

论用户模拟问题 7118

林 珊 (哈尔滨电工学院)

姜新法 (龙信公司)

摘 要

用户模拟问题在人工智能领域已成为广为瞩目的问题, 本文对该问题进行了系统而全面的讨论, 详尽地介绍了目前解决该问题的各种方法与技巧。

一、引言

用户模拟 (user modeling) 问题又称“学生”模拟问题, 它主要被用于对人工智能 (AI) 系统或软件系统进行调整, 使这些系统更适合于特定用户的特定需求。它是 AI 系统用户友善化的关键问题之一。尽管各种 AI 系统对拥有用户模型的需求越来越强, 但对这个问题开展得最早、进行得最深入的是 AI 中的智能教学系统 (Intelligent Tutoring System, ITS)。用户模拟在 ITS 中是不可缺少的四大组成部分之一^[1]。因在 ITS 中, 用户就是相对于系统的学生, 所以在本文中用户和学生是可互换的同义词, 这也是学生模拟这一术语的起源。本文要论及的大多数系统为 ITS。

早期研究中对用户的模拟主要采用概率论的方法。在收集一定统计数字的基础上, 用一平均值来表示任一用户。该方法的致命缺陷是: 以一般化代替特殊化。该方法认定任一用户都具有样板化用户所具有的特性。

随着人工智能的迅速发展, 用户模拟已与早期全然不同。它由表示用户当前知识状态的用户模型 (user model) 及推理出该模型的“诊断”模块组成。这种“诊断”是从用户可观测到的行为中推理出用户的知识状态—用户掌握或不掌握什么知识及掌握的程度。

模型与诊断互为依存, 用户模型是数据结构, 诊断则是对该数据结构进行各种操作

的模块, 这两者构成用户模拟问题。

二、用户模拟的应用

作为诊断模块结果的用户模型是一个反应用户知识状态的数据库, 它一般由 ITS 中其它模块来访问以改善系统性能, 最常见的应用为:

- 控制教学进程 根据用户模型来决定是否继续或结束某阶段教学。
- 调整教学方法 具有多种教学方法的系统周期性地访问用户模型以决定是否更改正在使用的教学方法。
- 给出提示、建议 为了在最适当的时候对学生进行干预 (提示), ITS 需访问学生模型。
- 出题 ITS 的出题系统若为动态生成, 需访问学生模型, 使新题或为学生刚错题的同类题, 或为略高于学生当前知识状态的题目。
- 解释 ITS 均具有解释能力, 为使对问题的解释更适合特定学生, 需访问用户模型。

三、用户模拟问题的分类

解决用户模拟的方法多种多样, 因此从不同角度有不同的分类。John^[2]根据算法是否唯一将用户模拟分为三类: 使用唯一算法类、多算法类及无固定算法类。本文遵循 Vanlehn^[3]的分类方法。根据下述三方面进行分类: (1) 输入带宽, (2) 知识类型, (3)

用户与专家差别的表示方法。

3.1 输入带宽

诊断模块的输入是系统与用户交互中所得到的各种信息(如用户回答了什么),根据这些信息,诊断模块要推出学生的知识状态。显然,信息越少(即输入带宽越窄)诊断越难。输入带宽一般为三级:思维近似状态,中间状态,和最终状态。

如果求解问题(如解 $76-18=?$ 等)的时间长于几毫秒,则可说用户在求解过程中经历了数个思维状态(mental states),最宽的一级本应当是这些思维状态,但实际上不可能直接接收到它们,故输入带宽的最高级为一系列思维近似状态。

在诸如求解代数式等过程中,可观测到学生的一系列求解步骤,即从问题的初态直至求解终态的一系列中间状态,输入带宽的第二级便是这种中间状态。

有的系统所能得到的关于用户的信息仅为用户给出的最终答案,这便是输入带宽的第三级:最终状态。

由于输入带宽将决定诊断所用算法,所以它比其它两方面更重要。在程序设计教学系统中可找出输入带宽这三级的典型例子。

Anderson的Lisp教学系统^[4]中具有一个对学生程序设计技巧认知过程记载详细的模型。在人机接口上,系统用菜单方式为学生提供多种选择:选下一求解目标,选一种解题策略,选用哪一代码段等。系统力图给出足够多的选择项从而能囊括学生能想到的任一求解方法,即囊括学生的任一条求解路径(对的和错的)。Anderson认为这种菜单方式不会干涉思维状态所经路径,它仅用于跟踪学生的认知过程从而获取一系列详尽的思维状态。该系统的输入带宽是典型的最高一级。

在“Spade ITS”中^[5],系统只允许语法正确的程序进入。系统可观测到学生编程的每一步,但无法直接得知学生在编程中的目的及策略。该系统的输入带宽基本上属于第

二级——中间状态。

“PROUST”^[6]是一Pascal教学系统,它所接收到的仅仅是一个编制好的Pascal程序。至于学生在编程中修改了什么及目的、策略,系统一概无法直接获得,这显然是输入带宽的第三级——最终状态。

3.2 知识类型

由于学生模型中所含的是学生所具有的知识,所以学生模型本身亦可对问题求解,预测出学生的答案(这是ITS不同于CAI的一大特点),显然还需要有一解释模块来应用学生模型中的知识并进行求解。学生模型中知识的类型左右着解释模块的复杂程度。

对过程性知识的解释较容易,一般不需进行搜索而仅需根据局部知识和问题的当前状态作出决策;对说明性知识的解释则复杂得多,常需要搜索整个学生模型才能最后得到答案。

实际上,知识类型还左右着诊断模块的复杂程度,因为诊断实际上是解释的逆过程。解释模块在给定知识库和问题的前提下求得解,而诊断模块是在给定问题和问题的解的前提下推出所用知识库。当解释说明性知识时,必须要遍历知识库;当要诊断出所使用的说明性知识时,学生的答案(对的或错的)可能来源于能产生该答案的任何知识中的一个。因此,诊断的复杂程度直接正比于解释模块的复杂度或者说受控于知识类型。

过程性知识又可进一步分为两类:单层及多层结构。前者无子目标而后者有。由于子目标与目标间某种非透明性(在目标分解成子目标时,子目标的动作状态及结果均为当前目标所不可见),多层结构型知识的诊断要更困难一些。

3.3 用户与专家间区别的表示

ITS中一般都具有两个模型:用户模型与专家模型(专家是指完全掌握该领域知识者)。专家模型的用途有多种,它可提供问题求解的正确方法,也可给出对该方法的解

释。用户将逐渐从他的初始知识状态向完全掌握该领域知识状态转化。专家模型实际上是一专家知识库，用户模型则是与之有差别的另一知识库（其中可能含错的或不完整的知识）。

但是基于效益及实现上的考虑，这两个知识库常合二而一，即常把用户模型用专家模型加上一个二者的差别集来表示。二模型的差别主要有两种：短缺的知识（专家具有但用户无）、错误的知识（用户具有而专家无）。

一类称之为覆盖型 (overlay) 的用户模型，只能表示出用户短缺的知识，故用户模型实质上只是专家模型的一个子集，仅覆盖专家模型的一部分。很多这类学生模型在专家知识库中用每项后附有的“权”来表示学生掌握该项知识的程度（如+1表掌握，-1表不懂），这类用户模型实现简便，故应用很广。

但覆盖方式的根本性缺陷是：它认定用户的错误仅来源于缺少或不充分掌握某些知识，这显然是一种不现实的简化。实际上用户的知识不仅仅是专家知识的子集而且常有对专家知识程度不同的曲解及错误概念，这些是专家模型中所没有的。

另一类用户模型则可同时表示短缺的及错误的知识，这一般由一个错误文库 (bug library) 及专家模型来实现，故称错误文库法。这是表示用户与专家区别的另一大方法。系统对用户进行诊断时，首先在错误文库中查找出相应错误并将其加入用户模型。系统力图使这样生成的用户模型恰好模拟用户的所作所为。

该方法以一很强的假设为基础，它认定用户能反复“正确”地运用错误概念，即：学生总能原封不动地应用曲解了的知识，就象计算机执行错误的程序一样。

生成错误文库是该方法的最大难点，找出用户可能犯的错误的方法不多且很耗时。另外，该文库很难具有完备性，用户的错误

难免在文库中找不到。补救方法一般是将文库中某些项组合以生成新项。

另一种错误文库法仅预设“错误零件”库，在诊断过程中用“错误零件” (bug parts) 动态地合成bug。例如ACM系统^[7]动态组合出的“错误”是一产生式，该产生式的左、右二部各来自两个错误零件库：“条件库”及“动作库”，若条件库有P项，动作库有A项，则系统可合成 $A \cdot 2^P$ 个错误。

尽管预生成错误零件库与预生成错误文库同样困难，但前者的规模小于后者，故实现较易。

总之，覆盖法，错误文库及错误零件库是目前表示用户与专家间区别的三大主要方法。如较著名的“差异法” (differential modeling) ^[8] 虽有其独特的诊断技术，但仍属于覆盖法。

四、诊断技术及典型的用户模型

4.1 要点跟踪法

该方法从整个问题求解中分析出一组解题时所用的主要知识即所谓要点 (issue)，它既不关心要点间如何相互作用，也不关心要点在求解时的作用，它只要求知道这组要点是否在解题中被用户应用了。显然，这是一种粗糙模拟法，下面以首推此方法的West^[8]系统为例来说明。

West是一棋类游戏教学系统，双方以首先走完70个方格的路径者为优胜。每一棋步都要选定一个由三个给定随机数及+、-、*、/等组成的表达式，该表达式的值即为应前进的步数。该游戏还具有将对方向出发点反弹、走捷径等机关。对每一棋步，West分别分析出用户与专家的所有要点，即走该棋步所需的主要知识。数学知识为如何适时使用+、-、*、/等，游戏知识为如何反弹对方、走捷径等。如用户此次棋步表达式为 $5 \cdot 7 + 2$ ，则West分析出的要点为“*”和“+”。

在分别分析出用户和专家棋步中要点的基础上，系统比较二者的要点并为每一要点设置两个计数器“took”和“missed”。只要用

户棋步中有某要点(不论专家棋步中有否), 则该要点的“took”计数器加“1”; 若专家的棋步中有某要点但用户无, 则该要点的missed计数器加“1”。系统粗略认定: 若 missed 计数器值高于took, 则用户没掌握该要点; 反之则说明掌握了该要点; 若二计数器均为“0”则应用该要点的时机尚未出现。

然而, 用户未用专家某一棋步的原因可能仅是没掌握诸要点之一(而非全部), 上述方法可能在该棋步含的所有要点的missed计数器上加了“1”, West对此所采取的弥补措施是: 仅当missed/took之比较大时才认定用户不掌握该要点。

4.2 用户模型的产生式表示

GUIDON^[9]系统用产生式系统表示用户, (该用户模型也是一个粗化的模型), 专家模型亦采用产生式如:

rule157 IF(1)患者有症状A and (2)有症状B THEN (a) 病灶在E处 (b) 器官E状态为W (c) 患者有疾病M

由于医疗领域知识繁杂, 模拟用户中的诊断也较困难。如用户对于吻合“rule157”的病例仅给出结论中的(b)而无(a), (c), 则实际上很难断言该用户是否使用了rule157, 因为该用户很可能根据患者其它症状而使用其它规则得出结论(b)。

为处理上述类似问题, GUIDON采用了一组诊断用产生式如:

IF (1) 用户结论中含rule(i)的结论 and
(2) 用户结论中不含rule(i)的全部结论

THEN 用户掌握rule(i)的程序降低70%
在GUIDON的覆盖式模型中, 每一知识项后附有一“权”, 该权值随着诊断模块运行类似上述的产生式而上下摆动。

4.3 模型跟踪

这是一细化的用户模拟方法, 它要求输入带宽为思维近似状态, 其解释模块为非确定性的。在问题求解的每一步, “非确定性解释模块选出一组当前可执行的产生式(确

定性的解释模块只选定出一个), 由诊断模块逐一执行后得到一组新状态, 若新状态之一正巧与用户的当前状态吻合, 系统便由此推论用户使用了该相应产生式。于是诊断模块在用户模型中记录下该信息。称此法为模型跟踪, 是因为诊断模块不断跟踪这一特定的问题求解模型—非确定解释模块, 并随后将其与用户状态比较。

若输入带宽不能为详尽的思维近似态系列, 或者说无法保证每二状态间仅施用一条产生式, 则不能采用此方法。

4.4 决策树

本方法及4.5节介绍的方法均为细化的用户模拟技术, 要输入均为输入带宽的最低级, 它们都采用了错误文库法并可预测一系列中间思维状态甚至思维近似态。

BUGGY系统^[10]采用了本方法。它设立了含55条错误的文库。因用户的错误常常不是孤立的单独出现, 系统还可处理55²个可能的错误“对”。

为了对如此众多的错误进行诊断, 系统预构成一棵用问题答案作索引的决策树。树根结点对应于第一个问题, 系统自动生成所有可能的答案(对的、错的、组合错误的)。这些答案都作为子女结点收入树内。有一种答案, 系统便链入树内一个以该答案为标志的结点。因为同一答案可能由几种不同的错误导致, 故该答案的结点内含多种诊断, 每一诊断都将作为下一步问题求解的基础, 即下一次仍用该诊断确定出的方法求解问题。对于第二个问题, 系统采用递归的方法并根据决策树的现状继续建造该树。

当系统工作结束时, 一棵硕大的决策树也就生成了。对用户的诊断就对应于从根至叶的一条条路径。系统花费最高的就是这棵决策树的建造, 至于在这棵树上对用户进行诊断则要简单得多, 它只不过是以用户答案制导从根找到叶而已。

BUGGY没对两个以上的错误组合进行处理。假如要处理4个错误的组(转封底)

(接第55页)

合, 系统将面对 $55^4 \approx 9 \times 10^9$ 种错误组合, 该数量级下的决策树已庞大得无法处理了。

4.5 交互式诊断

IDEBUGGY系统的诊断方法是交互式的, 它能动态生成有助于诊断的题目, 并可向用户发问以确定出适当的诊断。它与BUGGY显然不同, BUGGY预先定好测试题并预测出所有答案, IDEBUGGY则根据不同用户动态地生成题目。

IDEBUGGY每次都产生数个可能性的诊断, 为在其中选出最准确的, 系统力图生成一个对不同诊断能给出不同答案的测试题。若可生成, 当把用户给出的答案与预测的答案相对照时, 便可确认出最正确的诊断。

测试题生成算法为: 首先生成一问题, 然后用各候选诊断对所论问题求解(即使用由诊断所确定的知识对该问题求解), 若不

同诊断对该题给出同一答案, 则放弃该题并再次生成新题。上述过程直至不同诊断对某一问题给出不同答案为止, 这样选出的问题是系统最希望得到的。为处理无法找到理想题目的情况, 上述找题过程只占用一定的时间, 若在该段时间内找不到, 则在生成出的题目中选一较好者。

与题目固定的系统相比, 本方法可用较少的测试题更准确地找出学生学习中的错误。但该方法要求选题算法高效, 否则, 学生或用户等待时间过长。

五、结束语

用户模型实际上不仅应用于ITS。例如一个专家系统, 它一般应当具有向用户解释领域知识的能力, 该能力会因系统具有用户模型而大大增强。任一软件系统, 特别是智能性系统, 都会因有用户模型而大大改善接口友善性。

(参考文献略)

计算机科学

(1974年1月创刊)

第19卷第5期(双月刊)

1992年10月23日出版

国内统一刊号: CN51-1239

代号: 78-68

定价: 2.40元 国外定价: 5美元

编辑者: 中国科学技术情报研究所重庆分所

顾问: 《计算机科学》审编委员会

出版者: 中国科学技术情报研究所重庆分所

重庆市市中区胜利路132号

邮政编码: 630013

印刷者: 中国科学技术情报研究所重庆分所印刷厂

总发行处: 四川省重庆市邮政局

订购处: 全国各地邮政局