arXiv:2302.03088v1 [cs.RO] 6 Feb 2023

即时绘制机器人程序

David Porfirio 海军研究实验室 美国华盛顿哥伦比亚特区 david.porfirio.ctr@nrl.navy.mil

Laura Stegner 威斯康星大学麦迪逊分校 美国威斯康星州麦迪逊 stegner@wisc.edu

Maya Cakmak 华盛顿大学 美国华盛顿州西雅图 mcakmak@cs.washington.edu

Allison Sauppé 威斯康星大学拉克罗斯分校 美国威斯康星州拉克罗斯 asauppe@uwlax.edu

Aws Albarghouthi 威斯康星大学麦迪逊分校 美国威斯康星州麦迪逊 aws@cs.wisc.edu

Bilge Mutlu 威斯康星大学麦迪逊分校 麦迪逊，威斯康星州，美国 bilge@cs.wisc.edu

摘要

供家庭和工作场所个人使用的服务机器人，需要终端用户开发解决方案，以便在需要时迅速编写机器人任务脚本。许多现有解决方案通过简单的编程接口或限制任务复杂度，兼顾易用性、效率和便利性。另一些解决方案则便于精细设计任务，但往往以牺牲简易性和效率为代价。我们需要能够调和机器人学的复杂性与终端用户开发即时目标的机器人编程解决方案。为此，我们提出一种新颖的多模态即时开发系统 *Tabula*。受一项以原型为基础的形成性设计研究启发，*Tabula* 结合口语和草图：用口语指定机器人任务的核心，再通过草图为核心提供上下文。由此，开发者可以编写不完整、粗略的机器人程序，再由程序合成器补全和完善。最后，我们通过一系列应用场景展示 *Tabula* 的预期用例。

CCS 概念

• **以人为中心的计算** → **用于交互设计的系统与工具**；• **软件及其工程**；

关键词

人机交互、终端用户开发、绘 sketch

ACM 参考格式：

David Porfirio、Laura Stegner、Maya Cakmak、Allison Sauppé、Aws Albarghouthi 和 Bilge Mutlu。2023年。即时绘制机器人程序。载于*2023*年 *ACM/IEEE*人机交互国际会议论文集（*HRI '23*），*2023*年*3*月*13*日至*16*日，瑞典斯德哥尔摩。ACM，美国纽约州纽约市，10页。
https://doi.org/10.1145/3568162.3576991

1 引言

终端用户开发 (EUD) 机器人学解决方案必须让终端用户能够轻松、高效地创建机器人应用，以满足
即时需求。设想这样一个例子：杂货店经理必须引导人们远离饮料

允许为个人或课堂使用而免费制作本作品部分或全部内容的数字或纸质副本，前提是不得为营利或商业利益制作或分发副本，且副本必须附有此声明及首页上的完整引文。必须尊重本作品第三方组件的版权。其他用途请联系所有者/作者。*HRI '23*，*2023*年*3*月*13*日至*16*日，瑞典斯德哥尔摩© 2023年版权归所有者/作者所有。ACM ISBN 978-1-4503-9964-7/23/03。
https://doi.org/10.1145/3568162.3576991

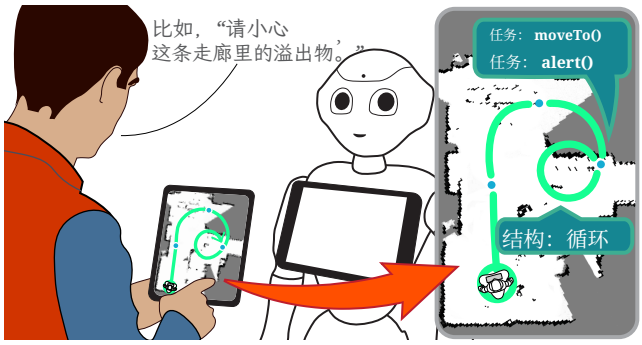

图1：*Tabula* 让机器人的终端用户开发者能够通过勾勒程序结构，为语音添加情境。

通道中的溢出物——这是一项非常适合由机器人执行、看似也很容易描述的任务。经理必须指引机器人前往溢出物所在的位置，同时确保机器人避开溢出物；机器人必须向任何接近走廊的人发出警告；溢出物清理完毕后，机器人必须返回充电站（图1）。

尽管概念简单，但要在需要时立即指定机器人的任务，可能颇具挑战。尽管学习技术凭借其在机器人任务训练中的有效性，有望提供帮助，但离线训练会舍弃关键的社交与环境情境，而在线训练则需要时间。相比之下，最终用户已具备指定任务所需的情境知识。因此，我们认为，面向最终用户的编程工具更适合用于即时任务规范。然而，现有的机器人学工具要么经过不切实际的设计，无法满足即时需求（例如桌面界面），要么要求具备基础编程知识，而这些知识与领域专家的专业知识和技能并无直接关联（例如[21, 36]中的自动机或基于积木块的编程）。相比之下，其他编程范式则通过限制表达能力来弥补不足，例如提供简单程序表示形式的范式（如 [26]中的触发器-动作编程），或者将开发者限定为只能编写机器人某一方面行为的范式（例如只能编写移动行为，而不能指定任务目标，如 [53]所示）。

为满足终端用户开发者随时编程的需求，我们创建了一种新颖的即时终端用户开发（EUD）解决方案 *Tabula*，旨在兼顾简洁性与表达能力。*Tabula* 的指导原则是：尽可能采用无需复杂设备的界面，捕捉开发者快速且不完整的输入，并自动加以完善，

即可实现这一目标。这一目标建立在两项设计选择之上。首先，作者开展并在本文中介绍的一项形成性设计研究表明，部分依赖语音的多模态界面，能够让终端用户轻松、高效地表达希望机器人执行的简单任务。其次，如今触摸屏和语音界面已随处可见，常见于手机、平板电脑和智能手表等设备。因此，终端用户开发者可以便捷地利用触摸交互，为口语陈述补充语境并填补逻辑空缺。

受这些理念的启发，*Tabula* 使最终用户能够通过多模态语音和绘 sketch 输入为机器人编程。在一次录制 会话中，开发者会说出一到两条口语状态陈述，这些陈述对应于任务的主要目标，即任务的核心。为了对核心进行情境化，开发者会在机器人目标环境的二维表示上绘制程序逻辑。录制会话结束后，*Tabula* 的程序合成器利用自动规划技术，通过以下方式组装任务：(1) 将机器人的目标嵌入开发者绘制的路径中；(2) 插入实现这些目标所需的任何其他步骤。如果提供了多次录制，合成器会将由此产生的所有任务计划合并为一个可执行自动机。

因此，我们的主要贡献是编程系统*Tabula*，以及由其实现的终端用户开发（EUD）范式。在本文中，我们首先介绍一项使用纯语音原型开展的形成性设计研究，该研究最终成为构思*Tabula* 的催化剂。随后，我们介绍*Tabula* 本身，重点说明如何整合绘 sketch 来对语音进行情境化并指定程序逻辑。

我们的贡献总结如下：

- 系统——一个功能完备的开发工具，即 *Tabula*，以及一组用于展示其应用的应用场景。
- 设计——一项设计研究，最终形成了用于创建面向机器人的即时终端用户开发工具的设计原则。
- 技术——一种利用程序草图对口语进行情境化的程序合成方法。

2 相关工作

我们的工作借鉴了终端用户开发、自然语言编程、程序合成以及人工智能中的规划（人工智能规划）等领域的文献。下面，我们将简要讨论相关的关键概念和相关工作。

2.1 终端用户开发

终端用户开发（EUD）旨在让新手也能参与编程。Lieberman 等人 [27] 将 EUD 描述为超越应用参数化与定制，并允许用户从头修改或创建程序。值得关注的一种 EUD 范式是触发条件—动作编程（TAP），它已被最终用户广泛采用并取得成功 [48]。然而，尽管 TAP 简单易用，其开发者仍可能在程序中引入不理想或不可预测的行为 [56]。在另一种 EUD 方法中，粗放式编程探索了将粗略的文本输入自动映射到应用程序接口能力的方法 [28]。在作为 EUD 近期流行说法的无代码运动浪潮下，许多商业产品允许用户创建复杂应用，包括用于制作复杂网页的*Webflow*[50], 、用于数据库的*AirTable* [1], , 以及用于自动化的*Zapier* [54]。

人们已在机器人学领域探索了多种 EUD 方法，但它们通常在表达能力上有所局限。这些局限源于限制较多的编程范式（如 TAP） [26, 41]和输入方法（如自然语言） [15], , 也可能是因为可供编程的机器人动作只有一小部分（例如，像 [53] 中那样仅限于运动轨迹）。然而，表达能力更强的终端用户开发界面可能会增加开发者出错的概率，并损害机器人的可靠性。终端用户软件工程（EUSE）领域的既有研究尝试通过向最终用户提供标准的软件工程实践（例如，故障定位）来保障可靠性 [7], , 因此或许也能为机器人学带来助益。草图绘制是机器人终端用户开发与控制中一个熟悉的概念。草图绘制尤其自然的一种用途，是指定机器人的导航路径及周围环境 [5]，或指定其他导航相关命令，例如画一个“X”表示“去这里” [43] , 或者画一个套索表示“清扫这片区域” [40]。*Tabula* 在很大程度上借鉴了*Roboshop*——一个可在机器人的环境俯视图上标注待执行任务的界面 [29],——以及*V.Ra*，一个将导航路径与机器人动作及程序逻辑相结合的任务编写界面 [8]。此外，与*Tabula* 类似的还有 Shah [42] 的研究，该研究将语音与草图绘制结合起来，以指定导航命令；以及 Correa 等人 [12] 和 Teller 等人 [45] 关于实时机器人控制界面的研究，该界面同样融合了语音与草图绘制。在这些研究中，*Tabula* 的创新之处在于能够根据粗略的即时多模态输入合成分支与循环程序。

2.2 自然语言编程

受自然语言在人际互动中广泛使用的启发，研究人员探索了多种不同方法，以实现人与机器人的自然语言交互。语义解析是将自然语言转换为逻辑表示[13, 51, 55], , 常用于通过语言规范来指定命令、目标或简单程序 [9, 32, 47, 49]。研究也表明，基于句法特征的方法在与适当的领域知识相结合时同样有效 [46]。在某些应用中，将受控英语子集直接映射到目标形式化语言已被证明足以满足需求 [24]。

自然语言对话系统通过多轮语言交互，更好地支持复杂指令的传达。在机器人学领域，这些系统使最终用户能够为导航 [25, 47],、组装 [15, 44], 和社交互动 [18]等任务指定可复用程序。一些研究者设想，人机对话可以让未来的家用机器人获取必要的环境特定知识，包括完成某项任务所需的动作，以及支配这个世界的规则 [11]。沿着这一思路，Mohan 和 Laird [34] 开发了一种基于解释的任务学习方法，通过情境化指令教机器人掌握新的层次化任务；后续研究则展示了这些任务表示如何推广到不同情境中 [23]。

2.3 规划与综合

程序合成用于根据不完整的开发者规范 [19] 自动构建完全可执行的程序。在人机交互中，程序合成已应用于机器人操作 [16, 20] 和社交领域 [10]。类似

于 *Tabula*，编程工具 *Figaro* 根据多模态语音和触摸演示合成机器人程序 [38]。不过，*Figaro*要求开发者按照最终程序中各项内容必须出现的确切顺序，依次说出语音和触摸指令。

为了合成程序，*Tabula* 采用了人工智能规划技术。Alterovitz 等人 [3] 将人工智能规划广义地定义为“计算机器人为实现指定目标而采取的动作和执行的运动”。根据 Ghallab 等人 [17], 的分类，我们的方法属于描述性 层面，计划包含机器人要执行哪些动作以及何时执行的信息；而非操作性 层面，后者会精确描述机器人应如何执行这些动作。*Tabula* 从人机交互领域规划技术的显著成果中汲取灵感，包括人类感知任务规划器，它根据社交规则规划机器人的动作 [2]；以及 Petrick 和 Foster [35] 的研究，该研究为社交酒保机器人规划其在多方人机交互中的动作。

3 语音原型：一种输入模式

在本节中，我们介绍我们的语音界面原型，它最终成为开发 *Tabula*的催化剂。[1] 为了尽可能以最少的输入赋予最终用户尽可能多的控制权，该原型探索了仅使用一种输入模态——语音——进行终端用户开发的可行性，原因有二。首先，语音是日常互动中与生俱来的直观沟通方式。其次，感知语音所需的设备极少（即只需一个麦克风），无需在机器人本身之外增添其他设备。

接下来，我们将介绍语音界面原型、对其进行的评估，以及为 *Tabula*提供参考的关键经验。

3.1 语音界面原型

该原型由*Tabula*早期版本的语音输入界面构成，包括一个唤醒词识别器、一个语音转文字引擎和一个语音分类器。在该原型中，语音界面还配有一个简单的视觉反馈界面。

要使用语音界面，最终用户首先说出唤醒词“听我说”，开始设计任务。随后，用户可以通过语音在界面中输入话语，每条话语都会被分配给可用指令集中的某条指令，这些指令包括：(1) 动作指令，用于指定机器人必须执行某项操作；或 (2)事件指令，用于指定机器人监听某个特定触发条件，并对此作出响应，例如有人走近或说话。因此，实例化事件指令是最终用户在程序中编码人类行为的主要方式。在原型中，我们开发了一组规模较小、用于探索的指令。下方列出了部分动作指令（前五条）和事件指令（后两条）：

moveTo: 位置	⟶	移动到 位置
put: 项目、位置	⟶	将指定的项目 放入位置
say: 语音	⟶	说出语音的内容
ask: 语音	⟶	询问语音的内容
讲述： 叙述	⟶	朗读叙述的内容
接近事件	⟶	人接近机器人
言语事件： 语音	⟶	人对机器人说语音

[1]§3 的部分内容曾发表于第一作者的博士学位论文[37]的第7章。§3 重点介绍了这项工作中促成 *Tabula* 开发的相关方面。

在本文余下部分中，我们将指令称为完全实例化的动作或事件，即该指令中的所有参数都已确定。指令类型指的是尚未实例化的指令。例如，**说：**‘你好’的类型是**说**，而实例化指令的参数是‘你好。’

为从话语中推断命令，该原型采用一种非学习型、基于关键词的方法，根据话语中的动词和名词分别与命令类型和参数的匹配程度为命令评分。评分通过在 WordNet [14, 33] 中查询关键词——动词、名词、命令类型和参数——并提取这些关键词的同义词之间的实值距离来计算。例如，在话语“把杂货放进厨房”中，动作命令 **put：** 杂货、厨房 的得分很高，因为“放”“杂货”和“厨房”分别与命令类型和参数相匹配。

如果话语中包含“if”或“when”等关键词，事件命令的得分会高于动作命令。例如，在话语“When someone says ‘hello,’”中，事件命令**eventSpeech：** *‘Hello’*（某人向机器人问候）的得分高于其对应的动作命令 **say：** *‘Hello’*（机器人说“hello”），因为该话语以“when”一词开头。

对于语音命令，我们要求用户提供机器人应说出的确切内容，或机器人能够识别的确切语音。例如，如果终端用户希望指定机器人发出问候，用户可以说：“机器人现在应该说‘很高兴见到你！’”，以生成相应的动作命令 **说：**‘很高兴见到你！’

终端用户开发者说出一系列话语时，原型会生成由相应指令序列组成的程序。该指令序列会显示在视觉反馈界面上，供用户检查。为了编辑正在进行的指令序列，原型包含三条简单的指令：“撤销”用于撤销指令，“重做”用于重做指令，“重置”用于删除所有指令并重新开始。

3.2 形成性评估

为了评估原型中的设计决策，我们通过分别进行的视频通话，对五名参与者开展了一项远程用户研究（3 名男性、2 名女性），年龄为 18 至 43 岁（$M = 24, SD = 10.7$）。参与者几乎没有机器人使用经验，编程经验则各不相同。该研究经机构审查委员会（IRB）批准。

在研究中，参与者接受了如何使用原型的培训，并在模拟家庭环境中完成三项任务（例如，欢迎某人回家）。对于选定的三项任务中的每一项，参与者有三分钟时间为机器人编写程序，并在低保真模拟器中测试机器人；在该模拟器中，参与者可以通过视频通话运行程序。在测试环境中，机器人的图标会在家中移动，并通过麦克风和扬声器与参与者互动。

研究结束时，我们请参与者填写系统可用性量表 *SUS*（包含 10 个条目，采用五级评分量表）[6]以及 *USE* 问卷 [30]„ 该问卷测量有用性、易用性、易学性和满意度（包含 30 个条目，采用七级评分量表）。该原型的系统可用性量表 *SUS* 平均得分为 77（$SD = 17.9$）。在 *USE* 中，参与者以 1 至 7 分进行评分，

该原型的有用性为 4.7（*SD* = 1.85），易用性为 4.45（*SD* = 1.82），易学性为 4.85（*SD* = 2.22），满意度为 4.31（*SD* = 2.15）。

此外，我们还与参与者进行了简短的半结构式访谈（每次5至10分钟），以更深入地了解他们的体验。为分析访谈内容，我们进行了开放式编码，并从访谈数据中提取了关键主题，概述如下。参与者以P1至P5指代。

- 主题*1*：即时任务规范：尽管语音识别和分类存在问题（见下文），一些参与者（P2、P4、P5）仍认为该界面直观易用。P1甚至形容它有望实现"即时"操作。与 [4], 提出的指南一致，P1指出，程序创建后仍需进行修改。

- 主题*2*：口语的局限：参与者表示，使用口语颇为困难，认为它"笨拙且低效"（P2），而且需要他们调整说话风格（P5）。P3还指出了语音本身固有的歧义，例如在任务涉及他人时，可能无法明确说话者指的是谁。

- 主题*3*，关于规范范式的偏好：尽管参与者普遍认可线性任务规范（P4），但P3认为界面"过于简单"，P1和P4则指出，该界面在处理复杂程序时可能存在不足。其他参与者更偏好其他输入方法，例如"打字"（P2、P4）、类似"Scratch"的界面 [39]（P1），或可自定义的"积木"命令（P2）。

除了问卷和访谈数据外，我们还观察到多种使用模式，有助于了解参与者如何使用该原型。参与者平均需要67.4秒（*SD* = 29.6 秒）来规范每项任务。一位参与者未能完成一项任务，因此我们未将该任务纳入平均规范时间的计算。我们观察到，参与者使用语音界面时会遇到困难，原因包括语音转录错误、语音完全未被听到、唤醒词未被识别，以及语音即使被正确听到也被错误分类。可能是由于这些困难，参与者平均每项任务使用"撤销"或"重置"指令1.47（*SD* = 1.30）次。

3.3 原型的启示

现在，我们来讨论由此产生的原型所带来的启示
根据我们的研究结果，这些结果与当前版本的*Tabula*有关。

语音。虽然参与者认为该原型是一个便于即时使用的工具（主题*1*），但他们也表示在使用语音界面时遇到了困难（主题*2*）。我们的客观观察进一步证实了这些困难，参与者使用"撤销"和"重置"指令的次数也反映了这一点（使用模式）。因此，我们认为*Tabula* 应减少对语音的依赖，并支持用户使用不够明确的口头命令。

其他交互模式。根据参与者的反馈（主题*3*），并为了弥补语音使用量的减少，我们确定 *Tabula* 应为用户提供第二个输入通道，以便 *(1)* 无需用户口头说明任务细节，便可推断这些细节；*(2)* 帮助最终用户更有效地理解和管理潜在的程序复杂性。该输入通道应避免需要额外的设备（例如要求使用键盘和鼠标）。

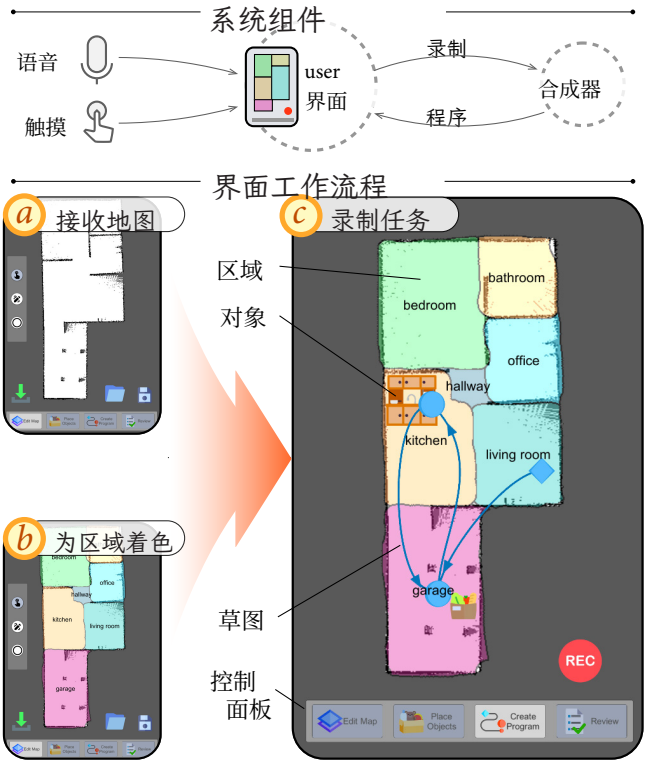

图2：（上）*Tabula* 由一个界面组成，该界面将录制内容传递给合成器，由合成器返回程序。（下）该界面使用户能够：(a) 接收来自机器人的地图，(b) 为重要的地图区域着色，以及 (c) 录制任务。

4 *TABULA*：两种输入模式

这项使用原型开展的设计研究揭示了终端用户开发者仅将语音作为输入方式时面临的各种挑战。鉴于这项研究的启示，以及先前研究所强调的多模态界面的诸多优点（例如包容性和无障碍性 [52]），我们为语音补充了另一种模态——绘 sketch，以创建*Tabula*。我们选择绘 sketch，是为了弱化语音的重要性，让最终用户能够通过触觉方式为一小组核心（可能定义不充分的）命令提供语境。绘 sketch 还使终端用户开发者能够构建难以用语言表达的程序逻辑（例如循环），同时也符合我们的目标：让开发者只需极少的设备即可完成开发任务。

Tabula 系统由两个通过 ROS Noetic[2] 通信的组件实现：一个在 Unity 2020.3.21f13 中实现、基于触摸或触控笔的界面，以及一个在 Python 3 中实现的合成器（图 2，上）。用户获得一张机器人所处环境的二维地图（图 2 下，a），地图上标注了区域（图 2 下，b）；随后，用户口述一组核心命令，并在地图上绘制机器人预期行进的路径（图 2 下，c）。接着，界面会将包含用户语音和草图的录制内容发送给合成器，

[2]http://wiki.ros.org/noetic [3]
https://unity.com/

合成器再向界面返回一个程序。[4]

接下来，我们将介绍：（1）如何配置*Tabula*以供使用；（2）用户随后如何通过语音和草图创建录制内容；最后，（3）如何根据录制内容合成程序。

4.1 准备使用*Tabula*

请看下面这个动机示例：一位用户希望为机器人编写程序，让它每次在用户购买杂货后回家时都来迎接，并帮忙卸下杂货。要使用*Tabula*，必须满足一些技术要求，也就是说，需要明确机器人能力的底层假设，并填充一张供*Tabula*使用的地图。

机器人假设。开发者必须能够使用一台机器人，该机器人能够创建其环境的二维地图，并能在地图中准确定位自身并识别物体。此外，机器人必须配备先进的路径规划器、运动规划器和任务规划器——它应能够导航至环境中的不同区域，与其识别出的物体交互，并在合理范围内处理任务中的边缘情况（例如，如果机器人的目标是拿取杂货，但杂货在用户的汽车里，机器人应知道先打开车门并寻找杂货，然后再拿取它们）。

知识处理。在使用之前，*Tabula*必须具备情境知识。*Tabula*中的知识处理大量借鉴了人工智能规划领域的既有研究，尤其是Petrick 和 Foster[35]，；*Tabula*包含一个固定的领域，用于描述已知且可能存在的实体范围，并假定机器人能够识别这些实体并与之交互（例如，各类物体和人类）、每个实体的语义（例如，“橱柜”是一种“容器”）、一组可用命令（包括机器人要执行的动作或应等待的事件），以及执行某些命令必须满足的前置条件，或执行后成立的后置条件。此外，按照常见做法，*Tabula*会将当前世界状态存储在一个动态且可修改的世界 数据库中。

地图设置。在指定机器人环境中的任务之前，最终用户可以使用 *Tabula* 请求机器人最新的二维地图（图 2 底部，a）。然后，*Tabula*用于标记感兴趣的区域，即地图上机器人需要前往的区域（图 2 底部，b）。最后，用户可以
使用界面将可能也会引起机器人兴趣的物体添加到地图上。例如，用户可以在车库区域放置一个“杂货”图标，将其添加到世界数据库中，并告知机器人可以在车库里找到杂货。将物体放置在区域中的这一步并非硬性要求。

4.2 录制一个任务

终端用户准备好为机器人编写程序时，会创建一段录制，如图 3a 所示。一段录制由一句话 μ 和一幅草图 σ 组成。话语旨在描述机器人要执行的任务核心，而草图旨在将话语与机器人周围的环境联系起来。

以我们的动机示例为例，当终端用户开发者准备开始购物行程时，他们拿出

[4]Tabula *Tabula*的实现工作在威斯康星大学麦迪逊分校进行。*Tabula*的代码和测试用例可在 https://github.com/Wisc-HCI/Tabula 获取。其他辅助材料可在 https://osf.io/jktph/ 获取。

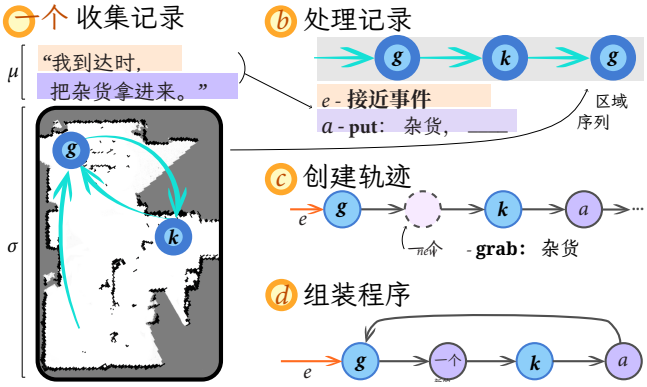

图 3：仅从一次录制内容进行程序合成的技术方法，其中 *e* 表示事件，*a* 表示动作。

手机，启动 *Tabula* 应用，然后按下“录制”按钮。录制过程中，终端用户的第一个动作是说出任务核心：“当我到达时，把杂货拿进来。”*Tabula* 使用斯坦福 CoreNLP 库 [31]，将“当我到达时”识别为从句，并据此将用户的语音拆分为两个独立部分。随后，*Tabula* 采用与 §3.1 类似的方法，将每个从句解析为单独的命令，如图 3a-b 所示，但有几处显著差异。首先，*Tabula* 不采用基于评分的方法，而是改用纯粹的关键词匹配，将 μ 中的名词映射为命令参数，并使用 VerbNet [22]（而非 WordNet）将 μ 中的动词映射为命令类型的同义词。*Tabula* 还支持部分指定命令，例如包含未填写参数的命令。经过这些调整，*Tabula* 将“当我到达时”解析为候选事件命令**接近事件**，将“把杂货拿进来”解析为候选动作命令**put**：杂货，____，其中空白处表示未指定的参数。

在口头说明任务核心之前、期间或之后，终端用户会绘制草图，标出机器人应访问的区域序列。从客厅区域开始，开发者用手指滑向车库区域，再滑向厨房区域，随后返回车库。*Tabula* 将草图 σ 解析为区域序列车库 → 厨房 → 车库，并省略第一个位置（客厅），以免限制机器人只能从地图上的某个特定区域开始任务。图 3a-b 展示了将 σ 解析为区域序列的步骤。

4.3 程序合成与输出

给定终端用户提供的一次或多次录制内容，合成器的目标是 (1) 从每次录制中创建轨迹，并 (2) 组装一个有限自动机，即程序，使其接受每条轨迹。

根据录制创建轨迹。给定一段录制 R，其中包含解析后的话语 μ和解析后的草图σ，合成器必须创建轨迹 t，使其满足解析μ 和 σ所设定的约束。轨迹是由机器人动作组成的序列 $a0 \xrightarrow{e} a1 \xrightarrow{1} \ldots \xrightarrow{n-1} an$，其中 a_i 是第 i 个机器人动作，而 e_i 是第 i 个事件。图 3c 展示了如何从各个组成部分 μ和 σ构造轨迹 t这一任务。

举例来说，回想一下我们的示例：话语是“当我到达时，把杂货拿进来”，草图则是从客厅到车库，再到厨房，最后返回车库。为了清楚说明*Tabula*如何为这段录制创建轨迹，我们先考虑一个更简单的示例，其中车库只经过一次（我们会在§4.3 的完整动机示例中再回到这个问题，循环）。话语仍然被解析为指令 **接近事件** 和`put: 杂货, ____`，但草图则被解析为缩短后的区域序列 车库 → 厨房。对于这个缩短后的草图，合成器的任务是按如下方式创建轨迹 t；未标记的转移表示空事件，即机器人无需提示便可连续执行动作：

空闲接近事件 ——————→ **moveTo:** 车库 → **grab:** 杂货 →**moveTo:** 厨房橱柜 → **put:** 杂货，厨房橱柜

在搜索轨迹 t时，合成器必须自主做出多个决策：（1）应将 μ 中的核心命令插入哪些区域；（2）如何解析这些核心命令中未填充的参数；（3）是否以及在何处需要插入额外的机器人动作，以满足轨迹中每条命令的前置条件；（4）是否以及如何修改世界数据库，以便机器人能够成功完成该轨迹。为做出这些决策，合成器采用 A*搜索，根据离散动作和位置规划出最优轨迹。规划空间包含以下惩罚项：

(1) 轨迹会产生等同于其长度的惩罚项。因此，较长的轨迹比较短的轨迹代价更高。(2) 机器人访问的每个区域或实体，如果机器人未在该位置执行任何动作，都会产生额外的惩罚项。(3) 轨迹中存在但尚未插入世界 的任何实体，都会产生额外的惩罚项。

规划空间包括以下额外约束：合成器只接受满足以下条件的轨迹：(1) 针对原始草图中出现的每个区域，都包含相应的`moveTo` 命令；(2) 包含最终用户的话语所指定的核心命令。如果某个对象存在于已接受的轨迹中，但尚不存在于*Tabula*对机器人环境的最新快照（即世界 数据库）中，则该对象会被添加到世界数据库中。

为了说明这个简化示例中的规划空间，合成器会做出以下决策。在机器人移动到车库之前，插入核心命令**接近事件**；机器人位于厨房时，插入核心`put: 杂货, ____` 命令。在决定如何处理`put: 杂货, ____`命令时，由于其参数尚未填写，无法确定机器人应将杂货放在哪里，合成器会在领域中搜索一个标记为“容器”且位于厨房区域的实体，并使用参数厨房橱柜补全命令。在判断t中是否需要以及在何处需要额外的机器人动作时，合成器根据规划域得知，`put` 的前置条件是机器人必须先持有一个实体，才能将其放到某处。因此，合成器决定在机器人位于车库时插入一条`grab: 杂货` 命令。最后，如果世界 数据库尚未表明杂货 可以在车库中找到，合成器就会相应地修改世界，并因此受到惩罚。

组装程序。 给定一条轨迹，合成器可能无事可做——轨迹本身就成为机器人要执行的逐步程序。如果开发者在录制内容中插入循环，或提供多次录制，合成器就需要进行更多工作。

循环。 在单次录制中，最终用户可以引入循环。为此，最终用户只需在草图过程中多次访问某个区域。举例来说，让我们回到最初的动机示例：其中录制内容仍由 μ“当我到达时，把杂货带进来”构成，而σ再次由机器人从客厅移动到车库、从车库移动到厨房，然后再从厨房返回车库构成。和之前一样，σ被解析为区域序列 车库 → 厨房 → 车库。

合成器会检测草图中的循环（车库 被重复），然后延长循环，使其总共包含两次迭代，且每次循环迭代都完全相同。将车库 → 厨房 视为一次循环迭代后，草图将被延长，使该迭代完整执行两次，生成以下修改后的草图 σ'：车库 → 厨房 → 车库 → 厨房。

要从 μ和 σ'生成轨迹，还需要一个额外的综合约束——对于轨迹中多次访问的任何位置（即区域或实体），在该位置发生的动作和事件序列必须始终相同。因此，生成的轨迹如下：

空闲 接近事件 ——————→ `moveTo:`车库 → `grab:` 杂货 →`moveTo:` 厨房橱柜 → `put:`杂货、厨房橱柜 →`moveTo:` 车库 → `grab:` 杂货 →`moveTo:` 厨房橱柜 → `put:`杂货、厨房橱柜

为了组装最终程序，合成器将重复的动作序列和条件语句组合成一个循环，如图 3d 所示。

多次录制。 提供初始录制后，可以将其他录制附加 到任何现有录制中。附加录制不能从世界中的任意位置开始；相反，它们必须从现有草图中的某个位置分支出来。合成器从每个录制中组装轨迹的方式，与只提供一个录制时并无不同。

从初始轨迹和一个或多个附加 轨迹组装可执行程序非常简单。如果终端用户在位置 l 以一个核心事件 命令开始附加录制（即“当我说‘别再帮我拿杂货了’时”），生成的程序将在 l 处包含一个分支；当该事件发生时，附加录制所生成的轨迹将立即执行。将轨迹彼此附加可能会产生非确定性，例如终端用户开始附加录制时未提供核心事件命令。

5 *TABULA* 的能力与局限

我们通过描述一组应用场景来展示*Tabula*的能力。接下来，我们使用一套共 33 个合成器测试用例，这些用例涵盖了这些场景的不同变体，但不限于此，以分析*Tabula*的可靠性。

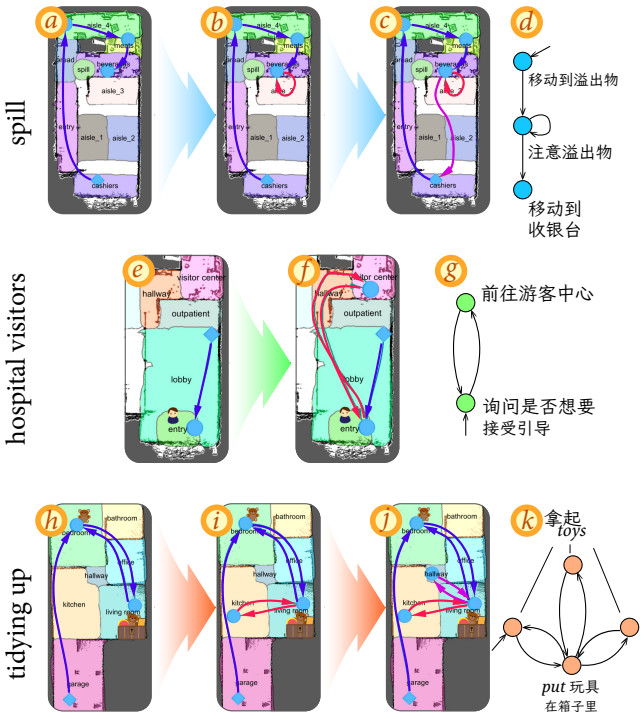

图4：我们的应用场景包括提醒人们注意溢出物（上）、在医院引导人们前往游客中心（中），以及在玩伴聚会后整理玩具（下）。为使图示更清晰，对手绘草图进行了图形增强。

5.1 应用场景

除了 §4 中介绍的杂货店场景外，我们还通过另外三个应用场景展示*Tabula*的能力。

提醒人们注意溢出物。此场景与第1节中介绍的场景相同：杂货店经理需要引导人流避开饮料通道中的溢出物。回顾一下任务要求：机器人必须在避开溢出物的同时移动到溢出物所在位置，向任何接近走廊的人发出警示，并在溢出物被清理后返回起点。此应用场景展示了如何即时进行任务情境化，以及如何利用*Tabula*的分支与循环功能来创建触发器-动作程序。

图4（上）展示了经理为机器人编写程序所采取的步骤。首先，经理通过绘制一个新区域，标明溢出物发生的位置，从而为任务添加背景信息。接着，经理创建一段录制，引导机器人从起点前往饮料区，并特意绕开溢出物，以免机器人从中穿过（图4a）。为确保任何进入机器人附近的人都能注意到溢出物，经理随后在饮料区走廊中绘制一个自环，并说："当有人走近走廊时，说：'请避开此处的溢出物。很快就会有人来清理。'"（图4b）。这段录制的作用是创建一个触发条件-动作程序，只要机器人位于饮料区走廊，该程序就会持续生效——每当有人靠近机器人，机器人就会提醒他们注意溢出物。最后，经理通过绘制一条轨迹，引导机器人返回起点从饮料区走廊返回收银台，并说："当我说'回家'时"（图4c）。图4d以去情境化的高层次示意图呈现了最终程序。

在医院引导访客。设想一位在繁忙医院病区工作的员工，希望简化访客的登记流程。机器人应主动提出护送访客从医院入口前往游客中心。此应用场景旨在突出如何将人类行为编码到程序中。

图4（中）描绘了员工采取的步骤。首先，员工在医院入口放置一个人物实体，向机器人表示它将在此区域遇到人。随后，员工从入口到游客中心画出一条路径，并说道："告诉人们去游客中心的路线。问他们：'您想让我带您过去吗？'"（图4e）。接着，员工说道："如果他们回答'是'，"当机器人听到"是"时，员工便画出一条从入口到游客中心再返回入口的路径（图4f）。图4g以去情境化的高层形式展示了生成的程序。该程序会无限循环：机器人走近医院入口的人，询问他们是否愿意由机器人带路；如果愿意，就带他们前往服务台。

整理玩具。设想一位家长在举办玩伴聚会后，家中散落着玩具。这位家长希望为机器人编写程序，让它从家中的三个特定房间收集玩具，并把玩具放进客厅的箱子里。此应用场景旨在展示循环任务，以及合成器将新对象放入场景中的能力。

图4（下方）展示了家长采取的步骤。家长首先说出"把玩具放进箱子里"，并绘制了一个循环：从机器人的起点出发，前往卧室、客厅，然后返回卧室。根据这一输入，*Tabula*将一个玩具对象放入卧室，并将一个玩具箱对象放入客厅（图4h）。随后，用户再次说出相同的话语，并绘制了一条路径：从箱子出发，前往厨房，然后返回箱子（图4i）。最后，家长第三次说出相同的话语，同时指引机器人前往走廊（图4j）。最终生成的程序以去情境化的高层形式展示于图4k，其中机器人不断从卧室、走廊和厨房拾取玩具，直到没有玩具剩下。

5.2 合成器可靠性

我们的 33 个合成器测试用例（记作 T1-33）使我们能够对*Tabula*的可靠性进行高层次分析。每个测试用例都会向合成器提供以下输入：（1）终端用户的语音，（2）终端用户的草图，以及（3）针对该测试用例定制的世界 数据库。根据这些输入，合成器会生成一个程序作为输出；在 Intel Core i7-1065G7 中央处理器（1.30 GHz）上，每个测试用例平均耗时 3.04 (SD = 0.70) 秒。根据构建测试套件的经验，我们观察到合成器可能无法生成预期输出的原因可分为三类，下面将详细说明。

语音信息不足。尽管*Tabula* 能够稳健地应对终端用户语音中的省略，但如果世界 数据库中也缺少辅助信息，就可能无法合成预期的程序。请看 §4.1 中介绍的杂货 场景，该场景被封装在测试用例 T8 中。假如终端用户

在语音中表达得含糊不清（例如说“把它们拿进来”，而不是“把杂货拿进来”），只要世界 数据库中存在杂货实体，T8 仍会生成正确的输出。然而，如果用户语音和世界数据库中都缺少杂货实体，合成器就会以非确定性的方式从领域中选择一件物品，将其放入世界中供机器人抓取；但该物品可能并不是杂货。

草图信息不足。 虽然机器人核心任务的大量信息通过语音提供，但草图为任务在机器人环境中的情境化提供了信息，也提供了有关程序结构（即分支和循环）的信息。请考虑第 5 节（T16）中介绍的医院 场景。该场景包含一个循环：机器人护送访客完成登记后返回入口。如果开发者没有明确绘制返回入口的路径，则该循环不会被插入程序中。

领域知识不足。 此类失败是指合成器不具备足够的先验领域知识来进行情境化。例如，请考虑第 5 节（T26）中介绍的整理 应用场景。终端用户或许并不希望机器人在捡起玩具和将其放入箱子之间循环，而是希望机器人先收集所有玩具，然后一次性将它们放入箱子。终端用户可能会正确绘制一条经过厨房、卧室和客厅的路径，并说“捡起玩具”。然而，在这种情况下，合成器并不具备足够的关于整理可以采用不同方式完成的先验知识，因此会指示机器人从单个房间而不是每个 房间捡起玩具。

6 讨论

6.1 即时终端用户机器人开发

我们需要能够让终端用户开发者快速、便捷地为突发情境编写机器人程序的工具。尽管机器人非常适合处理这类情境，但能够精心打造高度情境化应用的开发方案可能难以使用且速度缓慢。相比之下，源自机器学习的免手动技术在生成和改进机器人应用方面非常有效，但离线应用这些技术会产生脱离情境的任务规范，而在预期的交互情境中在线应用这些技术则需要耗费大量精力收集数据。借助 *Tabula*，我们认为，快速获取有关当前任务的情境化信息，最佳方式是让终端用户自己通过简单的输入形式提供信息，因为他们是机器人目标情境中的领域专家。

我们认为，*Tabula* 在这一方向上迈出了重要一步。首先，*Tabula* 使用起来很快捷。正如我们的应用场景和测试用例所展示的，只需将一条语音话语与一幅草图配对，即可开发出完整的应用程序。除此之外，*Tabula* 所需的设备也极少，只需一台机器人和一部个人移动设备。再者，*Tabula* 支持任务情境化，因为它能够自定义机器人的环境，并在该环境中对口语命令进行定位。因此，只要机器人能够在其环境中定位，用户就可以在任何场景下使用*Tabula* 创建机器人程序。最后，*Tab-ula* 能够处理复杂任务，而无需用户仔细琢磨任务细节。借助几条简单的口语命令，以及从用户草图中提取的高级程序逻辑，*Tabula* 便可合成一个有限状态自动机。

6.2 局限性与未来工作

Tabula 的一项主要局限是尚未对潜在的机器人终端用户进行评估。因此，我们无法断定 *Tabula* 在即时编写情境化机器人应用方面是否比现有最先进的解决方案更有效，也无法对如何改进 *Tabula* 提出确切的设计启示。因此，我们正在计划开展更多数据收集工作。

其次，尽管 *Tabula* 旨在面向非程序员和技术非专业人员，但我们仍然认为，最终用户必须接受培训，才能充分发挥 *Tabula* 的潜力。具体而言，在单次录制中构建循环，或利用多次录制内容创建带有分支的程序，都需要练习。此外，最终用户可能还需要接受少量培训，以学习如何通过自然语言恰当地说明目标。对 *Tabula* 进行定性探索性评估，将帮助我们准确了解哪些方面需要培训。

第三，*Tabula* 缺乏向最终用户提供反馈以及完善和纠正程序的能力，而这两者对于成功的人机智能系统至关重要 [4]。未来工作首先必须向最终用户提供有关潜在故障或意外程序行为的信息，例如触发事件未充分说明的分支，然后为最终用户提供解决这些问题的方法。要纠正这些问题，就必须扩展 *Tabula* 的完善能力，例如允许最终用户指定录制内容中的特定方面并对其进行微调。

第四，如 §4.1 所述，*Tabula* 要求在任务规范制定之前具备领域知识，包括环境地图。尽管*Tabula* 在一定程度上已经放宽了这一要求——如果终端用户为尚未绘制地图的区域绘制路径，环境映射便可在任务规范制定过程中完成，而不必事先完成——但目前尚不支持与领域中尚不存在的实体交互。关于*Tabula* 的未来工作应在*Tabula* 中集成对新实体的自动分类功能，或修改*Tabula* 的用户界面，提示终端用户为机器人分类这些实体。

7 结论

我们介绍了*Tabula*，这是一个用于即时终端用户开发机器人程序的系统。*Tabula* 的设计旨在满足对简单编程界面的需求，同时保留机器人开发所需的表达能力。因此，我们从头开始设计 *Tabula*，首先制作了一个仅支持语音的初始原型。根据设计研究的结果，我们创建了*Tabula* ，在语音之外增添了另一种输入方式：绘制草图。通过一系列应用场景，我们展示了*Tabula* 如何通过语音和绘制草图创建有意义的机器人程序。

致谢

本研究得到美国国家科学基金会（NSF）资助项目1925043、美国国家科学基金会研究生奖学金计划资助项目DGE-1747503，以及Cisco Wisconsin杰出研究生奖学金的支持。本研究开展期间，DP隶属于威斯康星大学麦迪逊分校；研究完成时，DP是海军研究实验室的国家研究委员会博士后研究助理。

参考文献

[1] AirTable。2022。Airtable | 面向所有人的应用平台。https://airtable.com/。[2] Samir Alili、Rachid Alami和Vincent Montreuil。2009。自主社交机器人的任务规划器。载于《分布式自主机器人系统8》，Hajime Asama、Haruhisa Kurokawa、Jun Ota和Kosuke Sekiyama（编）。Springer Berlin Heidelberg，柏林，海德堡，335–344。https://doi.org/10.1007/978-3- 642-00644-9 30[3] Ron Alterovitz、Sven Koenig和Maxim Likhachev。2016。现实世界中的机器人规划：研究挑战与机遇。《AI Magazine》37卷第2期（2016年7月），76–84。https://doi.org/10.1609/aimag.v37i2.2651[4] Saleema Amershi、Dan Weld、Mihaela Vorvoreanu、Adam Fourney、Besmira Nushi、Penny Collisson、Jina Suh、Shamsi Iqbal、Paul N. Bennett、Kori Inkpen、Jaime Teevan、Ruth Kikin-Gil和Eric Horvitz。2019。人机交互指南。载于《2019年CHI计算机系统人因会议论文集》（苏格兰格拉斯哥）（CHI '19）。美国计算机协会，纽约，美国，1 13 10 1145 3290605 3300233 –。[5] https://doi.org/。/。Federico Boniardi、Abhinav Valada、Wolfram Burgard和Gian Diego Tipaldi。2016。使用草图界面绘制地图和路线，实现自主室内机器人导航。载于《2016年IEEE机器人与自动化国际会议》（ICRA）。2896 2901 10 1109 2016 7487453 –。https://doi.org/。/[6] ICRA。。John Brooke等。。系统可用性量表（SUS）：一种简便粗略的可用性量表。《工业中的可用性评估》189，194（1996），4–7。[7] Margaret Burnett、Curtis Cook和Gregg Rothermel。2004。终端用户软件工程。《Commun. ACM》47卷第9期（2004年9月），53–58。https://doi.org/10.1145/1015864.1015889[8] Yuanzhi Cao、Zhuangying Xu、Fan Li、Wentao Zhong、Ke Huo和Karthik Ramani。2019。V.Ra：一种用于增强现实机器人物联网任务规划的现场视觉创作系统。载于《2019年交互系统设计会议论文集》（美国加利福尼亚州圣迭戈）（DIS '19）。美国计算机协会，纽约，美国，1059–1070。https://doi.org/10.1145/3322276. 3322278[9] David Chen和Raymond Mooney。2011。通过观察学习解读自然语言导航指令。《AAAI人工智能会议论文集》25卷第1期（2011年8月），859–865。https://doi.org/10.1609/aaai.v25i1.7974[10] Michael Jae-Yoon Chung和Maya Cakmak。2022。通过概率功能响应式程序合成人类模拟器。载于《2022年第17届ACM/IEEE人机交互国际会议》（HRI）。727–730。https: //doi.org/ 10 1109 53351 2022 9889630 . /HRI . . [11] Jonathan Connell。。机器人行为的口语编程。arXiv预印本arXiv:1911.09782（2019）。[12] Andrew Correa、Matthew R. Walter、Luke Fletcher、Jim Glass、Seth Teller和Randall Davis。2010。与自主叉车进行多模态交互。载于《2010年第5届ACM/IEEE人机交互国际会议》（HRI）。243 250 10 1109 2010 5453188 –。https://doi.org/。/HRI。。Li Dong和Mirella Lapata。。借助神经注意力，将语言转换为逻辑形式。载于《第54届计算语言学协会年会，ACL 2016——长篇论文》，第1卷。计算语言学协会（ACL），33–43。https://doi.org/10.18653/v1/P16-1004 第54届计算语言学协会年会，ACL 2016；会议日期：2016年07月08日至2016年08月12日。[14] Christiane Fellbaum。1998。WordNet：电子词汇数据库。麻省理工学院出版社。https://doi.org/ 10 7551 7287 001 0001 . /mitpress/ . . Maxwell Forbes、Rajesh P. N. Rao、Luke Zettlemoyer和Maya Cakmak。2015。通过示教式机器人编程，结合情境化空间语言理解。载于《2015年IEEE机器人与自动化国际会议》（ICRA）。 2014 2020 10 1109 2015 7139462 –。https://doi.org/。/ICRA。。Yuxiang Gao和Chien-Ming Huang。。PATI：一种基于投影的增强型桌面机器人编程界面。载于《第24届智能用户界面国际会议论文集》（加利福尼亚州Marina del Ray）（IUI '19）。美国计算机协会，纽约，美国，345–355。https://doi.org/ 10 1145 3301275 3302326 . / . Malik Ghallab、Dana Nau和Paolo Traverso。2016。自动规划与行动。剑桥大学出版社。https://doi.org/ 10 1017 9781139583923 . /CBO Javi F. Gorostiza和Miguel A. Salichs。。通过对话对社交机器人进行终端用户编程。《Robot. Auton. Syst.》59卷第12期（2011年12月），1102–1114。https: //doi.org/ 10 1016 2011 07 009 . /j.robot. . . Sumit Gulwani、Oleksandr Polozov和Rishabh Singh。2017。程序合成。《编程语言基础与趋势》4卷第1-2期（2017年），1–119。https: //doi.org/ 10 1561 2500000010 . / Justin Huang、Dieter Fox和Maya Cakmak。2019。根据一次人类示范合成机器人操作程序。载于《2019年IEEE/RSJ智能机器人与系统国际会议》（IROS）。4585–4592。https://doi.org/ 10 1109 40897 2019 8968543 . /IROS . . Justin Huang、Tessa Lau和Maya Cakmak。2016。服务机器人的快速编程系统设计与评估。载于《2016年第11届ACM/IEEE人机交互国际会议》（HRI）。295–302。https://doi.org/10.1109/HRI.2016.7451765 [22] Karin Kipper、Hoa Trang Dang、Martha Palmer 等人。2000。基于类别构建动词词汇表。AAAI/IAAI 691（2000），696。[23] James R. Kirk 和 John E. Laird。2019。学习层次化符号表示，以支持交互式任务学习和知识迁移。载于第二十八届国际人工智能联合会议论文集，IJCAI-19。国际人工智能联合会议组织，6095 6102 10 24963 2019 844 – 。https://doi.org/ 。/ [24] ijcai。/ Hadas Kress-Gazit、Georgios E. Fainekos 和 George J. Pappas。2008。将结构化英语转换为机器人控制器。《Advanced Robotics》22，12（2008），1343–1359。https://doi.org/10.1163/156855308X344864 arXiv：https://doi.org/ 10 1163 156855308 344864 。/ X Stanislao Lauria、Guido Bugmann、Theocharis Kyriacou 和 Ewan Klein。2002。使用自然语言进行移动机器人编程。《Robotics and Autonomous Systems》38，3（2002），171–181。https://doi.org/10.1016/S0921-8890(02)00166-5《机器人技能学习进展》。[26] Nicola Leonardi、Marco Manca、Fabio Paternò 和 Carmen Santoro。2019。通过触发条件—动作编程个性化人形机器人行为。载于2019年CHI计算机系统人因会议论文集（苏格兰格拉斯哥）（CHI '19）。美国计算机协会，纽约州纽约市，1 13 10 1145 3290605 3300675 – 。https://doi.org/ 。/ 。Henry Lieberman、Fabio Paternò、Markus Klann 和 Volker Wulf。2006。终端用户开发：一种新兴范式。Springer Netherlands，多德雷赫特，1–8。https://doi.org/ 10 1007 1 4020 5386 _1 。/ - - -X Greg Little、Robert C. Miller、Victoria H. Chou、Michael Bernstein、Tessa Lau 和 Allen Cypher。2010。粗放式编程。载于《No Code Required》，Allen Cypher、Mira Dontcheva、Tessa Lau 和 Jeffrey Nichols（编）。Morgan Kaufmann，波士顿，289 307 10 1016 978 0 12 381541 5 00015 8 – 。https://doi.org/ 。/B - - - - 。- Kexi Liu 、 Daisuke Sakamoto 、 Masahiko Inami 和 Takeo Igarashi。2011。Roboshop：用于机器人家务分配与管理的多层草图界面。载于SIGCHI计算机系统中的人为因素会议论文集（加拿大不列颠哥伦比亚省温哥华）（CHI '11）。美国计算机协会，纽约州纽约市， 647 656 10 1145 1978942 1979035 – 。https://doi.org/ 。/ 。Arnold M Lund。。使用USE问卷测量可用性。《可用性与用户体验通讯》（STC可用性特别兴趣小组）8，2（2001年1月），3–6。[31] Christopher D. Manning、Mihai Surdeanu、John Bauer、Jenny Finkel、Steven J. Bethard 和 David McClosky。2014。斯坦福 CoreNLP自然语言处理工具包。载于计算语言学协会（ACL）系统演示。 55 60 14 14 5010 – 。http://www.aclweb.org/anthology/P/P /P - Cynthia Matuszek、Evan Herbst、Luke Zettlemoyer 和 Dieter Fox。。学习解析自然语言命令，以用于机器人控制系统。Springer International Publishing，海德堡，403–415。https://doi.org/10.1007/978-3-319-00065-7 28[33] George A. Miller。1995。WordNet：英语词汇数据库。《Commun. ACM》 38, 11 1995 39 41 10 1145 219717 219748 （11月），– 。https://doi.org/ 。/ 。Shiwali Mohan 和 John Laird。。从情境化交互式指令中学习目标导向的层次化任务。《AAAI人工智能会议论文集》 28, 1 2014 10 1609 28 1 8756 （6月）。https://doi.org/ 。/aaai.v i 。Ronald Petrick 和 Mary Ellen Foster。。在机器人调酒师领域中规划社交互动。《国际自动规划与调度会议论文集》23，1（2013年6月），389–397。https://doi.org/ 10 1609 23 1 13589 。/icaps.v i 。David Porfirio、Allison Sauppé、Aws Albarghouthi 和 Bilge Mutlu。2018。创作和验证人机交互。载于第31届ACM用户界面软件与技术研讨会论文集（德国柏林）（UIST '18）。美国计算机协会，纽约州纽约市，75–86。https://doi.org/ 10 1145 3242587 3242634 。/ 。David J Porfirio。。创作人类与机器人之间的社交互动。博士论文。威斯康星大学麦迪逊分校。[38] David J. Porfirio、Laura Stegner、Maya Cakmak、Allison Sauppé、Aws Albarghouthi 和 Bilge Mutlu。2021。Figaro：用于人机交互的桌面式创作环境。载于2021年CHI计算机系统人因会议论文集（日本横滨）（CHI '21）。美国计算机协会，纽约州纽约市，文章编号414，共15页。https://doi.org/ 10 1145 3411764 3446864 。/ 。Mitchel Resnick 、 John Maloney 、 Andrés Monroy-Hernández 、 Natalie Rusk 、 Evelyn Eastmond、Karen Brennan、Amon Millner、Eric Rosenbaum、Jay Silver、Brian Silverman 和 Yasmin Kafai。2009。Scratch：面向所有人的编程。《Commun. ACM》 52, 11 2009 60 67 10 1145 1592761 1592779 （11月），– 。https://doi.org/ 。/ 。Daisuke Sakamoto、Koichiro Honda、Masahiko Inami 和 Takeo Igarashi。2009。Sketch and Run：面向家用机器人的基于笔画的界面。载于SIGCHI计算机系统中的人为因素会议论文集（美国马萨诸塞州波士顿）（CHI '09）。美国计算机协会，纽约州纽约市，197–200。https://doi.org/10.1145/1518701.1518733[41] Emmanuel Senft、Michael Hagenow、Robert Radwin、Michael Zinn、Michael Gleicher 和 Bilge Mutlu。2021。用于人机协作的情境化实时编程。载于第34届ACM用户界面软件与技术年度研讨会（线上活动，美国）（UIST '21）。美国计算机协会，纽约州纽约市，613–625。https://doi.org/10.1145/3472749.3474773

[42] Danelle Shah。2012。运用草图与语音实现自然且稳健的人机交互。博士论文。康奈尔大学。[43] Danelle Shah、Joseph Schneider 和 Mark Campbell。2010。用于自然机器人控制的稳健草图界面。载于*2010*年*IEEE/RSJ*智能机器人与系统国际会议。4458–4463。https://doi.org/10.1109/IROS.2010. 5649345[44] Maj Stenmark 和 Pierre Nugues。2013。工业机器人的自然语言编程。载于*IEEE ISR*

2013 1 5 10 1109 2013 6695630 。– 。

[45] https://doi.org/ 。/ISR. 。Seth Teller、Matthew R. Walter、Matthew Antone、Andrew Correa、Randall Davis、Luke Fletcher、Emilio Frazzoli、Jim Glass、Jonathan P. How、Albert S. Huang、Jeong hwan Jeon、Sertac Karaman、Brandon Luders、Nicholas Roy 和 Tara Sainath。2010。一种可通过语音命令操作、能在人类身边工作的机器人叉车，适用于准备程度有限的户外环境。载于*2010*年*IEEE*机器人与自动化国际会议。526–533。https://doi.org/10.1109/ROBOT.2010. 5509238 [46] Stefanie Tellex、Thomas Kollar、Steven Dickerson、Matthew Walter、Ashis Banerjee、Seth Teller 和 Nicholas Roy。2011。理解用于机器人导航和移动操作的自然语言命令。*AAAI*人工智能会议论文集25，1（2011年8月），1507–1514。https://doi.org/10.1609/aaai.v25i1.7979[47] Jesse Thomason、Aishwarya Padmakumar、Jivko Sinapov、Nick Walker、Yuqian Jiang、Harel Yedidsion、Justin Hart、Peter Stone 和 Raymond J. Mooney。2019。通过人机对话改进具身自然语言理解。载于*2019*年机器人与自动化国际会议（*ICRA*）。6934–6941。https://doi.org/10.1109/ICRA.2019.8794287[48] Blase Ur、Melwyn Pak Yong Ho、Stephen Brawner、Jiyun Lee、Sarah Mennicken、Noah Picard、Diane Schulze 和 Michael L. Littman。2016。真实环境中的触发条件—动作编程：对20万条IFTTT配方的分析。载于*2016*年*CHI*计算机系统人因会议论文集（美国加利福尼亚州圣何塞）*(CHI '16)*。美国计算机协会，纽约州纽约市，

3227 3231 10 1145 2858036 2858556 – 。https://doi.org/ 。/ 。

[49] Nick Walker、Yu-Tang Peng 和 Maya Cakmak。2019 。。采用匿名化的神经语义解析，用于通用服务机器人。载于*RoboCup 2019*：第二十三届机器人世界杯，Stephan Chalup、Tim Niemueller、Jackrit Suthakorn 和 Mary-Anne Williams（编）。Springer International Publishing，尚贝里，

337 350 10 1007 978 3 030 35699 6_26 – 。

[50] https://doi.org/ 。2022 / - - - - Webflow。 。创建定制网站：无代码网站构建器。https: // webflow.com/.[51] W. A. Woods。1973年。自然语言理解的进展：在月球地质学中的应用。载于*1973*年*6*月*4*日至*8*日全国计算机会议暨展览会论文集 （纽约，纽约）*(AFIPS '73)*。美国计算机协会，纽约，美国，441–450。https://doi.org/10.1145/ 1499586.1499695[52] Marcelo Worsley、David Barel、Lydia Davison、Thomas Large 和 Timothy Mwiti。2018年。面向包容性学习的多模态接口。载于教育人工智能，Carolyn Penstein Rosé、Roberto Martínez-Maldonado、H. Ulrich Hoppe、Rose Luckin、Manolis Mavrikis、Kaska Porayska-Pomsta、Bruce McLaren 和 Benedict du Boulay（编）。Springer International Publishing，尚贝里，389–393。https://doi.org/ 10 1007 978 3 319 93846 2_73 . / - - - - James Young、Kentaro Ishii、[53] Takeo Igarashi 和 Ehud Sharlin。2012年。示范式风格：向机器人教授交互式动作风格。载于*2012*年*ACM*智能用户界面国际会议论文集 （里斯本，葡萄牙）*(IUI '12)*。美国计算机协会，纽约，美国，

41 50 10 1145 2166966 2166976 – 。https://doi.org/ . / . Zapier。 。

[54] 2022 助力你不断前进的自动化。https://zapier.com/. [55] Luke S. Zettlemoyer 和 Michael Collins。2005年。学习将句子映射为逻辑形式：基于概率范畴语法的结构化分类。载于第二十一届人工智能不确定性会议论文集 （爱丁堡，苏格兰）*(UAI'05)*。AUAI出版社，阿灵顿，弗吉尼亚州，美国，658–666。[56] Lefan Zhang、Weijia He、Jesse Martinez、Noah Brackenbury、Shan Lu 和 Blase Ur。2019年。AutoTap：利用线性时序逻辑性质合成和修复触发器-动作程序。载于第*41*届国际软件工程会议论文集 （蒙特利尔，魁北克，加拿大）*(ICSE '19)*。IEEE出版社，281–291。https: // doi.org/10.1109/ICSE.2019.00043