

智能的填写表格式人机接口

郭 江 (北京航空航天大学软件工程研究所)

摘要:大多数填写表格式人机接口很少给用户理解方面的支持,很少直接进行数据类型检查,很少显示静态的 *HELP* 信息,而这些对人机接口又非常重要。因此本文将讨论这些问题的解决办法——智能的填写表格式人机接口。

一、引言

填写表格方式是复杂的人机间操作的一种特别方便的通讯形式。在屏幕显示一系列表格区域,然后由用户往这些区域中输入信息,这样用户就能将有关信息提交给计算机去执行,而不需要象以前那样必须对信息进行详细地组织。例如,用户可以通过填写屏上的表格区域来输入特定的项组合式信息。但这种操作如果用命令语言或半自然语言接口来实现就更为困难了,而还要顺序输入一长串的信息;如果用菜单来实现也不方便,因为菜单的选择项集通常总是以预先确定的顺序来呈现的。因此,选择项和自由响应回答区域同时呈现在相应的屏幕表格区域之中,是填写表格这种人机交互方式最适用的情况。

填写表格方式接口的另一个优点是帮助用户组织输入信息,并将之传送给计算机系统。实际上,表格本身就对这些输入信息施加了一种特定的组织形式。这种接口形式和其它接口类型相对照的特点:是所有输入依赖了以前或以后的输入指令和诊断信息。因此,为用户提供“怎样把他们的任务映射到可用的工具集上”的实时信息。[1]

二、填写表格方式的操作

1.概述

为了理解实现表格的任务,有必要回顾表格的特点和功能。尽管表格有很多定义方式,但最一般的定义还是如下的定义“

“一种结构化的提问和回答会话方式。其中,显示的提问和允许的回答都受到严格的限制。”[2]

这也就是说一个表格能较为方便地组织成一种通讯形式,它可以作为一种特殊种类的通讯介质形式来指导书面的会话。它是一种可见的会话形式而不是动词语言。正是因为具有这样的特点,表格由下列几个部分组成:

- (1)问题和解释;
- (2)回答和选择区域;
- (3)数字化部分和过程化指令。

因而必须由用户来填写表格然后把信息发送给一个特定的目标,这就是与文本式信息不同之处。

MILLER[3](1984)区别了数据查询式和处理式的表格。数据查询式表格服务的目的是企业,是从具体对象中引出确定的状态信息,它们的本质是调查表,是用来决定与一些事物相关的具体对象的,它们的注意力主要集中在设计体上。处理式表格服务的目的是具体对象,是用于传递从企业引出的具体服务的请求,它们的本质是应用表格,设计的目的是用于引出一些信息,这些信息是提供某种请求服务的充分必要条件,因此从这种角度上看,处理式表格具体化了请求服务参数。

处理式的填写表格方式是由用户填写表格式接口实现的,这是由于接口功能就是用这些表格来确定设备服务的申请参数。

2.理解问题

由于书面正式语言的使用,使得理解表格上的问题变得很困难。常常因为用户不能理解表格语言中所包含的术语含义而造成错误。而且,句子语言的复杂程度也影响到理解的难易程度。一般而言,用户不容易理解冗长的句子,而那些包含了肯定或否定表达形式并隐含了大量专业术语的句子就更难理解了。近几年做了大量的

努力来简化表格上的语言;并且提出了各种建议以消除二义性,增加问题的理解程度;在这方面自然语言的处理已有很大的进步。

对用户来说,在理解表格上的问题方面,最困难的是与问题相关的上下文而不是所使用的语言,因而必须努力给用户一个明确、清晰的提问。但是,对表格上的问题设计来说,最重要的问题是一个提问中具体用户决定其准确含义所需的信息随着用户的不同而变化。因此,必须限定提问的范围以便使具体的用户能理解其真正的含义。

3. 回答问题

在正确理解了提问后,用户必须用一个回答来作为响应。但时常会出现一些特殊情况,特别是在一些需要与事实有关的具体信息的情况下,用户可能不知道该如何回答提问。

在大多数情况下,所需的信息可以通过某种方式的查找来决定。而在一些特殊情况下,所需的信息则根本不能访问或只能部分地访问。因此就要确定用户是否只要知道回答的一部分就够了,或者说是否省略一个回答将严重影响他对服务的请求。换句话说,用户在这种情况下需要知道回答的精确程度。

一般用户至少关心回答的精确性的三个方面。第一个是回答问题所需的详细程度。即问题在多大程度上必须专门回答?第二个是回答问题时对错误或不准确的容忍程度。即如果回答仅部分出了错,会有问题吗?第三个是回答信息的重要性。即回答相关的问题有多重要?

实际上,这些问题都是内部相关的。例如,回答出生日期,用户已能想知道是否必须准确到年、月、日。这对那些只记得年的用户来说可能会担心回答的准确性,也就是准确性错误的容忍程度。和这种情况相关的是如果回答不够准确,那么会发生什么情况?是否会被当作无效的输入来处理?但经常的情况是,这些疑问在表格上找不到任何答案。另一个相关的问题是对所需的表格应采取什么回答方式。这也是一个普通的问题。这是由于需要用户的进一步细节情况而引起的。因而随着具体用户情况的不同,用户常常不知该填些什么,而且也没有足够多的线索来给出问题所需的详细程度。这里用户的问题不是决定问题的含义以及与之相关的事情,而是确定的回答它的方式。

4. 表格的导航问题

对一个用户来说,一个主要的困难是通过表格上的那些相关问题来找到他的路径。这困难主要是由于表格的设计通常是用来满足各种各样的人使用,而每个用户的具体环境却又各不相同,因此这种变化的范围越大,具体用户在称为表格导航的过程中所经历的困难就越大。在一些情况下,由于通过了太多不相关问题的导航所引起的混乱可以分别设计表格来加以减轻。但是多个表格总要比一个总的表格代价要高,因此表格的设计者为此经常宁愿在总的表格上改进路径或路径指令。

由于特定的问题总是偶尔与前面回答的提问相关联,因此它们在一个表格上的顺序就部分地依赖于它们之间的逻辑关系。尽管特定的问题之间有逻辑的偶然性,但总是可以设计成相对独立的问题组,然后由这些问题组给出提问顺序。由于不同的问题将占据不同形状和大小的空间,因而这种问题的布置顺序将在表格上产生出明显的路径结构。

这样的讨论就是提醒用户不要希望通过阅读所有表格上的问题来排列出他们之间的依赖关系。而且有些用户还期望通过合适的路径指令和问题布置来使他们离开不相关的问题而转向相关的。因此,当指令和布置不明确时就会产生混淆,就会在表格之间产生不一致。

三、设计智能的填写表格式接口

1. 填写表格方式对支持的需求

一些填写表格问题可能会通过较好的表格设计来减轻,但这种方式不能解决所有的问题。这是由于许多填写表格问题只由特定的用户经历过。因此,屏上表格的一个比较现实的策略是为具体的用户提供合适的支持,减轻用户总体上对这种支持的需求。

那么,什么是为具体用户提供合适的支持呢?对这个问题的回答可以考虑:用户在填写表格时会遇到什么问题、什么困难。例如,一个用户可能不理解一个表格需要什么信息,通过重述问题所提问的东西,定义所用的术语,或解释这些问题在服务请求中的目的等几种方式就能帮助用户解决问题。在用户不能肯定需要什么样的答案以及怎样回答时,可以提供个回答的例子来帮助他,并指出回答中所需的准确程度、有关怎样表示回答和接受反馈的指令、以及由用户所给的回答的一致性和顺序

性。最后,在用户不能确定怎样处理一个表格时,可以给出合适的路径、隐藏不相关的问题并指出完成表格的当前状态来帮助他。

总而言之,用户在完成表格的过程中可能需要下面几种可能的支持和帮助:

(1)问题在请求服务的上下文中以及在当前的具体环境下的解释;

(2)用例子描述的方式来提供有关信息以及在特定的容忍程度内所需的问答;

(3)有关怎样在表格上表示回答的指令;

(4)检查每个回答的有效性以及和表格上其它回答之间的逻辑一致性;

(5)对服务请求的一些可能的回答顺序作一些指示。

(6)一些最佳填写顺序的建议,这些顺序表明了问题和反馈的顺序,因而改善了表格的完成效率。

2.智能在提供联机支持时的作用

在完成纸上的表格所遇到的各种问题也可能在完成屏上的表格时遇到。因此,能利用人工智能技术的方法来给填写表格式人机接口的用户提供一种较满意的支持方式。

例如,为了解释每个提问背后的基本含义、去确定回答所需的准确程度、去检查它们的有效性和一致性以及指示可能的顺序时,就需要与服务有关的知识。特别是一些操作依赖于对一个标准的理解,计算机服务承认该标准而参数则受该标准的限制。而且,在合适的时间提供这种支持就需要持续不断地监控当前的表格状态以及用户完成表格的当前过程。另外,为了提供有关回答表示的合适方法的指令以及有关最佳路径的策略,就需要表格自己本身内容和组织的详细知识。如果这种支持的任何部分均与具体的用户相关联。那么就应该相应于用户在填写表格任务中的理解能力和技巧水平,这也将需要一些有关用户的知识。

因此,就需要在计算机系统中表示大量的知识,以便使其能给用户复杂的填写表格的支持。以一种灵活的和面向用户的方式来表示的知识,提示我们这需要人工智能技术。人工智能的知识系统能很好地满足填写表格式接口的需求,主要是用程序设计语言为这个目的而开发特定的存储和操纵技术。现在有许多系统都是基于

框架的知识表示和自然语言理解。这清楚地表明这种方法建立的计算机系统能够指导用户进行智能会话的潜力。

3.智能的填写表格式接口框架

上面已经描述了填写表格式接口所需的支持,下面将给出一个能满足上述要求的系统框架。这个系统框架是用于表示表格帮助系统的。如图1所示。

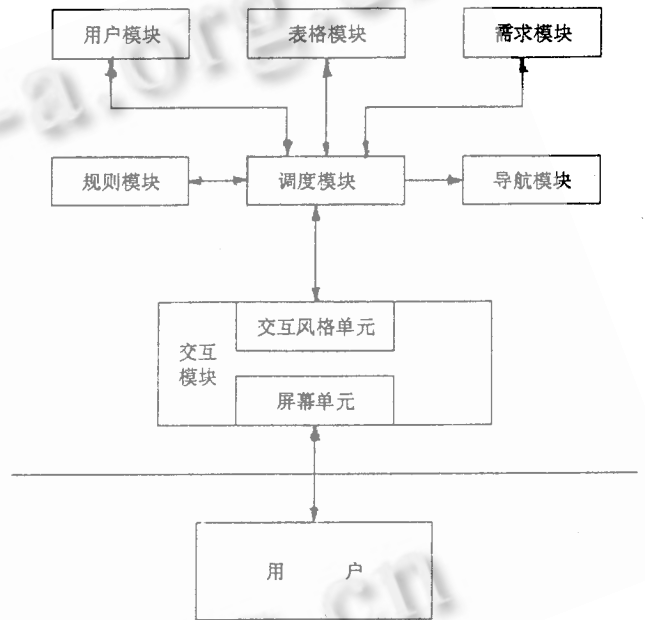


图1

表格帮助系统是由独立而又互相关联的模块组成的。这些模块的关联效果是控制和支持填写表格的过程。每个模块完成一个特定的功能,这是通过其中知识表示的应用特性来实现的。每个模块功能概述如下:

(1)调度模块:模块间的通讯是通过调度模块来完成。一个对象接收并发送出信息。而由调度模块来管理这些信息,并通过选择这些信息传送到模块的次序来控制系统的总体操作。

(2)交互模块:交互模块的功能是和用户之间进行通讯。这个模块的一个组成部分是屏幕单元,它是用来实现窗口管理功能;主要处理问题的显示、回答区域、指令、解释以及对对象的控制等。这个模块也接收所有用户输入到系统的信息并将它们显示到屏幕上合适的地方。另一

个组成部分是交互风格单元,它是用来控制提问和回答方式的。通过显示交互性的选择方法,该模块以适当的方法来满足用户的当前需求,并检查用户输入的有效性。

(3)表格模块:表格模块是与表格本身有关的信息仓库(即问题以及所期望的回答方式的仓库),而且也是当前回答的存储地方,它主要用作由其它模块使用的数据库。

(4)规则模块:规则模块实现了一个运行检查机制来对用户的回答序列提供补充,并且将检查结果直接显示给用户。这种检查也可以显示在系统的其它部分以便表示出回答是否合格。由于用于决定是否合格的有关标准信息放在该模块中,因此也能用它来构造表格上的问题标准。

(5)导航模块:导航模块提供了有关表格中下一个要填写的是哪一个区域的信息。这是通过使用在表格中的路径结构实现的,并且相关于交互模块所指示的用户当前活动。

(6)需求模块:需求模块监控和记录着需求的当前状态。这将用需求模块来检查回答之间的一致性并指示出用户在完成表格时的进展。

(7)用户模块:用户模块提供了与用户有关的信息,它将帮助交互模块匹配用户所需要的会话。这方面至少有两种可行的方法来加以实现。首先是通过直接提问与用户有关的以前经验、填写控制性的表格和操作计算机的经验,就可以沿这个方向来进行分类。这些信息也可以由用户模块来对所需的支持形式作总的提示。其次是,通过监控用户和系统之间的交互,就可能评价用户的当前活动与他的目标有多大程度上匹配。这些信息可以用作特定需求的及时支持。

这种设计方法的优势是对系统结构强加了一种严格的模块特性。这就允许设计者通过不断加入其它模块的方法来构造系统;这就提供了一种自然而又容易的方法来理解和解决不同模块之间的需求冲突;而且这种方法还适合于用户对系统的输入都以同样的方法来进行处理。

四、后记

从上面的分析可以看出,用户希望请求和控制计算机设备的服务。为了实现这一点,用户必须给计算机提供足够的信息以便能确定请求设备服务的条件。在这个过程中用户所需要的支持主要是这些条件的明确解释以及确定它们的方法。

分析中的另一个局限性是表格的具体用户之间各不相同造成的,改善表格设计所减少的对填写表格支持的需求是有限的,这种局限性对接口设计也是同样成立的,在计算机系统范围内总是或多或少地需要联机支持,而且这些联机支持依赖于系统的复杂性以及用户的特征。

最后,完成人机接口的设计至少需要用户、用户任务、交互方式以及任务领域等几个方面的知识。

参考文献:

- [1] Croft W. B. (1984). *The role of content and adaptation in user interfaces*. *International Journal of Man-Machine Studies* 21:283-292
- [2] Wright P. (1990). *Strategy and tactics in the design of forms*. *Visible Language* 14:151-193
- [3] Miller R. B. (1984). *Transaction structures and format in form design*. *Information design*. Chichester: John Wiley & Sons
- [4] Fikes R. E. (1991). *Odyssey: A Knowledge-based assistant*. *Artificial Intelligence* 16:331-361.

IBM 协会第九届年会十月份在昆明召开

IBM 计算机用户协会于 1993 年 10 月 6 日至 8 日在昆明市召开了第九届年会。会上 IBM 公司介绍了一系列用户关心与感兴趣的技术问题:

大型机产品发展方向、开放系统、Right Sizing/ Downsizing、Client/ Server Solutions、AS/ 400、RISC/ 6000。

年会征集到论文 70 多篇,收入论文集内 41 篇,评选出优秀论文 10 篇。