Agent 系统适应性设计问题

赵洪彪 李 刚 景 利 俞克群 (北京北方计算中心 100091)

摘要：在智能软件中，适应性相关的功能需求逐渐增多，适应性也是Agent软件的一个重要特征，本文讨论了适应性的概念、相关的设计问题和解决途径。

关键词：Agent 适应性 软件设计

1 引言

软件Agent包括了多种类型，每种类型有各自的特征，目前还不存在某个Agent能够满足全部的特征要求，某个Agent都或多或少具有Etzioni 和Weld以及Franklin 和Graesser所列举的属性特征[1][2] 中的一部分，其中最重要的特征有：

(1) 反应能力 (Reactivity)：有选择地感觉和行动的能力。

(2) 自治性 (Autonomy)：目标驱动的，提前动作和自动开始动作的能力。

(3) 合作行为 (Collaborative Behavior)：可以和谐的与其他Agent协作工作以实现共同的目标。

(4) 知识级通信能力 (Knowledge-level Communication Ability)：使用类似人类语言而不是符号级程序到程序的协议与人或者其他 Agent 通信。

(5) 推理能力 (Inferential Capability)：进行抽象的任务描述，利用先前有关一般目标的知识和更特殊的方法获得灵活性，突破给定的信息限制，建立描述用户、描述情景和描述其他 Agent 的模型。

(6) 时间连续 (Temporal Continuity)：身份和状态具有时间上的持续性。

(7) 人格 (Personality)：具有可信任的人的特性。

(8) 适应性 (Adaptability)：能够学习和改进经验。

(9) 移动性 (Mobility)：能够以自导航的方式从一个硬件平台移动到另外一个硬件平台。

人工智能的研究者对这些属性的混合体进行了划分，按照不同分类模式确定描述术语，提出弱Agent和强Agent的区分，弱Agent具有明显的精神和情绪特征。Moulin和Chaib-draa将 Agent 划为反应型、意向型和社会型。反应型 (Reactive) Agent 能够对环境的变化或者其他 Agent 的消息做出反应，而意向型 (Intentional) Agent 能够根据自己的意图、信念推理，制定行动计划，并且执行这些计划；而社会(Social)Agent具有意向型Agent和其他 Agent 的能力[3]。而适应性功能的设计则是一个没有引起足够重视的设计问题。

2 适应性功能要解决的问题

在目前的应用系统中，非常需要适应性功能，适应性也是 Agent 软件的一个重要特征。下面的一些需求都是与适应性相关的：

(1) 当一个计算机系统观察到用户一遍又一遍执行同样的任务时，系统能否产生一个程序来完成这个任务。

(2) 一个电话簿记录了若干被检索的电话号码，如何利用这些信息提高被频繁查询的号码的易访问性。

(3) 个人辅助程序根据观察到的用户行为进行推断以后，能否在很拥挤的日程里插入新的安排。

(4) 多用户数据库注意到在某一段时间似乎不相关的两条记录X和Y，在某一个会话里面老是被重复检索，它根据这个信息，自动增加当X被检索时Y被检索的可能性，或者相反。

(5) 系统接受日常英语提出的查询，然后对查询进行翻译，顺序地返回相关结果，用户可以选择某个条目，也可以答复，"替我找更多类似这样的东西"。

(6) 在识别系统中，将识别到的手写、语音、手势、绘画或者其他形式的表达从模糊的模拟表示转变成结构化的数字表示。

3 适应性功能带来的设计问题

适应性功能同时带来了新的设计问题：

3.1 用户理解问题

用户往往需要搞清楚系统发生了什么和发生的原因，例如一个智能辅导系统，它的功能是教给小学生物理知识，如果系统注意到当信息用图表表示的时候，学生的学习效果最好，随后系统就改变了它的表达方式，尽量使用图表而避免使用方程表示信息。但学生同时也在观察系统和试图判断系统在做什么，如果学生注意到系统的表达都是图表而不是方程，学生就很自然地想知道为什么，"系统认为我愚蠢吗，如果我做得好，它会变回到方程吗？"。这时候，系统应该采用更好的设计将用户方的误解降低到最小。

3.2 控制问题

控制问题就是用户如何改变系统的问题。如果系统犯了一个错误，无论是在注意阶段，解译阶段还是反应阶段，只要与用户的希望和期待不一致，这时候设计预计的反应应该是什么？

在正常情况下，用户应该得到系统的控制权，撤消适应性功能所做的动作，但是这点并不那么容易做到，因为这并不是一个简单提供undo能力的问题，如果是用户发起的动作，将导致下面的几个问题：

(1) 由于不是用户引发的这种改变，所以用户不清楚撤消什么动作才能让系统恢复到

原来的状态。

(2) 在用户的描述和系统实际发生之间可能还有一个错位，用户注意的可能仅仅是系统动作的某些副作用；

(3) 用户需要辅助功能才能发现实际发生的相关动作，而如何提供这种辅助功能本身又是一个问题。

(4) 用户需要理解系统在第一位置到底发生了什么，需要什么样的系统模型才能让用户理解系统内部的行为？

(5) 不仅需要明确系统的哪些方面可以被控制，还需要向用户提供执行控制的方法和表达。

4 实现适应性功能的结构

适应性功能在设计上通常由下面三种构件实现：

(1) 注意。尝试检测潜在的相关事件。

(2) 解译。尝试采用若干识别规则来识别事件，通常这意味着从外部事件到系统词汇的基本元素做映射。

(3) 反应。用一组行动规则对解译的事件进行操作，或者采取一些动作去影响用户，或者通过学习修改自己的规则。

5 一个实例分析

DowQuest是一个基本命令行界面的商用系统，但是有非常实用的功能，它支持访问最近6-12个月的350种信息资源，允许用户输入类似"告诉我关于阿拉斯加火山喷发的事情"这样的查询请求，返回一系列的文章，然后用户可以要求"给我找更多的像这样的文章"，下面的例子是一个查询过程。

第一次交互

用户："告诉我关于阿拉斯加火山喷发的事情"

DOWQUEST 第一次目录 标题(1-4页)

1 总统寻求放宽对内部限制...，能源杂志，11/27/90 (935字)

2 阿拉斯加火山喷发火山灰，引起震动，DOW JONES新闻服务，01/09/90 (241字)

3 航空运输：火山灰云关闭所有四所...，航空周刊和空间技术，01/01/90(742字)

4 火山爆发阻止空中运输，华盛顿邮报第一部分，01/04/90(679字)

……

第二次交互

用户： 搜索2 4 3

DOWQUEST 第二次目录 标题(1-4页)

1 航空运输：火山灰云关闭所有四所...，航空周刊和空间技术，01/01/90(742字)

2 阿拉斯加火山喷发火山灰，引起震动，DOW JONES新闻服务，01/09/90(241字)

3 火山爆发阻止空中运输，华盛顿邮报第一部分，01/04/90(679字)

4 阿拉斯加的Redoubt火山涌出火山灰，01/03/90 (364字)

……

在第一次交互后，系统不用那些高频度的词去搜索，而是用低频度的词去搜索数据库，然后返回16篇最相关的文章，这种相关性是用统计算法对各种特征进行计算得到，用户无法控制；这时候，用户可以选择阅读某篇文章，或者继续进行下一阶段的查询处理。在第二次交互中，用户告诉系统哪些文章是需要的，用户可以指定整个文章，也可以指定其中一段，系统对整个选择的文字，去掉高频度的噪音词，使用最有信息量的词进行新的查询，然后返回新的最相关的16篇文章。这种过程可以循环多次。

在这个例子中，系统似乎能够理解简单的英语句子，但实际上它仅仅使用了统计的手法，并没有真的理解句子，误解让用户有了更高的期待，而当系统返回一些显然不相干的题目的时候，用户就会认为系统不可信任，而放弃使用，以为DowQuest具有智能的用户最终发现它并没有智能。系统显示结果是以相关性排序的，对计算机专业的人，与一个物体最相关的就是物体自身，所以用户的输入自然是最相关的输出，这毫不奇怪，但用户看到这个结果就会不适应，用户对系统的理解和期待都带来不少的问题。

6 结束语

DowQuest让用户摆脱了查询语言，它不成功的适应性功能设计却值得设计者注意，它在相关性反馈问题上的失败，影响了它的适应性，从而影响到它的功能。相关性反馈是适应性功能实现的一个重要问题，选择合理的方式形成相关性反馈，对系统的解释部件进行合理地选择，就可以在不增加新构件的前提下，改善系统的适应性，更好的发挥系统功能。

参考文献

1 tzioni，O.& Weld，D.S.，1995. Intelligent Agents on the Internet: Fact，fiction and forecast. IEEE Expert，10(4)，44-49。

2 Franklin，S.，and Graesser，A.1996. Is It an Agent or Just a Program? A Taxonomy for Autonomous Agents. In proceedings of the Third International Workshop on Agent Theories，Architectures，and Languages. New York: Spring-Verlag。

3 Moulin，B. and Chaib-draa. B. 1996. An Overview of Distributed Artificial Intelligence. In Foundation of Distributed Artificial Intelligence .3-55，New York:Wiley。

4 Cypher,A. 1991. EAGER: Programming Repetitive Tasks by Example. In Human Factors in Computing Systems, The proceedings of CHI'91, 33-39, New York: ACM Press。

5 Mitchell. et al. 1995，Experience with a Learning Personal Assistant. Communications of the ACM。

6 Dow Jones 1989,Dow Jones News/Retrieval User's Guide. Princeton, N.J.: Dow Jones and Company。