

自适应网络学习用户界面设计和实现

李 晓 王雪飞 邱玉辉

(西南师范大学计算机与信息科学学院 重庆400715)

Design and Implementation of an Adaptive User Interface for Web-Based Learning

LI Xiao WANG Xue-Fei QIU Yu-Hui

(Faculty of Computer and Information Science, Southwest China Normal University, Chongqing 400715)

Abstract An adaptive user interface helps to improve the quality of human-computer interaction. Most people at present join to Web-Based Learning by common browser. Due to the one-fits-all user interface, they have to face with the problem of lack of the support on personalized learning. The design and implementation of the adaptive user interface for Web-based learning in this paper is grounded in our work done before, for example interaction model, adaptive user models including domain models. The adaptivity is mainly expressed on learning contents and representation including layout as well as operation.

Keywords Human-computer interaction, User interface, Adaptivity, Web-based learning, MAS

1 引言

自适应用户界面促进了人机交互的质量。最近几年,自适应用户界面在基于 Web 的信息访问中其应用已变得日益重要。自适应用户界面通过构造和应用用户模型,能够动态适应当前用户和当前任务,为用户提供与手边任务和用户背景知识相关的信息,减少信息过载,辅助用户快速达到目标。

目前,加入到网络学习的大多数用户,通常经由标准浏览器(如 IE, NETSCAPE 等)加入到 Web 学习,因为“千人一面”的用户界面而面临这样一个共性问题:缺乏个性化如因材施教的支持。本文设计和实现的自适应网络学习界面在对学习用户的自适应方面做了一些有益的工作。

早期的自适应用户界面像 Let's Browse^[1]和 Letizia^[2]通过对文档关键字匹配来分析用户行为、预测用户的兴趣。而 CodeBroker 是一个辅助 Java 程序设计的智能 Agent^[3],能自动传送与当前用户任务相关的、而用户尚未知的类库,从而减轻用户的认知负担。文[4]中介绍的自适应用户界面通过探测和分析用户的事件序列来获取用户的行为模式,进而为用户提供及时的帮助。

本文研究的界面不同于上述方法。该自适应网络学习界面建立在我们对自适应人机交互的研究工作基础之上。该用户界面考虑了用户对人机交互的自适应性要求,通过自适应用户模型的构建,并结合 MAS 技术设计实现。其自适应主要体现在三个方面:一是学习内容的自适应;二是内容表达包括界面布局的自适应;三是用户操作的自适应。此外,为了保持用户学习过程中必要的注意力,我们在界面中还设计了一个浮动的信息提示框,以提供新的刺激或信息。

2 用户模型

在自适应用户过程中,对用户适应性的程度取决于用户模型,它是系统关于用户的知识。

2.1 自适应人机交互模型

在用户看来,人机交互应该具有这样的适应性:交互系统为用户提供的信息不仅仅需要考虑用户的认知因素,也应该考虑用户的非认知因素,并同时具有催化和激发机制,这样才能更好地适应当前用户和当前任务。

文[5]中给出的四元交互模型 GIRD,综合考虑了人机交互中“人”“机”的特征,并充分考虑用户的个体和反应特征。该模型主要由四个部分构成:目标、个体特征、反应特征和领域模型。用户目标的达成在于对用户适应其个体特征的资源,而该资源的组织和呈现方式取决于用户的被分解了的目标或任务,以及用户个体特征。用户对界面信息的反应特征被用来修正用户的资源模型构建和界面的调整。

该模型中的个体特征结合了用户的认知和非认知因素,以能充分反应用户的心理适应。认知因素方面主要是用户的背景知识、注意和工作记忆等特征,而非认知方面主要是用户的信息偏爱(包含用户的信息内容偏爱和信息表达偏爱)、情感、动机和个性。催化和激发用户的刺激主要由用户的动机或者个性所决定。反应特征主要是指系统提供信息(刺激)引起用户在行为上发生变化的反应类型以及采取的行为方式。领域模型则表达了用户在人机交互过程中所需要的资源。

系统在与用户的交互过程中,通过观察用户或者用户的操作,了解用户的反应特征,从而推导用户的个体特征。然后在此基础上,为用户组织适应性的资源,并在恰当的时间提供给用户。此外,系统将定期或不定期提供新的刺激,以能维持必要的学习注意力。

2.2 用户模型

自适应用户系统中,用户模型即是其在适应用户过程中,对所需要的个体用户(用户组)的信息和假设的收集^[6]。用户模型通过对用户的建模实现。

本文的自适应用户界面采用了一个自适应的用户模型。该模型在四元人机交互模型 GIRD 的基础上构建,其目的在

李 晓 副教授,博士研究生,主要研究方为网络应用、人机界面,人工智能。王雪飞 博士研究生,主要研究方向为多 Agent 的协商和协作,人工智能。邱玉辉 教授,博士生导师,主要研究方向为人工智能,电子商务、网络应用等。

于“理解”并在必要的时候激发用户。用户的学习行为模式通过 Web 可用性挖掘获得,学习的终极目标和信息偏爱可由用户自己决定。此外,构造了领域模型以表达用户的信息资源偏爱。该模型基本表达了用户的知识,满足自适应性的要求。

用户模型构建思路如下:首先构造用户组模型,并将之作为首次登录用户的缺省用户模型。然后在首次登录用户提供必要的信息后,通过相似度的计算,为之确定一个最适合的用户模型。之后在与该用户的不断交互中,实现对其的个体自适应用户模型构建。

该模型在界面程序实现中采用 XML 如下表示:

```
<UserModels>
  <Sys id="SystemHelpP" exist="False" ...../>
  <Fun id="..." show="True" ...../>
</Sys>
< Sys id = " SystemInterface" exist = " true" comment = "
  UserInterface" ...../>
  <Fun id="HeadOfInterface" show="true" comment="TopOf
  Page" ...../>
  <Fun id = " TailOfInterface " show = " true" comment = "
  BottomOf Page" ...../>
</Sys>
</ UserModels>
```

其中 exit, show 的取值(true 或 false)体现了用户的信息表达偏爱。模型中的信息内容偏爱通过系统自动构造用户的领域模型(参见下节)实现。

3 设计实现自适应学习界面

3.1 体系结构

图1给出了基于 MAS 实现的自适应网络学习用户界面体系结构。客户端用户经由界面 Agent 与系统进行交互;服务器端则使用了三个彼此合作的 Agent:监控 Agent、建模 Agent 和资源 Agent,为用户提供信息服务。

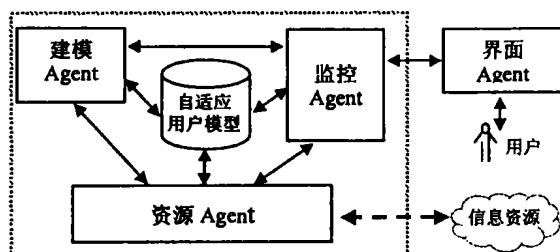


图1 自适应网络学习用户界面体系结构

界面 Agent 是唯一与环境交互的 Agent,它通过感知用户行为的变化,将必要的环境参数传递到监控 Agent。此外,还按照用户的信息表达偏爱实现信息资源的界面布局,并提供对用户操作适应性的支持。监控 Agent 一方面协调建模 Agent 和资源 Agent 的活动,另一方面响应界面 Agent 的请求,传递用户需要的信息资源;建模 Agent 的主要功能是动态、渐进构造并更新自适应用户模型;资源 Agent 的主要功能是根据个体认知和非认知特征,构造用户的领域模型。各 Agent 采用 Java 编程实现,而 Agent 间的通信则基于消息机制实现。

3.2 领域模型

领域模型即个体偏爱的信息内容结构的描述,该模型实际上表征了系统以用户信息内容偏爱为核心的、整合的信息资源。例如:在基于 Web 的学习中,该领域模型表现为适合用户特征的、与当前学习单元或知识点相关的学习资源组织。该部分的工作参见文[7~9]。图2是领域模型的结构,图3则是领

域模型的一个实例。

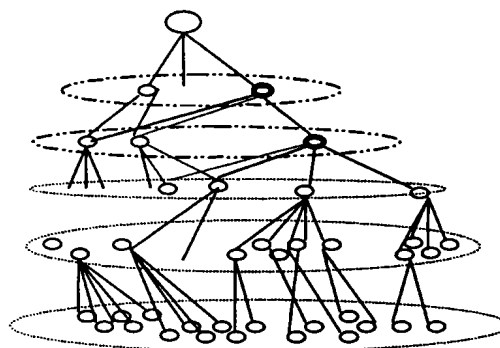


图2 领域模型

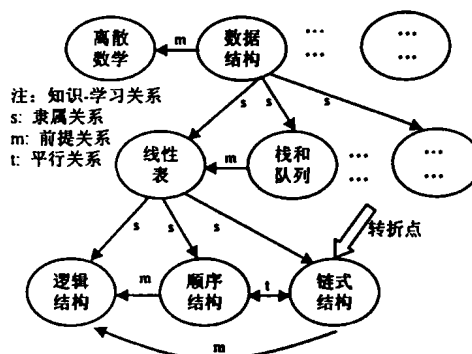


图3 领域模型实例

界面设计中,领域模型采用 XML 表示如下:

```
<Course id="..." courseName="课程1" author="..." subject="
  ..." ...../>
<Goals>
  <Goal id="..." goalName="目标1" upperScore="85"
  passScore="60" ...../>
  <Goal id="..." goalName="目标2" upperScore="85"
  passScore="60" ...../>
</Goals>
<Sections>
  <Section caption="第1章" id="..." parentIndex="0"
  parentSectionid="..." ...../>
  <Section caption="第2章" id="..." parentIndex="1"
  parentSectionid="..." ...../>
</Sections>
<KnowledgeNodes>
  <Knode id="..." caption="第1章内容" outline="1.....2....."
  teachRequest="1. 初步理... 2. 初步了解....." ...../>
  <Instruction id="..." caption="第1章" instype="0" ...../>
  <Content type="文本/视频..." id="..." texttype="1/2..."
  ...../>
  <Content type="文本/视频..." id="..." texttype="1/2..."
  ...../>
  <Content type="文本/视频..." id="..." texttype="1/2..."
  ...../>
  <Test id="..." caption="第1章测试" ...../>
  <Question ...../>
  <Answer ...../>
  <Choice ...../>
  <Content type="文本/视频..." id="..." texttype="1/2..."
  ...../>
  </Choice>
  </Answer>
  </Question>
</Test>
```

3.3 界面的自适应体现

目前,该自适应界面已基本实现。原型系统开发平台: Windows2000 Server, Tomcat v4.0, SQLServer2000。界面 Agent 基于 JSP 和 Servlet 技术实现,它能感知环境的变化如用户的选择、鼠标悬停等,然后将必要的环境参数如用户 ID

或者访问 ID 等通过 request 对象或 HttpServletRequest^[10]传递到服务器端的监控 Agent, 然后由其进行相应的处理。

界面的自适应主要体现在三个方面: 一是学习内容的自适应; 二是内容表达包括界面布局的自适应表示(见图4和图5); 三是用户操作的自适应。此外, 为了保持用户在学习过程中必要的注意力, 我们还特别设计了一个浮动的信息提示框, 以能在必要的时候提供用户新的刺激信息, 该信息可以是当前学习内容的学习技巧或专家知识, 也可以是一个小游戏(见图4椭圆形所示五子棋游戏)。图4和图5是具有不同学习背景和信息偏爱的用户1(ID=rwl)和用户2(ID=student)的自适应学习界面。

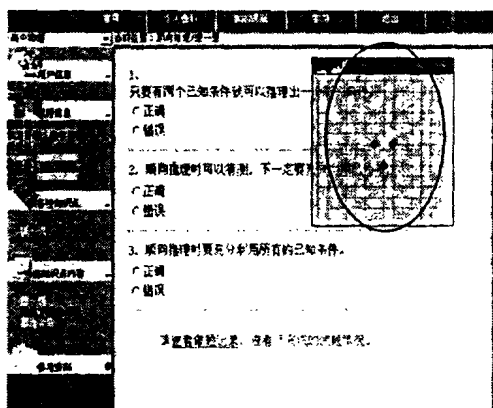


图4 用户1适应性学习界面

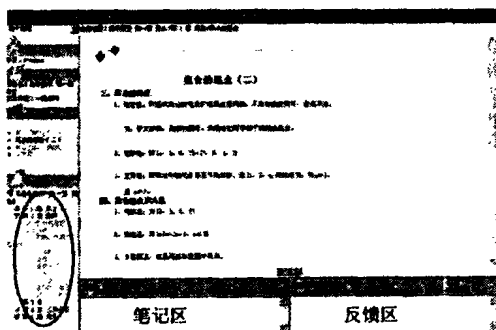


图5 用户2适应性学习界面

用户1是个小学生, 其学习目标是基本了解推理问题, 其

心理特征表现不稳定, 注意力易分散, 但受家人影响喜欢五子棋。界面的设计投其所好, 色彩方面绿色为主黄色跳跃, 比较生动。除提供与他当前学习内容相关的资源(见左边列预备性知识点和当前知识点内容)外, 一段时间后会在浮动的信息提示框显示五子棋游戏进行调节。而用户2是高中学生, 心理基本稳定, 更注重独立思考, 积极反馈学习内容, 所以为当前用户的设计提供了笔记本和反馈功能。另外, 考虑到他学习的条理性, 界面提供了完整的课程目录(见图5椭圆形处)。至于用户操作的自适应, 这两个用户都宁愿选择 GUI 模式。

结束语 自适应用户界面有助于用户学习快速达到学习目标, 促进了人机交互的有效性和高效率。本文研究的自适应网络学习用户界面能够动态适应当前用户和当前学习任务, 为该用户提供与其学习任务和背景知识相关的学习信息, 提高了用户的学习效率。经过原型系统的实验和一个商品化软件的应用, 证明了我们研究思路的可行性和正确性。在自适应用户过程中, 对用户适应性的程度取决于用户模型。下一步工作我们将进一步完善该用户模型。

参考文献

- 1 Lieberman H. Letizia: An Agent That Assists Web Browsing. <http://lieber. www. media. mit. edu/people/lieber>, 1995
- 2 Vivacqua, Lieberman, Dyke. Let's Browse: A Collaborative Web Browsing Agent. <http://lieber. www. media. mit. edu/people/lieber> 1997
- 3 Programming with an Intelligent Agent. Intelligent System. IEEE Computer Society Press, 2003, 18(3): 43~47
- 4 Liu Jiming, Wong Chi Kuen, Hui Ka Keung. An Adaptive User interfaces based on personalized learning. Intelligent System. IEEE Computer Society Press, 2003, 18(2): 52~57
- 5 李晓, 邱玉辉, 虞云川. 基于 MAS 和自适应用户模型的人机交互. 见: 中国人工智能学会全国学术年会, 《中国人工智能进展: 2003》专著, 北京邮电大学出版社(出版中), 2003
- 6 Kobas. Supporting User Interfaces for All Through User Modeling. In: Proc. Of HCI Int'95, 1995. 155~157
- 7 Li Xiao, Li Jianguo, Zhang Xiaozhen. The Tuneup and Integration of Resources in Web-Based Learning. In: Proc. of the IEEE Intl. Conf. on Advanced Learning Technologies 2001, IEEE Computer Society Press. 2001. 391~394
- 8 Li Xiao, Li Jianguo, Zhang Xiaozhen, Qiu Yuhui. Resource Organization and Learning State Controlling for Adaptive Learning System. In: Proc. Of ICALT2002, IEEE Computer Society Press, 2002. 50~52
- 9 李晓, 邱玉辉. 基于 MAS 的 Web 用户数据预处理. 见: 第七届中国人工智能联合学术会, 广西师大学报(自然科学版), 2003, 21(1): 160~163
- 10 非思科技中心. JSP 应用和开发. 电子工业出版社, 2000

(上接第82页)

位信息更新和请求等均采用 LDAP 协议。

LDAP 目录服务器, 一般连接局域网, 也可以是跨区域的互连网。从减少网络流量和提高响应速度出发, 将目录服务代理本地化, 根据不同的管理域和 DIT 划分, 目录服务代理之间可以是自动复制的关系, 也可以是分布式 referrals 关系, 主要依据业务的规模和管理域的规划来决定, 所有的服务定位请求缺省地指向本地目录服务代理, 为了提高定位信息的可获性, 定位请求可以指向一组目录服务代理列表, 当本地目录服务代理不可获得时, 依次绑定列表中的其它目录服务代理, 实现全局资源的有效利用。

结束语 本文提出了一个基于元计算环境下的目录服务模型, 实现动态异构计算资源有效管理, 利用推荐扩展元计算目录服务, 采用服务代理机制, 简化了操作实现。下一步工作中, 由于目录服务管理还与网络监听、故障侦测、网络资源状

态感应等密切相关, 也是亟待进一步研究和完善的。

参考文献

- 1 Fitzgerald S, et al. A Directory Service for Configuring High-Performance Distributed Computations. <http://www.globus.org/>, 1998
- 2 Litzkow M, et al. Condor a hunter of idle workstations. In: Proc. 8th Intl. Conf. on Distributed Computing Systems, 1988. 104~111
- 3 Lee J, et al. Monitoring Data Archives for Grid Environments. IEEE Mass Storage Conf. 2001
- 4 Tierney B, et al. A Monitoring Sensor Management System for Grid Environments, 1999
- 5 Wahl M, et al. Lightweight Directory Access Protocol (v3). Available from <ftp://ftp.isi.edu/in-notes/rfc2251.txt>
- 6 Howes T, et al. The ldap application program interface. RFC 1823, 08/09 95
- 7 Ayd R, et al. A Grid Monitoring Architecture. IEEE, 2002
- 8 Wolski R, et al. The Network Weather Services: A Distributed Resource Performance Forecasting Service for Metacomputing, Future Generation Computing Systems, 1999