1}^{c(E_i)} d(K_{ij}, E_i)} / N \quad (2)$$

式中,  $N$  为测试的次数;  $E_i$  为第  $i$  个测试, 其中包含的知识点个数为  $c(E_i)$ ; 函数  $d(K_{ij}, E_i)$  用于计算知识点  $K_{ij}$  和测试  $E_i$  之间的距离。如果是对当前窗口进行计算, 则得到的兴趣度称为短期兴趣。由短期兴趣可以得到长期兴趣, 用式(3)进行计算, 其值随时间衰减。

$$w_{(per)} = \frac{1}{N} \sum_{i=1}^N w_{t_i} e^{-\ln(t_c - t)/k} \quad (3)$$

式中,  $N$  为统计的窗口数,  $k$  为半衰期,  $t$  为  $w_{t_i}$  计算的时间,  $t_c$  为当前窗口所处的时间。随着用户学习过程的进行, 系统将计算得到的短期兴趣不断地更新长期兴趣, 以更贴近用户的偏好。显然, 在获得了长期兴趣之后, 可以将其用于对当前学习窗口以及后续的学习窗口的调整, 例如, 调整学习窗口的学习模式等, 以适应用户的个性特点。

### 3.3 多因素集成的个性化学习序列生成

在综合了前提关系的约束、用户学习水平和个性化建模及计算的步骤后, 可以得到更贴近用户学习个性、具有适应性

的学习窗口序列。那么在将学习窗口序列转换为学习网页序列之后,将其提交给用户进行学习。同时随着学习的不断进行,系统通过对历史学习记录的学习,捕获用户更确切的学习水平和个性特点,以提供更合适的学习服务。

综合了以上因素的学习序列生成算法的主要步骤如下:

0) 系统初始化,如设定学习窗口的容量和时间大小  $t$ ,设定依赖强度阈值等;

1) 建立课程知识的知识关联图 KAG;

2) 调用  $KSequenceGen(KAG)$ ,以生成满足前提条件约束的学习序列  $KList$ ;

3) 用式(1)计算用户的学习水平,得到  $l_s$ ;计算  $l_a$ ;

4) 根据学习水平  $l_s$  和  $l_a$  的大小,修改学习窗口的容量;将学习序列  $KList$  划分为由学习窗口组成的学习序列  $WList$ ;

5) 根据式(2)计算得到短期兴趣;根据式(3)将短期兴趣更新为长期兴趣,得到优化的长期兴趣;

6) 将学习序列  $WList$  加入对应的资源;根据得到的短期或长期兴趣,调整学习序列  $WList$ ,包括对当前学习窗口和后续学习窗口进行优化,修改和确定合适的学习模式等,得到个性化后的学习序列  $PList$ ;

7) 返回  $PList$ ,以用于生成网页序列并提交给用户。

## 4 系统实现及实例分析

作为本文方法的实例化应用,参考了 AIES 系统的基本结构,我们在一台高性能 PC 机上实现以上算法和原型系统开发,系统的流程框架如图 4 所示。其中,领域知识库主要用于存储用户需要学习的领域知识,包括知识关联图和各种学习资源等。学生知识库用于存放与学生相关的知识,包括学生的学习能力、学习水平、个性偏好、以往的学习记录(学习的时间、对应的知识点等)和考试记录等。

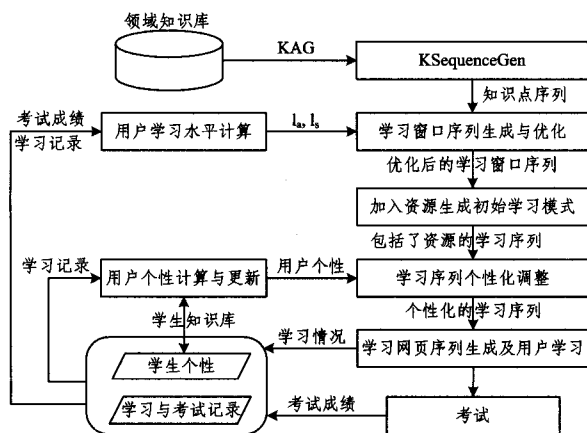


图4 原型系统框架图

开发的硬件环境: Intel E5200 CPU, 2G DDR, 250G HD, 开发平台: JDK1.6 + MyEclipse8.6, 数据库采用 SQL Server 2005。采用 Javabeen 进行服务器端各模块的开发,用 JSP 控制用户界面的动态生成。采用 Applet 实现客户端的图形显示及与服务器端的数据交换。

下面,以图1所示的知识关联图为例,来验证本文所提方法的有效性和正确性。对该知识关联图运行  $KSequenceGen$  算法后获得的学习序列为: K6, K61, K611, K6111, K6112, K612, K6121, K6122, K6123, K613, K6131, K6132, K6133,

K62, K621, K6211, K6212, K6213, K622, 满足知识间的逻辑联系和前提条件的约束。

在预设了上述学习序列的知识点难度系数之后,计算得到其总的学习工作量为 3.6,其中,知识点 K61(包含二叉树的性质,难度系数 0.2)、K6121—K6123(二叉树的先序、中序和后序遍历,难度系数 0.2)、K6133(二叉树的中序线索化,难度系数 0.5)等知识点难度系数较大。在对学生的学习能力进行统计之后,获得其平均值约为 0.6,因此,可以将以上学习序列大致划分为 6 个学习窗口:  $LW_1$  (K6, K61, K611, K6111, K6112),  $LW_2$  (K612, K6121, K6122),  $LW_3$  (K6123, K613, K6131),  $LW_4$  (K6132, K6133),  $LW_5$  (K62, K621, K6211, K6212),  $LW_6$  (K6213, K622), 学习窗口的时间大小为 1 学时,窗口的容量约为 0.6。在某学习过程中,学生 A 选用模式(1)对此窗口序列进行学习,在学习了  $LW_1$  和  $LW_2$  后,系统获得该学生的学习能力  $l_a=1.2$ ,则系统自动将剩余的学习窗口调整为 2 个,即  $LW_3'=LW_3 \cup LW_4$  和  $LW_4'=LW_5 \cup LW_6$ ,窗口时间大小不变,容量变为 1.2。学生 A 继续学习了  $LW_3'$ ,系统通过对其测试情况的考察,进而由式(1)计算得到该用户的学习水平  $l_s$ 。在观察到其学习水平  $l_s$  低于用户平均值后,表明该学生对知识的掌握情况并不好(因为  $l_a$  值较大),则调整学习窗口的大小,将窗口容量减少为 0.8,那么  $LW_4'$  会被拆分成两个学习窗口,同时对学生 A 给出合适的提示信息。

下面讨论兴趣度的计算和学习模式的选择。系统在每个知识点对应的学习网页上都提供相应的测试链接,然后观察用户在学习过程中的选择。注意用户在点击某测试链接时,系统会自动地将该知识点以前的序列中没有进行的测试合并到该测试中,以处理用户学习知识点时推迟测试的情况。假定用户 C 在学习完每一个知识点之后,立即选择进行相应的测试,则根据式(2)可以计算得到其兴趣度  $w_A=1$ ;用户 D 在学习完每 2 个知识点之后,选择进行相应的测试,可求得兴趣度  $w_B=2.8$ 。显然,用户选择的测试推迟的距离越大,则兴趣度  $w$  值越大,然后可建立兴趣度与学习模式之间的映射函数。在实际应用中,系统在观察了一定数量的用户行为之后,可计算得到用户的兴趣度,进而得到用户喜欢的学习模式,那么在该用户后续的学习过程中,系统可以自动生成符合用户个性偏好的学习网页序列。

**结束语** 适合学生个性特点并能进行适应性改变的学习序列的生成,一直是实现远程教育系统智能性和适应性的关键。单纯地依靠教学计划来生成学习序列,或抛开知识点之间的关系,通过随机的学习序列的试探都难以实现。本文提出了采用知识点之间的内在关系作为启发信息,结合拓扑排序的学习序列生成算法,讨论了学生学习水平和学习个性的建模和计算,并以此为基础对学习序列进行调整和优化,给出了综合多个因素的个性化学习序列生成的算法。系统可通过与学生的交互,对当前和以往的交互记录进行学习,并改进学习序列以获得适合该学生学习特点的学习序列。采用学习窗口作为学习活动组织的单元,给出了时间成本和工作量描述的方法,采用学习模式描述知识内容的组织方式及对应学习方式的不同。

作为本文方法的实例化,已在本单位的数据结构学习平

台中投入应用,目前运行效果良好,表明了本文提出的方法是可行的和合理的,且具有较大的发展潜力。下一步的工作将对学习序列在自适应学习导航中的应用、个性化的学习模型等方面进行研究,并进一步提高原型系统的自适应能力。

## 参考文献

- [1] Iglesias A, Martínez P, Aler R, et al. Learning teaching strategies in an Adaptive and Intelligent Educational System through Reinforcement Learning[J]. *Applied Intelligence*, 2009, 31(1): 89-106
- [2] 许高攀,曾文华,黄翠兰.智能教学系统研究综述[J]. *计算机应用研究*, 2009, 26(11): 4019-4022
- [3] Cabada R Z, Barrón Estrada M L, Reyes García C A. EDUCA: A web 2.0 authoring tool for developing adaptive and intelligent tutoring systems using a Kohonen network[J]. *Expert Systems with Applications*, 2011, 38(8): 9522-9529
- [4] Arnau D, Arevalillo-Herráez M, Puig L, et al. Fundamentals of the design and the operation of an intelligent tutoring system for the learning of the arithmetical and algebraic way of solving word problems[J]. *Computers & Education*, 2013, 63: 119-130

(上接第 175 页)

SWT/JFACE 组件类型和数量相关。表 4 显示一个基本参数类型的生成时间在 5ms 左右。当前版本的 AUTOSAR 基础软件模块参数采用容器嵌套方式组织参数,一个当前配置界面中,最大不超过 20 项参数,最大生成时间不超过 100ms,非常快。因此本文提出的生成方法完全能满足汽车电子配置工具的需求。

文献[3]基本类型的 SWT 组件设计与本文设计略有不同,虽然大部分基本类型的平均总消耗时间与本文的测试结果在一个数量级上,但是 string 类型的消耗时间与本文的结果相差很大。例如:500 个 string 类型的参数界面生成需要消耗的时间在 200s 左右,而本文的消耗时间在 1.3s 左右。根据文献[3]表 2 的定义, string 类型参数采用与 Integer、Float 一样的 SWT 组件,但运行结果相差极大。

**结束语** 本文根据汽车电子基础软件模块配置的实际需求,提出一种动态生成界面方法,即基于 AUTOSAR ECU 参数定义模型文件,根据当前配置界面对应的参数列表及参数类型,动态生成配置界面。该方法分离了界面与参数,使得配置工具的可扩展性和可维护性很高。可行性实验选取参数定义类型有代表性的 OS 模块生成界面,其生成的界面美观、友好。性能测试实验显示生成一个基本参数类型非常快,完全满足汽车电子配置工具的需求。基于上述方法开发的配置工具不但可以支持 AUTOSAR 现有几个版本的模型文件,还可支持 AUTOSAR 标准升级版本的模型文件,而无需对配置工具作修改。

## 参考文献

- [1] AUTOSAR Partnership. AUTOSAR Methodology V1. 2. 2 R3. 1 [S]. <http://www.autosar.org/>, 2009
- [2] AUTOSAR Partnership. AUTOSAR ECU Configuration Pa-

- [5] Steichen B, Ashman H, Wade V. A comparative survey of Personalised Information Retrieval and Adaptive Hypermedia techniques [J]. *Information Processing & Management*, 2012, 48(4): 698-724
- [6] Andre E, Finkler W, Graf W, et al. Intelligent multimedia presentations[M]// Maybury M T, ed. *The automatic synthesis of multimodal presentations*. Cambridge: MIT Press, 1993
- [7] Sleeman A. A system which allows student to explore algorithms[C]// *Proceedings of international joint conference on artificial intelligence*. 1977: 780-786
- [8] Murray R C, Vanlehn K, Mostow J. A decision-theoretic approach for selecting tutorial discourse actions[C]// *Proc of the NAACL workshop on adaptation in dialogue systems*. Pittsburgh, PA, USA, 2001: 41-48
- [9] 刘斌. 智能体的一种个性模型[J]. *计算机科学*, 2008(12): 203-206
- [10] Schiefele U. The influence of topic interest, prior knowledge, and cognitive capabilities on text comprehension[M]// Pieters J M, Breuer K, Simons P R J, eds. *Learning Environments*. Springer-verlag Berlin Heidelberg, 1990: 323-338

rameter Definition V2. 2. 0 R3. 1 [S]. <http://www.autosar.org/>, 2009

- [3] Li Nan, Li Hong, Zhong Xiao-feng, et al. AUTOSAR Based Automatic GUI Generation[C]// *Proceedings of the 13th IEEE International Symposium on Object/component/service-oriented Real-time distributed computing*. Carmona, Spain, 2010: 156-162
- [4] Voget S. AUTOSAR and the Automotive Tool Chain[C]// *Proceedings of the Conference on Design, Automation and Test in Europe*. 2010: 259-262
- [5] Puerta A, Eisenstein J. XIML: A Common Representation for Interaction Data [C] // *Proceedings of IUI2002*. ACM Press, 2002: 214-215
- [6] Schneider K A, Cordy J R. AUI: A Programming Language for Developing Plastic Interactive Software[C]// *Proceedings of the 35th Annual Hawaii International Conference on System Sciences*. IEEE Press, 2002: 281-290
- [7] Baron M, Girard P. SUIDT: A task model based GUI-builder [C]// *Proceedings of TAMODIA 2002*. Printing House, 2002: 64-71
- [8] Jelinek J, Slaik P. GUI Generation from Annotated Source Code [C]// *Proceedings of TAMODIA 2004*. Prague, Czech Republic, 2004: 129-136
- [9] Monteiro M, Oliveira P, Goncalves R. A source code based model to generate GUI: GUI generation based on source code with declarative language extensions[C]// *Proceedings of the 3rd International Conference on Software and Data Technologies, INSTICC Press*. 2008: 21-28
- [10] Monteiro M, Oliveira P, Goncalves R. A proposal to delegate GUI implementation using a source code based model[C]// *Proceedings of the 19th International Conference on Software Engineering and Knowledge Engineering*. Boston, MA, Knowledge Systems Institute Graduate School, 2007: 29-32
- [11] 张宁, 刘正捷. 自助服务终端界面交互设计研究[J]. *计算机科学*, 2012, 39(6): 16-20