

基于录制回放的移动应用可访问性增强方法

李向民, 沈立炜, 董震

引用本文

李向民, 沈立炜, 董震. 基于录制回放的移动应用可访问性增强方法[J]. 计算机科学, 2023, 50(12): 32-48.

LI Xiangmin, SHEN Liwei, DONG Zhen. [Mobile Application Accessibility Enhancement Method Based on Recording and Playback](#) [J]. Computer Science, 2023, 50(12): 32-48.

相似文章推荐 (请使用火狐或 IE 浏览器查看文章)

Similar articles recommended (Please use Firefox or IE to view the article)

[面向移动应用评分推荐的多任务图嵌入深度预测模型](#)

Multi-task Graph-embedding Deep Prediction Model for Mobile App Rating Recommendation

计算机科学, 2023, 50(9): 160-167. <https://doi.org/10.11896/jsjcx.220700035>

[基于合成图像和Xception改进模型的安卓恶意家族分类方法](#)

Android Malware Family Classification Method Based on Synthetic Image and Xception Improved Model

计算机科学, 2023, 50(4): 351-358. <https://doi.org/10.11896/jsjcx.220300200>

[面向多机器人环境中动态异构任务的细粒度动作分配与调度方法](#)

Fine-grained Action Allocation and Scheduling Method for Dynamic Heterogeneous Tasks in Multi-robot Environments

计算机科学, 2023, 50(2): 244-253. <https://doi.org/10.11896/jsjcx.220500117>

[空间众包任务的路径动态调度方法](#)

Dynamic Task Scheduling Method for Space Crowdsourcing

计算机科学, 2022, 49(2): 231-240. <https://doi.org/10.11896/jsjcx.210400249>

[面向Android第三方库的共谋行为检测](#)

Collusion Behavior Detection Towards Android Third-party Libraries

计算机科学, 2019, 46(5): 83-91. <https://doi.org/10.11896/j.issn.1002-137X.2019.05.013>

基于录制回放的移动应用可访问性增强方法

李向民 沈立炜 董震

复旦大学计算机科学与技术学院 上海 200438

上海市数据科学重点实验室(复旦大学) 上海 200438

(20212010020@fudan.edu.cn)

摘要 移动应用的可访问性指在不受身体和认知障碍的影响下便捷地使用移动应用的能力,其对老年人与残疾人群体具有重要意义。缩短一个应用使用过程中的交互路径(减少操作步骤数)是增强移动应用可访问性的一种重要方式。录制回放技术基于所录制的脚本自动执行交互过程中的固定操作,实现交互操作的缩减。然而,现有的录制回放工具仍存在局限性,包括依赖 ROOT 权限或使用侵入式手段才能进行录制并实现脚本可迁移。另外,已有工具录制的脚本不支持参数化的操作。针对这些问题,提出了基于录制回放的移动应用可访问性增强方法。在该方法的录制过程中,以可访问性服务为媒介,避免申请 ROOT 权限或使用侵入式手段,设计路径索引算法保证脚本的可迁移性,设计脚本参数化算法记录参数化操作,从而生成具有终端迁移性和操作数据泛化性的应用执行脚本。基于该方法开发了录制回放原型工具 RRA 并构建了面向 10 个流行应用的 50 个常见执行脚本。使用这些脚本在同一设备上的回放成功率为 80%,与对比方法 SARA 相当。对 RRA 成功回放的 40 个脚本中的 5 个脚本进行参数化录制,在同一设备上的回放成功率达 100%。对两种方法均能录制成功的 29 个脚本以及 5 个参数化脚本进行迁移执行,RRA 的回放成功率为 94%,高于 SARA。

关键词: 安卓;移动应用;可访问性;录制回放;可迁移

中图法分类号 TP311

Mobile Application Accessibility Enhancement Method Based on Recording and Playback

LI Xiangmin, SHEN Liwei and DONG Zhen

School of Computer Science and Technology, Fudan University, Shanghai 200438, China

Shanghai Key Laboratory of Data Science(Fudan University), Shanghai 200438, China

Abstract Accessibility of mobile applications refers to the ability to use mobile applications conveniently without being affected by physical and cognitive impairments, which is of great significance to the elderly and disabled groups. Shortening the interaction path(reducing the number of steps) during the use of an application is an important way to enhance the accessibility of mobile applications. The recording and playback technology automatically executes fixed operations in the interactive process based on the recorded scripts to reduce the interactive operations. However, existing recording and playback tools still have limitations, including relying on ROOT permissions or using intrusive means to record and realize script migration. In addition, scripts recorded by existing tools do not support parameterized operations. In response to these problems, this paper proposes a mobile application accessibility enhancement method based on recording and playback. In the recording process of this method, the accessibility service is used as the medium to avoid applying for ROOT permission or using intrusive means. The path index algorithm is designed to ensure the portability of scripts, and the script parameterization algorithm is designed to record parameterized operations, so as to generate application execution scripts with terminal mobility and operation data generalization. Based on the proposed method, a prototype tool RRA for recording and playback is developed and 50 common execution scripts for 10 popular applications are constructed. The playback success rate on the same device using these scripts is 80%, which is comparable to the comparison method SARA. Parameterized recording of 5 scripts out of 40 scripts successfully played back by RRA with a 100% playback success rate on the same device. The 29 scripts and 5 parameterized scripts that can be successfully recorded by both methods are migrated and executed, and the playback success rate of RRA is 94%, which is higher than that of SARA.

Keywords Android, Mobile application, Accessibility, Recording and playback, Transferable

1 引言

软件应用的可访问性指应用能够被所有的用户群体方便地使用,包括有视觉、听觉、触觉、语言或认知障碍的用户^[1]。由于残疾人和老年人在使用应用时会受到认知或操作能力的限制,例如盲人难以使用为视觉交互而设计的网站^[2-4],老年人在使用地图时会遇到不少困难^[5],因此可访问性对这类特殊群体具有重要意义。除此以外,可访问性的增强也能为普通用户带来使用体验上的提升^[6-7]。已有诸多工作旨在增强可访问性,例如 W3C 提供了一系列的无障碍原则^[8]以确保特殊人群能够正常地使用 Web;谷歌提供了一系列无障碍能力(如屏幕阅读器)以增强操作系统的可访问性^[9],提供了一系列的无障碍能力开发接口以增强应用的可访问性^[10]。即便有了这些努力,可访问性的增强仍然存在很多的问题与挑战。例如 W3C 提供的无障碍原则不足以确保可访问性^[11-13];目前的大部分移动应用仍然存在很多可访问性问题^[14-18]。其中,移动应用是软件应用的一种特殊形式,随着其逐渐成为日常生活中不可或缺的一部分,增强移动应用的可访问性变得更加重要。

为了增强移动应用的可访问性,解决现有应用的部分可访问性问题,缩短应用使用过程中的交互路径是一种重要的方式^[19-22]。不同于现有工作^[16-18,23-25]通过定位和修复潜在可访问性问题,缩短交互路径以方便用户使用为基点,通过优化使用过程帮助应用被更广泛的用户接纳,进一步增强可访问性。缩短交互路径意味着减少用户使用应用的过程中所需的步骤,简化操作过程,降低出错的可能性。具体来说,当用户使用应用达成某个特定需求时,其使用过程中会出现一些不会因受到需求影响而发生变化的固定操作,通过自动化手段执行固定操作可以减少用户的操作步骤,达到缩短交互路径的目的。举个例子,用户使用美团购买水果时,需要经历 7 个步骤,即启动美团应用、点击蔬菜水果图标、点击搜索框、输入水果名称进行搜索、选择店家、选择水果加入购物车以及结算。其中,启动美团应用、点击蔬菜水果图标、点击搜索框和结算属于固定步骤。因此,可以将使用美团购买水果的交互路径缩短为输入水果名称、选择店家和选择水果。

录制回放技术能够记录用户与移动应用的交互操作并能够在不同设备中自动且有序地执行这些操作^[26-30]。因此,录制回放技术可用于自动执行应用使用过程中的固定操作,从而缩短人工参与的交互路径,本文将其视为有助于增强移动应用可访问性的有效手段。一个理想的录制回放工具应该具备如下 5 个特征^[31]:工具是开源的;工具是非侵入式的,不需要对目标应用进行插桩修改,不需要对操作系统进行定制化修改;不需要获取设备的 ROOT 权限;能够基于组件录制可迁移脚本;能够记录设备的状态、操作的时序关系。然而,现有的录制回放工具^[32-46]无法同时满足这 5 个特征。部分工具通常会采取侵入式手段(对应用使用动态插桩^[32])或获取设备 ROOT 权限^[33]来定位用户与应用交互过程中的目标控件和相应操作。同时,也依赖这两种技术来生成可在不同设备终端上迁移的执行脚本,以避免为不同设备单独定制脚本。另外,已有方法或工具录制的脚本一般不支持在回放过程中

调整执行的操作,需要针对变化的需求重复录制相应脚本。除此之外,虽然用例的跨平台迁移研究中会包含测试用例生成与执行,但是现有工具^[47-49]主要是基于事件转换和探索的方式,依赖于第三方工具,受到这些工具的限制,并且目标和场景约束了它们对一些特定于平台的事件处理能力。

针对以上问题,本文提出了基于录制回放的移动应用可访问性增强方法。首先利用可访问性服务接口在录制过程中获取用户与应用交互过程中的操作信息,以此避免申请 ROOT 权限或使用侵入式手段。其次,设计路径索引算法在录制时构建应用界面的无障碍节点信息对象树,确定目标控件位置并生成目标的路径索引,在回放过程中根据对象树和路径索引定位目标控件,从而保证脚本在不同终端设备上的可迁移性。最后,设计脚本参数化算法,允许用户通过语音交互为特定操作设置执行参数,从而生成具有操作数据泛化性的脚本,使得同一脚本在回放执行时能够识别参数化操作并接收用户通过语音方式给出的特定化输入数据。本文基于所提方法开发了原型工具 RRA(Record Replay Accessibility)并参照 SARA^[34]实验中的录制回放场景构建了面向 10 个流行应用的 50 个常见执行脚本。通过不同的回放实验,从适用性、鲁棒性和可迁移性 3 个维度对 RRA 进行了评估并与 SARA 进行对比。首先,使用 RRA 录制的脚本在同一设备上的回放成功率为 80%,与对比方法 SARA 相当。其次,对 RRA 成功回放的 40 个脚本中的 5 个脚本进行参数化录制,在同一设备上的回放成功率达 100%。最后,在两个屏幕分辨率和大小不同的设备上对两种方法均能录制成功的 29 个脚本以及 5 个参数化脚本进行迁移执行实验,RRA 的整体回放成功率为 94%。在 29 个非参数化脚本的回放中,RRA 成功回放 27 个,多于 SARA 成功回放的 26 个。

2 研究动机

当前,人们对生活类应用的使用需求大大增加,而存在视觉、听觉、触觉、语言或认知障碍的人们更是如此,他们需要使用美团、盒马这些移动应用去购买蔬菜、水果、肉类等菜品来保证每天的餐食。在特殊时期,各个生活类应用都推出了定时发放定量的物资供用户抢购的消费模式。在这种消费模式下,正常用户都难以使用应用购买到所需的物资,对于特殊人群来说更是难上加难。同时,增强移动应用的可访问性既能给正常用户带来更好的使用体验,也能让特殊人群正常使用移动应用,满足日常需求。

以在美团购买水果的场景为例,用户需要经历如图 1 所示的步骤。1)在应用界面中找到美团应用的图标并点击;2)在美团的主界面上找到蔬菜水果图标并点击;3)在界面中找到搜索框并点击;4)在搜索界面输入水果名称;5)输入水果名称后点击搜索;6)滑动查看搜索结果,点击心仪商家进入店铺;7)滑动查看店铺中售卖的水果信息,选择想要购买的水果并加入购物车;8)点击结算按钮。

虽然在美团购买水果的场景并不是很复杂,但是仍会给特殊人群带来较大的使用阻碍。通过对在美团购买水果的场景的观察可以发现,在该场景中如果保持核心需求不变,那么重复执行这个场景完成需求的过程中,除了水果名称和心仪

的内容会因需求而改变之外,其他步骤每次执行时基本不会变化。因此,如果能够缩短美团应用使用过程中的交互路径,简化购买水果的过程,优化重复执行的固定步骤,那么使用美团购买水果的过程就会更为简单,仅由输入想要购买的水果名称、选择心仪的商家和选择想要购买的水果 3 个参数化步骤构成。



图 1 美团购买水果
Fig. 1 Buying fruit on Meituan

因此,为了实现通过缩短交互路径增强应用可访问性的目标,首先,RRA 需要正确地录制和执行应用执行脚本中的操作,满足自动执行 7 个固定操作(点击美团应用的图标、点击蔬菜水果图标、点击搜索框、输入内容后点击搜索、点击心仪商家进入店铺、将想要购买的水果加入购物车以及点击结算按钮)的要求;然后,RRA 需要对输入内容和选择内容进行参数化设置,满足录制和回放 3 个参数化步骤(输入想要购买的水果名称、选择心仪的商家和选择想要购买的水果)的要求;最后,RRA 需要将界面上的关键信息进行播报,提供另一种从信息繁多的界面上捕获到关键信息的方式。

根据上述需求,在录制过程中,RRA 首先请求用户设置脚本名称(如购买水果);其次,用户正常使用美团去购买水果,RRA 自动捕获并记录过程中的 7 个固定操作,即点击

美团应用的图标、点击蔬菜水果图标、点击搜索框、输入内容后点击搜索、点击心仪商家进入店铺、将想要购买的水果加入购物车以及点击结算按钮;接着,当用户执行输入想要购买的水果名称这一步骤时,RRA 触发关键词搜索进行参数化设置;然后,当用户执行选择心仪的商家和选择想要购买的水果这两个步骤时,RRA 触发列表选择,进行参数化设置。

关键词搜索的参数化设置过程如下:RRA 在检测到输入框时,通过语音询问“是否设置提示词”,如果场景中输入的内容不发生变化,就可以不设置。如果希望设置提示词,那么 RRA 会提示“请输入提示词”,然后用户通过语音设置输入框的提示词(如水果名称)。

列表选择的参数化设置过程如下:用户点击元素时,RRA 会检测元素是否在列表中。如果发现元素在列表中,则会通过语音询问“是否设置为列表元素”;如果场景中选择的都是同一个元素,则不设置;如果场景中选择的元素可能会发生变化,那么可以将其设置为列表元素。上述录制完成后,RRA 中会出现名为“购买水果”的参数化应用执行脚本,用户可以通过点击“购买水果”触发应用执行脚本的执行。

根据上述需求,在执行过程中,RRA 会遵循应用执行脚本所描述的交互过程。首先,RRA 有序地执行 3 个固定操作(点击美团应用的图标、点击蔬菜水果图标、点击搜索框);其次,执行到搜索关键词的步骤时(输入想要购买的水果名称),RRA 会提示“请输入水果名称”,用户语音设置具体的水果名称;接着,在输入内容后点击搜索这一步骤被执行后,需要执行选择心仪的商家这一步骤时,RRA 会顺序地播报搜索结果,当播报到用户心仪的结果时,用户通过语音进行确认;然后,在点击心仪商家进入店铺这一步骤被执行后,需要执行将想要购买的水果加入购物车这一步骤时,RRA 会顺序播放店铺中的水果信息,当播报到想要购买的水果时,用户通过语音进行确认;最后,RRA 执行点击结算按钮这一步骤,整个执行过程结束。

综上所述,RRA 希望为日常生活中应用使用场景构建相应的参数化应用执行脚本,通过语音交互控制参数化步骤的执行,简化应用的使用过程,避免重复性的应用交互,缩短应用使用过程中的交互路径,增强移动应用的可访问性,让有视觉、听觉、触觉、语言或认知障碍的人们与正常人一样,能够平等地使用移动应用,享受移动应用给日常生活带来的便捷。

3 研究方法

本文提出基于录制回放的移动应用可访问性增强方法,主要涉及目标控件检索方法、应用执行脚本的参数化方法、应用执行脚本的录制方法以及应用执行脚本的回放方法,根据其工作流程分为录制阶段和回放阶段两个阶段,如图 2 所示。

在录制阶段,首先,RRA 利用可访问性服务获取用户与目标应用的交互情况,如果用户与应用进行了交互并产生了操作事件,那么通过可访问性服务,RRA 可以获取该操作事件对应的无障碍事件对象以及对应的无障碍节点信息对象;其次,利用路径索引生成方法获取被操作的控件的路径索引;然后,利用参数化操作识别方法对当前操作进行识别;最后,结合上述信息使用应用执行脚本录制方法将当前操作编织到

当前的应用执行脚本中。其中,路径索引生成方法旨在获取目标控件在当前界面中的路径信息描述,生成其在当前界面中的路径索引并作为关键标识。参数化操作识别方法旨在判断当前操作是否为参数化操作,生成相应的参数化描述信息;应用执行脚本录制方法旨在根据事件序列有序地记录用户操作,生成相应的操作描述信息,并将其加入到应用执行脚本中的指定位置。

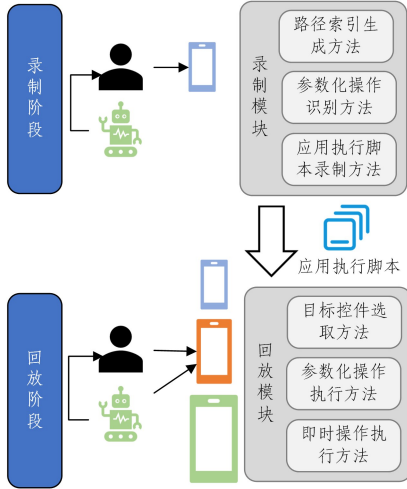


图2 移动应用可访问性增强方法工作流程

Fig. 2 Mobile application accessibility enhancement method workflow

在回放阶段,首先,RRA 读取应用执行脚本,获取待执行的操作,利用目标控件选取方法获取当前界面的目标控件。其次,如果该操作为参数化操作,则使用参数化操作执行方法执行该操作;如果该操作是即时操作,则使用即时操作执行方法执行该操作。最后,重复上述流程直至应用执行脚本回放完毕。其中,目标控件选取方法旨在获取目标控件的无障碍节点信息对象;参数化操作执行方法旨在保证参数化操作的正确执行;即时操作执行方法以操作描述信息为输入,根据控件类型选择执行方法执行操作,旨在保证即时操作的正确执行。

3.1 方法分析

本文的方法是通过缩短交互路径对移动应用的可访问性进行增强,需要使用一种不需要 ROOT 权限且不需要使用侵入式手段的录制回放方法,因为用户目前难以为移动终端申请 ROOT 权限,并且,对最终用户的移动终端和移动应用进行侵入式修改是不可接受的。

受到 Lukic 等^[50]的远程控制工具的启发,本文利用移动终端平台提供的可访问性服务实现录制回放方法。首先,可访问性服务能够从应用接受信息并自定义处理逻辑、代表应用与用户进行交流;其次,该服务的使用不要求设备具备 ROOT 权限;最后,大部分设备都支持可访问性服务,例如安卓 4.0 及后续版本都会提供可访问性服务^[51]。因此,本文使用可访问性服务满足对录制回放的需求。

另外,在移动应用中,控件树一般表示控件在界面中的结构关系,控件在控件树中的位置一般不会随着屏幕分辨率和大小的改变而变化,并且界面的控件树是唯一的,因此,使用

控件在控件树中的路径索引能够在屏幕分辨率和大小不同的移动终端中正确地定位到目标控件,保证应用执行脚本的可迁移性。

除此之外,应用执行脚本代表用户使用应用的需求,虽然用户使用应用实现需求的过程相对固定,但是需求实现的过程中存在一定的变化情况,例如购买的水果由西瓜变为了葡萄、选择的店铺从 A 店变为 B 店,在应用执行脚本中设置参数化步骤能够支持这些变化。

3.2 概念模型

RRA 的概念模型如图 3 所示。对于一个安卓应用而言,用户可以根据需求录制和回放多个应用执行脚本。一个应用执行脚本由脚本编号、脚本名称、目标应用的启动包名和界面名称多个属性构成,包含了多组操作序列。一组操作序列包含多个连续在同一个应用界面中执行的控件操作。一个控件操作由操作编号、执行界面名称、目标控件路径索引等多个属性构成,可以分为筛选操作、即时操作、输入操作等多种类型。回放一个应用执行脚本时,目标应用需要依次通过每组操作序列的应用界面,在特定的应用界面中有序地执行控件操作。

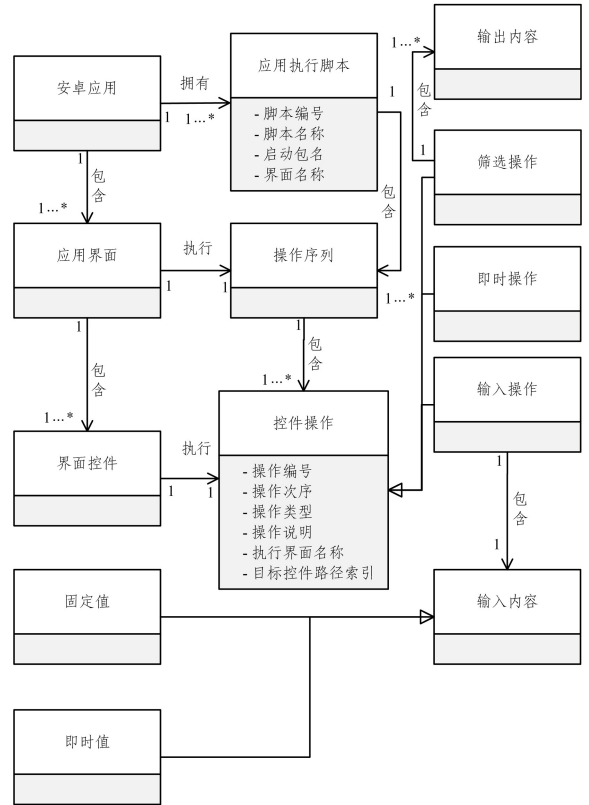


图3 RRA 概念模型

Fig. 3 RRA conceptual model

在本文方法中,不同的控件操作类型有不同的执行处理。首先,即时操作(如按钮点击)是由 RRA 直接根据记录信息自动地执行且不需要输入与输出的操作。然后,输入操作(如登录时输入用户名)指需要输入的操作。一个输入操作会包含一个输入内容(如需要输入的用户名),输入内容既可以是一个固定不变的固定值,也可以是一个根据需求变化的即时值。固定值指用户在录制应用执行脚本时输入的内容(如小明),在回放应用执行脚本时会自动填充。即时值指用户在

回放应用执行脚本时通过语音交互指定的输入内容,需要在录制应用执行脚本时设定输入提示词(如用户名)。输入提示词会被记录在控件操作的操作说明中,在回放应用执行脚本时使用语音播放。最后,筛选操作指用户操作了集合容器中的某个控件且回放过程中目标控件可能会变化为集合容器中的其他控件。一个筛选操作会包含多个输出内容,输出内容由集合元素的关键信息决定。筛选操作在录制应用执行脚本时会在控件的操作说明中记录集合容器的路径索引,在回放应用执行脚本时获取集合元素的关键信息作为输出内容并使用语音播放。

在上述的操作类型中,即时操作和输入内容为固定值的输入操作属于非参数化操作;输入内容为即时值的输入操作和筛选操作属于参数化操作,不但需要在录制过程中设置参数化信息,还需要在回放过程中获取参数化内容。

3.3 目标控件检索方法

根据录制阶段和回放阶段,目标控件检索方法可分为路径索引生成方法与目标控件选取方法。

路径索引生成方法以目标控件为起点,以当前界面的无障碍节点信息对象为根节点,采用自下而上的方式进行检索。在每次向上检索发现新的信息对象时,会扫描该对象的子节点以确定目标信息对象的下标并更新新的信息对象为目标信息对象。如图4所示,目标控件为无障碍节点信息对象20,路径索引生成方法会向上依次检索发现无障碍节点信息对象12、无障碍节点信息对象6、无障碍节点信息对象4以及无障碍节点信息对象1,确定信息对象的下标分别为0,1,0,2,0,从而得到路径索引为0->2->0->1->0。

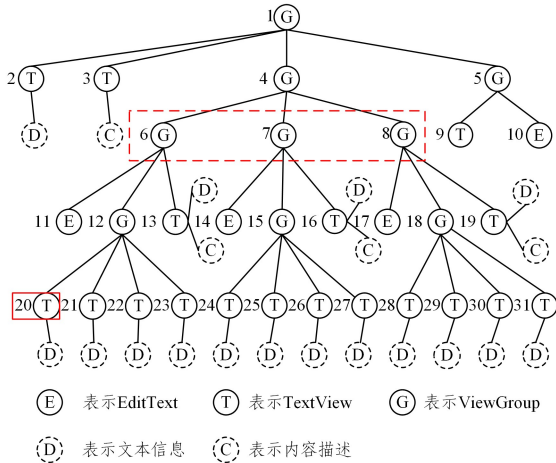


图4 无障碍节点信息对象树

Fig. 4 Accessibility node information tree

目标控件选取方法以当前界面的无障碍节点信息对象作为根节点,根据目标控件的路径索引以自上而下的方式遍历节点树,得到可能的目标控件的无障碍节点信息对象。在自上而下的遍历过程中,同步记录所经过的节点信息。遍历完毕之后,将录制时记录的节点信息描述与此次遍历过程中得到的节点信息描述进行对比,如果两者一致,那么目标节点选取正确;如果两者存在差异,那么目标节点选取有误。如图4所示,根据目标控件的路径索引0->2->0->1->0,

目标控件选取方法会依次经过无障碍节点信息对象1、无障碍节点信息对象4、无障碍节点信息对象6以及无障碍节点信息对象12,得到目标控件的无障碍节点信息对象为节点20。

3.4 应用执行脚本的参数化方法

如图3所示,参数化操作可以分为附带提示的输入操作与筛选操作。其中,附带提示的输入操作的识别较为简单,只需识别出当前的操作是一个输入操作且目标控件是可输入的即可;筛选操作则需要识别出当前操作是一个点击操作且作用于一个集合容器之中,而识别出筛选操作的点击事件所作用的集合容器依赖于算法1。

当录制模块接收了点击类型的无障碍事件对象时,算法会根据目标控件的路径索引获取检索路径中的集合容器(第1行)和每个集合容器中被检索到的子控件(第2行)。如果算法根据目标控件的路径索引未获取到集合容器,那么将返回null,这意味着当前操作是普通的点击操作,其产生的点击事件未作用于集合容器中(第3-4行)。反之,算法会将获取到的这些集合容器作为候选项,执行迭代过程,识别出点击事件作用的集合容器(第5-17行)。在每次迭代中,算法会获取该候选集合容器中被检索到的子控件(第10行),尝试找到候选集合容器中所有与该子控件相似的其他子控件(第12-15行)。如果候选集合容器中被检索到的子控件与其他子控件均相似,那么算法会结束迭代过程并将该候选集合容器返回(第16-17行),这意味着当前用户操作是筛选操作,其产生的点击事件作用于该候选集合容器中。最后,如果算法进行了完整的迭代过程仍未找到点击事件作用的集合容器,那么将返回null(第18行),表示当前操作未找到对应的集合容器,即当前操作非筛选操作。

算法1 筛选操作识别算法

输入:界面的无障碍节点信息对象 root;目标控件的路径索引 path
输出:点击事件作用的集合容器 container

```

1. containers ← getListContainers(root, path);
2. successors ← getListContainerSuccessor(root, path);
3. if containers.size() = 0 then
4.   return null;
5. for index ← 0 to containers.size() do
6.   container ← containers.get(index);
7.   count ← container.getChildCount();
8.   if container.getChildCount() = 1 then
9.     Continue;
10.  successor ← successors.get(index);
11.  similarElements ← ∅;
12.  for compare ← 0 to count do
13.    element ← container.getChild(index);
14.    if isSimilar(element, successor) then
15.      similarElements.add(element);
16.  if similarElements.size() = count then
17.    return container;
18. return null.
  
```

如图4所示,当前被点击的控件为节点21且其路径索引为0->2->0->1->0(1->4->6->12->20)。

在算法 1 中,首先得到了路径上的集合容器节点为节点 1、节点 4、节点 6 以及节点 12;其次,选取节点 1 作为候选集合、节点 4 作为被检索到的子控件,将节点 1 中的子节点 2,3,4,5 进行比较,得到相似元素集合大小为 0;然后,选取节点 4 作为候选集合、节点 6 作为被检索到的子控件,将节点 4 中的子节点 6,7,8 进行比较,得到相似元素集合大小为 3,与节点 4 所包含的子节点数一致,即节点 4 中的全部元素相似;最后,得到该点击事件作用的集合容器为节点 4。

算法 1 的核心是如何判断集合容器中被检索到的子控件与其他子控件是相似的(第 16 行)。对于给定的两个控件 $view_1$ 和 $view_2$,如果两者的相似度 $similarity(view_1, view_2)$ 大于给定的相似度阈值 $threshold$,那么算法将这两个控件识别为相似的。其中,相似度的计算基于如下规则:

1)如果两个控件的类型不一致,那么两者的相似度为 0。如图 4 所示,因为节点 2 为 TextView 类型,节点 4 为 ViewGroup 类型,所以两者的相似度为 0。

2)如果两个控件不属于集合容器且类型一致,那么两者的相似度为 1,当且仅当两个控件的信息对象拥有的描述信息一致,即 $view_1$ 和 $view_2$ 要么都包含文本信息要么都不包含文本信息,以及 $view_1$ 和 $view_2$ 要么都包含内容描述要么都不包含内容描述。如图 4 所示,虽然节点 2 和节点 3 都为 TextView,类型一致,但是由于节点 2 只拥有文本信息、节点 3 只拥有内容描述,因此两者的相似度为 0;因为节点 13 和节点 16 类型一致,且两者都拥有文本信息与内容描述,所以两者的相似度为 1。

3)如果两个控件均为集合容器,那么两者相似度的计算依赖于式(1)。其中,childPairs 是根据子控件类型及其在集合容器中的次序映射得到的相似度计算对。如图 4 所示,节点 6 和节点 7 均为集合容器,两者的相似度利用式(1)计算。首先,得到两个容器的相似度计算对集合为 $\{(11,14), (12,15), (13,16)\}$;然后,分别使用计算对计算相似度得到结果为 $\{1,1,1\}$;最后得到节点 6 和节点 7 的相似度为 1。

$$similarity(view_1, view_2) = \frac{\sum_{(child_1, child_2) \in childPairs} similarity(child_1, child_2)}{childPairs.size} \quad (1)$$

3.5 应用执行脚本录制方法

使用 RRA 录制应用执行脚本的交互过程如图 5 所示。首先,用户选择应用执行脚本对应的目标应用,RRA 通过语音询问脚本名称。其次,用户通过语音设置脚本名称,RRA 启动目标应用以及显示结束录制的悬浮按钮。然后,用户正常操作应用。在用户正常操作应用的过程中,RRA 利用可访问性服务获取操作事件反馈,根据操作事件反馈确定当前的应用界面、目标控件与操作类型,根据操作类型确定是否可以进行参数化信息设置。如果可以设置参数化信息,那么 RRA 通过语音进行一系列参数化信息的询问,用户通过语音进行回应。最后,用户想要结束应用执行脚本的录制时,可以点击结束录制的悬浮按钮,RRA 会将脚本名称、目标应用包名作为应用执行脚本的属性记录下来并且将录制过程中捕获的应用界面、操作信息与参数化信息编织成控件操作的内容进行记录。

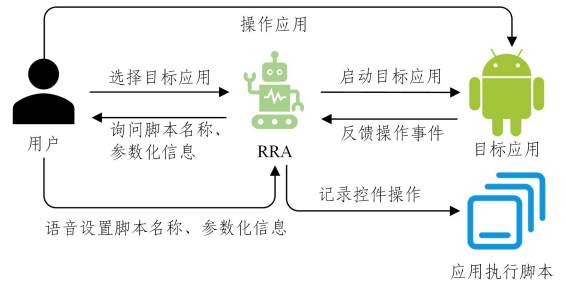


图 5 录制应用执行脚本交互过程

Fig. 5 Interaction process of recording execution script

RRA 录制应用执行脚本的流程如图 6 所示。首先,RRA 启动目标应用,记录当前应用界面。其次,RRA 捕获无障碍事件并确定无障碍事件类型。如果无障碍事件是反映应用界面变化的事件,那么更新当前应用界面的记录;如果无障碍事件属于用户操作事件,那么根据无障碍事件捕获操作信息。

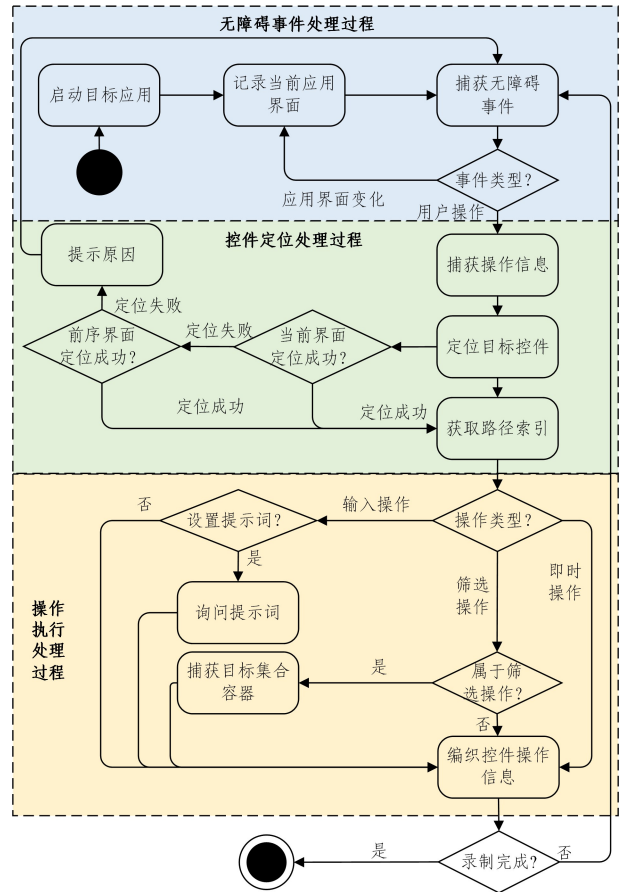


图 6 应用执行脚本录制流程

Fig. 6 Recording execution script process

然后,在无障碍事件属于用户操作事件的前提下,根据操作信息定位目标控件。如果能够当前界面中定位到目标控件,那么获取目标控件的路径索引;如果未能在当前界面中定位到目标控件,那么尝试在前序界面中定位目标控件,若定位到目标控件,则获取目标控件的路径索引。在前序界面中获取到路径索引意味着此次操作产生了界面跳转,实际的操作执行界面为前序界面。如果在当前界面和前序界面中均未定位成功,那么提示定位失败的原因,忽略当前操作并等待捕获

后续无障碍事件。

接着,在获取路径索引的前提下,根据操作信息确定操作类型。如果当前操作是即时操作,那么 RRA 不进行参数化处理。如果当前操作是输入操作,那么 RRA 通过语音询问是否需要为当前输入操作设置提示词。如果用户语音回答是,那么 RRA 通过语音提示用户设置提示词并由用户通过语音进行设置;如果用户语音回答不是,那么 RRA 记录当前输入操作的输入内容为固定值。如果当前操作的目标控件位于集合容器中,那么 RRA 认为这可能是一个筛选操作,会通过语音与用户进行确认,如果用户确认该操作是一个筛选操作,那么会记录当前操作为筛选操作并捕获集合容器的路径索引进行记录。

最后,将当前的控件操作信息编织到应用执行脚本中。如果录制完成,那么结束录制流程,生成应用执行脚本。如果录制未完成,那么 RRA 会等待捕获后续无障碍事件。

在录制过程中,会产生界面跳转的用户操作通常会产生操作事件和界面变化事件两个无障碍事件。而这两个无障碍事件被捕获的顺序是随机的,有可能是操作事件先被捕获,也可能是界面变化事件先被捕获。为了避免无障碍事件被捕获的随机性对录制的影响,本文对目标控件的定位进行了特殊处理,即当前界面和前序界面均尝试定位目标控件。同时,因为开发者可以根据需要对可访问性服务进行防御处理,所以本文的方法有可能无法定位到目标控件。对于这种情况,本文会向用户提示定位失败的具体原因。

另外,由于在输入框内输入多个内容会产生多个输入事件,例如输入“abcdef”7个英文字母会产生7个输入事件,因此需要对输入事件进行特殊处理。在捕获到输入事件时,根据时间阈值判断输入事件的关联性。如果多个输入事件被识别为连续的,那么仅最后一个输入事件会进入录制流程;如果多个输入事件被识别为独立的,那么这些输入事件均会进入录制流程。

3.6 应用执行脚本回放方法

使用 RRA 回放应用执行脚本的交互过程如图 7 所示。首先,用户选择需要回放的应用执行脚本,RRA 根据应用执行脚本启动目标应用。其次,RRA 从应用执行脚本中获取控件操作。如果控件操作属于参数化操作,那么 RRA 会与用户进行语音交互以获取参数化内容并执行控件操作。如果控件操作属于非参数化操作,那么 RRA 直接执行控件操作。最后,RRA 利用可访问性服务获取控件操作执行结果,检验控件操作的执行情况。

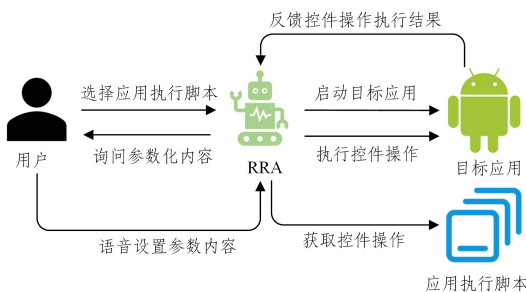


图 7 回放应用执行脚本交互过程

Fig. 7 Interaction process of replaying execution script

RRA 回放应用执行脚本的流程如图 8 所示。首先,RRA 根据用户选择的应用执行脚本启动目标应用,记录当前应用界面。其次,RRA 根据当前应用界面从应用执行脚本中获取操作序列,依次获取控件操作,基于当前应用界面利用控件操作中的目标控件路径索引定位目标控件。如果能够正确定位到目标控件,那么识别操作类型;如果无法定位到目标控件,那么在重试次数阈值内尝试重新定位。如果超过限定次数仍未定位到目标控件,那么提示定位失败的原因,继续回放后续控件操作;如果在限定次数内定位到目标控件,那么继续识别操作类型。

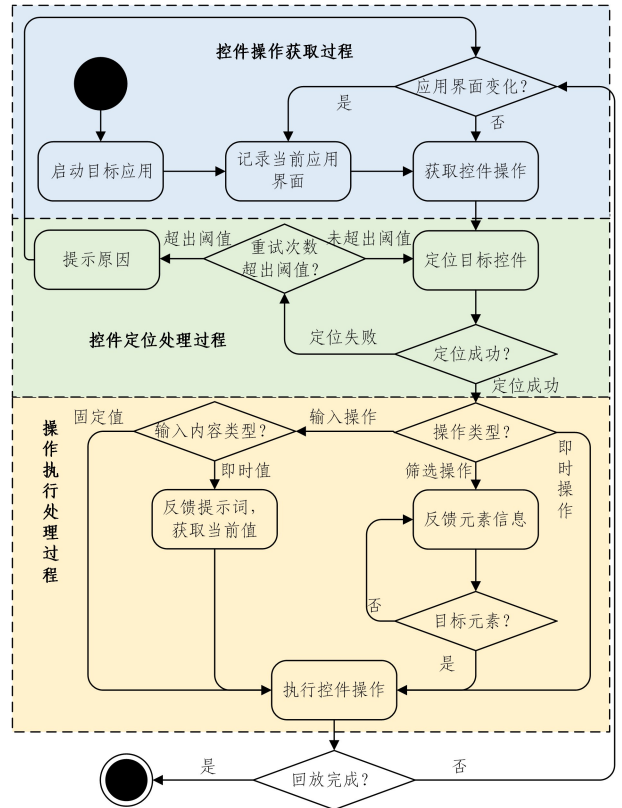


图 8 应用执行脚本回放流程

Fig. 8 Replaying execution script process

然后,在定位到目标控件的前提下,根据操作类型进行相应的回放处理。如果当前的操作是即时操作,那么 RRA 执行控件操作。如果当前操作是输入操作,那么会根据输入内容类型进行处理。输入内容为即时值时,RRA 利用语音播放提示词,用户根据提示词利用语音设置相应的输入内容,RRA 根据用户设置的输入内容执行控件操作;输入内容为固定值时,RRA 利用固定值执行控件操作。如果当前操作是筛选操作,那么 RRA 会利用集合容器的路径索引去定位集合容器,根据目标控件的路径索引检索集合元素的描述信息,利用语音播放提取出关键信息。如果用户不选择当前描述的集合元素,那么会获取下一个集合元素;如果用户选择了当前描述的集合元素,那么 RRA 会将当前的集合元素作为目标控件执行控件操作。

最后,如果回放完成,那么回放流程结束。如果回放未完成,则判断当前执行的控件操作是否会导致应用界面发生变化。如果当前执行的控件操作会导致应用界面发生变化,则

RRA 会更新当前应用界面的记录,然后获取控件操作并继续后续流程。如果当前执行的控件操作不会导致应用界面发生变化,那么 RRA 会获取控件操作并继续后续流程。

在回放过程中,无障碍信息节点树的更新延迟会导致目标控件定位失败。为了避免无障碍信息节点树更新延迟的影响,本文增加了目标控件定位的延迟重试机制。另外,对于筛选操作,回放过程会涉及集合容器的滚动,为了避免反馈重复的元素信息以及确定是否滚动到底部,本文会对集合容器滚动前后的元素信息进行对比。除此之外,在回放过程中,本文会将当前界面和目标执行界面进行对比,如果不一致,那么会在前序界面中进行对比以提高应用执行脚本的回放成功率。

4 系统实现

RRA 是一个安卓应用,由脚本管理服务、语音交互服务、可访问性服务、脚本录制服务与脚本回放服务 5 个核心服务构成。其中,脚本管理服务主要负责管理应用执行脚本;语音交互服务负责为用户提供语音交互能力;可访问性服务负责在应用执行脚本录制与回放过程中反馈与模拟控件操作;脚本录制服务负责在应用执行脚本录制过程中记录控件操作,编织应用执行脚本;脚本回放服务负责在应用执行脚本回放过程中有序地模拟控件操作,正确地回放使用场景。

4.1 脚本管理服务

脚本管理服务主要负责管理应用执行脚本的注册、应用执行脚本内容的可视化,以及应用执行脚本的执行和删除。应用执行脚本的注册信息包含脚本编号、脚本名称、目标应用的启动包名和界面名称。应用执行脚本的内容包含注册信息和操作序列,操作序列中的操作信息包含操作编号、操作类型、目标控件的路径索引、执行界面名称和备注信息。其中,目标应用的启动包名指应用的完全限定名称,例如美团的启动包名是 com.sankuai.meituan;界面名称指应用界面的完全限定类名,例如美团首页的名称是 com.sankuai.meituan.MainActivity。此外,脚本管理服务还起到了应用管理服务的作用,负责获取应用列表、启动和关闭目标应用。脚本管理服务利用安卓的 PackageManager 服务获取手机中安装的应用信息以实现获取应用列表的能力,应用信息包含应用程序的图标、名称、包名等内容。脚本管理服务启动目标应用是通过传递应用的启动包名、界面名称显示调用消息传递对象的方式实现。脚本管理服务关闭目标应用是通过调用安卓的 ActivityManager 服务来关闭后台应用的方式实现。

4.2 语音交互服务

语音交互服务通过集成第三方语音引擎支持语音识别能力与语音合成能力来达到为用户提供语音交互能力的目的,目前其实现使用的是科大讯飞的语音引擎,包含流式版的语音听写能力和流式版的在线语音合成能力。其中,语音听写能力用于识别用户的语音;在线语音合成能力用于将文本转换为发音人的语音进行播放。

在应用执行脚本注册时,语音交互服务使用语音合成能力询问执行脚本的名称,使用语音识别能力获取脚本名称以

补充应用执行脚本的注册信息;在应用执行脚本的录制过程中,语音交互服务使用语音合成能力询问参数化信息,使用语音识别能力获取用户的回答信息以帮助用户设置参数化操作;在应用执行脚本的回放过程中,语音交互服务使用语音合成能力播放提示词、列表元素描述信息,使用语音识别能力获取提示词内容、确定所选择的列表元素。

4.3 可访问性服务

可访问性服务是获取界面交互信息和模拟用户执行控件操作的关键服务。其中,界面交互信息由无障碍事件对象 (AccessibilityEvent) 和当前界面的无障碍节点信息对象 (AccessibilityNodeInfo) 构成。本文通过继承 AccessibilityService 类实现了 RecordReplayAccessibility 类,并在 RRA 的应用清单中声明 RecordReplayAccessibility 类为无障碍服务,初步创建了可访问性服务。为了能够在系统启动可访问性服务时调整可访问性服务的配置信息,本文重写了 onServiceConnected 方法并自定义了可访问性服务的配置文件。

如图 9 所示,本文调整了界面交互事件的类型 eventTypes 的配置,让可访问性服务能够监听点击类型、文本改变类型、窗口状态改变类型、选择类型、滚动类型以及长按点击类型 6 种常见的界面交互事件。同时,为了能够获取控件在界面中的层级结构,本文设置了 canRetrieveWindowContent 属性为 true。通过这个属性的配置,本文能够在检测到界面交互事件时,利用无障碍事件对象执行 getSource 方法得到目标控件的无障碍节点信息对象,利用该对象可以得到生成该事件的目标控件,获取目标控件的父视图和子视图的信息,从而得到控件在界面中的层级结构信息。除此之外,为了获得更多的用户交互信息,本文调整了 accessibilityFlags 属性。另外,为了能够在系统检测到交互事件时获取无障碍事件对象,本文重写了 onAccessibilityEvent 方法。当 onAccessibilityEvent 方法被触发时,可访问性服务接收到了相应的无障碍事件对象,利用 getRootInActiveWindow 方法可以获取当前界面的无障碍节点信息对象,将两者组成界面交互信息,然后根据 RRA 的状态对界面交互信息进行分发处理。

```
<accessibility-service xmlns:android="http://schemas.android.com/apk/res/android"
```

```
    android:accessibilityEventTypes="typeViewClicked|typeViewTextChanged|typeWindowState|typeViewSelected|typeViewScrolled|typeViewLongClicked"
```

```
    android:accessibilityFeedbackType="feedbackAllMask"
```

```
    android:accessibilityFlags="flagDefault|flagRetrieveInteractiveWindows|flagIncludeNotImportantViews|flagReportViewIds|flagRequestTouchExplorationMode|flagRequestFilterKeyEvents"
```

```
    android:canRetrieveWindowContent="true"
```

```
    android:description="@string/accessibility_service_description"
    android:notificationTimeout="100" />
```

图 9 可访问性服务配置文件

Fig. 9 Accessibility service configuration

4.4 脚本录制服务

脚本录制服务是基于可访问性服务提供的界面交互信息来实现应用执行脚本的录制。目前,本文实现了 RecordProcessor 类来管理完整的脚本录制过程,可访问性服务通过调用该类的 processEvent 方法传递界面交互信息。其中,RecordProcessor 主要关注点击类型、文本改变类型、选择类型、长按点击类型以及窗口状态改变类型这 5 种类型的界面交互信息。通过关注窗口状态改变类型的界面交互信息,脚本录制服务能够根据无障碍事件对象获取当前应用界面的名称并进行记录。通过关注点击类型、文本改变类型、长按点击类型以及选择类型这 4 种类型的界面交互信息,脚本录制服务能够捕获每个应用界面内的用户交互序列,完成应用执行脚本的编织。当用户在目标应用中执行了相应类型的操作时,脚本录制服务可以接收到相应的界面交互信息,并且可以利用无障碍信息对象获取界面交互事件的目标控件的信息对象。

对于即时操作,脚本录制服务将路径索引、操作信息、界面信息等基本信息整合生成控件操作。对于筛选操作,脚本录制服务利用目标控件检索方法生成该筛选操作所对应的集合容器的路径索引,借助语音交互服务确认设置是否为筛选操作。如果确认设置为筛选操作,那么将集合容器的路径索引及其遍历节点的描述信息作为补充信息,结合基本信息生成控件操作;反之则直接使用基本信息生成控件操作。对于输入操作,脚本录制服务识别出输入操作后,借助语音交互服务确认是否设置提示词。如果需要设置提示词,那么将提示词信息作为补充信息,结合基本信息生成控件操作;如果不需要设置提示词,那么直接使用基本信息生成控件操作。

4.5 脚本回放服务

脚本回放服务能够根据脚本录制服务编织的应用执行脚本有序地回放用户操作。目前我们实现了 ReplayProcessor 类来管理应用执行脚本的回放过程,可访问性服务通过调用该类的 processEvent 方法传递界面交互信息。其中,ReplayProcessor 主要关注窗口状态改变类型和滚动类型两种界面交互信息。通过关注窗口状态改变类型的界面交互信息,脚本回放服务能够确定用户当前执行操作的应用界面,为应用执行脚本的回放提供了确实的信息反馈。同时,利用该类型的界面交互信息,脚本回放服务能够获取当前应用界面的无障碍节点信息对象,为目标控件的检索提供了根节点信息,保证了目标控件获取的正确性。通过关注滚动类型的界面交互信息,脚本回放服务能够有效地执行筛选操作的回放。根据筛选操作的定义,这类操作的目标控件大多都位于列表之中,对目标控件的筛选会涉及列表滚动。利用滚动类型的界面交互信息,脚本回放服务能够获取完整的候选控件集合,保证了其目标控件获取的有效性。

对于筛选操作,脚本回放服务可以通过目标集合容器的路径索引获取相应的信息对象。然后,脚本回放服务根据目标控件的相对路径索引获取候选控件集合,利用候选控件的信息对象获取相应的描述信息,通过语音交互服务将描述信息反馈给用户,由用户根据控件的描述信息确定目标控件。

最后,脚本回放服务在用户选定的目标控件的信息对象上执行特定的操作。对于输入操作,脚本回放服务根据目标控件的路径索引获取相应的信息对象。然后,如果输入内容是固定值,则获取控件操作记录的固定值;如果输入内容是即时值,则获取控件操作记录的提示词,通过语音交互服务获取用户需要输入的即时内容。最后,在信息对象上执行特定的操作,将获取的内容输入目标控件中。

5 实验评估

为了评估本文提出的基于录制回放的移动应用增强方法的有效性,本文设计实验来探究如下 3 个问题。

1) RQ1(适用性):对于真实应用的应用执行脚本,RRA 提供的录制能力和回放能力如何?

2) RQ2(鲁棒性):为真实应用的使用场景重新构建参数化的应用执行脚本,在不同的参数设置下,RRA 的回放效果如何?

3) RQ3(可迁移性):RRA 编织的应用执行脚本在不同设备上回放的效果如何?

5.1 实验准备

下面主要介绍进行实验研究之前所做的准备工作和相关设置,包括实验数据集的选择与构建、实验过程中使用到的硬件设备的信息与软件要求,以及对比方法。

5.1.1 数据集

在实验中,RQ1 主要研究 RRA 是否支持真实应用的应用执行脚本录制与回放。这个问题的探究需要选取一些真实应用的使用场景,本文从 SARA 的实验数据集中通过人工筛选得到了如表 1 所列的实验场景。

在 SARA 的实验数据集中,一共有 53 个应用,这些应用被划分为 24 种类型,每个应用都有 5 个使用场景。首先,从该数据集中选取 10 种与日常生活息息相关的应用类型;然后,通过应用商店下载这些应用;最后,记录这些应用的版本号、应用大小、类别以及相应的使用场景,构成本文的实验数据集。其中,由于 Gmail 无法在实验设备上运行,本文使用 QQ 邮箱作为 Gmail 的替代。同时,为了让实验与研究动机中的案例接近,本文使用美团作为 Trivago Hotel Search 的替代。另外,本文实验数据集的使用场景中包含了很多人们在日常生活常见的操作类型,可以分为以下 5 种类型。

1) 在线购物:在线浏览商品或服务,选择心仪的商品或服务进行购买(A5, F2, 共 2 个)。

2) 信息获取:通过应用提供的搜索能力或直接从候选列表中浏览,获取需要的信息(A1, A2, A3, A4, B1, B2, C3, D1, D2, D3, E3, F3, H2, H5, I5, J2, J4, 共 17 个)。

3) 信息管理:自主调整应用中的个人信息(B3, B4, C1, C2, C4, C5, D5, E1, E2, E5, F4, F5, G3, H3, H4, I1, J3, 共 17 个)。

4) 发布信息:通过应用向外分享个人的看法(B5, D4, G1, G5, H1, 共 5 个)。

5) 特定任务:为了达到个人目标,激活特点任务(E4, F1, G2, G4, I2, I3, I4, J1, J5, 共 9 个)。

表 1 实验数据集
Table 1 Experimental datasets

编号	应用	版本号	分类	大小/MB	使用场景
A	美团	12.6.405	生活休闲	267	1. 搜索酒店;2. 筛选酒店;3. 酒店排序; 4. 确认酒店;5. 预订房间
B	淘宝	10.20.0	购物	476	1. 搜索产品;2. 浏览商店;3. 管理购物车; 4. 增加一个新的地址;5. 写评论
C	谷歌相册	5.33.0.362978573	图片	134	1. 管理相片;2. 创建相册;3. 搜索相片; 4. 存档照片;5. 创建电影
D	QQ 音乐	12.0.5.8	音乐	456	1. 搜索音乐;2. 浏览主页;3. 看音乐视频; 4. 写评论;5. 管理播放列表
E	Keep	7.44.0	健康与健身	375	1. 增加计划;2. 开始锻炼;3. 浏览锻炼项目; 4. 做挑战;5. 设置
F	支付宝	10.3.36.7200	金融	522	1. 转账;2. 订票;3. 浏览交易记录;4. 添加联系人; 5. 开始群聊
G	微信	8.0.32	社交	777	1. 发朋友圈;2. 打开链接;3. 新的群聊; 4. 转账;5. 评论
H	QQ 邮箱	6.4.1	邮件	165	1. 发邮件;2. 看邮件;3. 添加/移除收藏; 4. 切换账户;5. 搜索邮件
I	Citymapper	6.21.1	地图导航	188	1. 设置家庭地址;2. 导航;3. 寻找公交车; 4. 寻找公交车/路线;5. 阅读帖子
J	谷歌计算器	8.3(477856174)	工具	6.27	1. 计算公式;2. 查看历史记;3. 改变答案格式; 4. 获取帮助;5. 复制粘贴

RQ2 的实验数据集是基于 RQ1 的实验结果构建的。首先,从 RQ1 的数据集中筛选出实验成功的使用场景;其次,根据场景类型和场景步骤选取合适的场景;然后,对被选中的场景进行参数化调整;最后,将这些使用场景作为 RQ2 的实验数据集。

RQ3 的实验数据集是基于 RQ2 和 RQ1 的实验结果构建的。首先,从 RQ1 和 RQ2 的数据集中筛选出实验成功的场景;其次,从这些成功的场景中选取与设备无关的场景;最后,将这些实验成功的且与设备无关的场景及其对应的应用执行脚本作为 RQ3 的实验数据集。

5.1.2 实验环境

在本文的实验中,需要使用两个移动设备进行实验,具体参数分别如表 2 和表 3 所列。其中,RQ1 和 RQ2 使用实验设备 A 进行应用执行脚本的录制和回放实验;RQ3 使用实验设备 B 进行应用执行脚本的迁移实验。

表 2 实验设备 A

Table 2 Experimental equipment A

设备名称	IQOO5	屏幕分辨率/像素	2340×1080	系统版本	Android12
运行内存	12GB	手机存储/GB	256	型号	V2024A
硬件版本号	MP_0.1	软件版本号	PD2024C_A_8.11.1	内核版本号	4.19.125-pef+
基带版本	MTEFS_PACK-			处理器	2.84GHz 骁龙 865 八核
	1.383597.41.397522.94				

表 3 实验设备 B

Table 3 Experimental equipment B

设备名称	华为 P20	屏幕分辨率/像素	2244×1080	系统版本	Android10
运行内存/GB	6	手机存储/GB	64	型号	EML-AL00
EMUI 版本号	10.0.0	版本号	10.0.0.188 (C00E186R1P5)	内核版本	4.14.116
基带版本	21C20B369S015C000			处理器	HiSilicon Kirin 970

5.1.3 对比方法

SARA 是一个开源的桌面应用形式的录制回放工具,其使用动态插桩技术实现了基于组件的录制脚本并且通过屏幕自适应算法保证了录制脚本的迁移性。本文实验使用 SARA 作为对比系统。具体而言,本文在 RQ1 和 RQ3 的实验研究中将 RRA 的结果与 SARA 的结果进行对比分析。

另外,因为 SARA 录制的脚本不具备参数化能力并且目前未找到相应的工具或系统具备相似参数化能力,所以本文在 RQ2 的实验研究中不和类似的系统进行对比,主要探索 RRA 在不同的参数设置下回放应用执行脚本的效果。

5.2 适用性(RQ1)

为了评估 RRA 的适用性,本文使用实验设备 A 尝试为表 1 中的 50 个使用场景构建相应的应用执行脚本,测试这些应用执行脚本能否被成功地录制和执行,然后针对录制失败或执行失败的应用执行脚本进行讨论分析。

RQ1 的实验结果如表 4 所列。在 50 个使用场景中,RRA 可以成功地录制和执行其中的 40 个,另外的 10 个录制或执行失败(A5,B4,E5,G1,G3,J2,J3,J4,F2,F3);SARA 可以成功地录制和回放其中的 40 个,剩余的 10 个(A5,B4,D3,E4,F1,F2,F3,G4,H2,H5)录制或回放失败。从整体结果来看,RRA 和 SARA 录制和回放的表现相当,两者录制回放的成功率都是 80%。而在不同类别的使用场景下,两者各有优劣。

1)在线购物:RRA 的表现与 SARA 相同,均有 2 个场景(A5,F2)失败。

2)信息获取:RRA 成功地执行了 14 个场景,其余的 3 个场景(F3,J2,J4)是失败的;SARA 成功地执行了 13 个场景,其余的 4 个场景(D3,F3,H2,H5)是失败的。RRA 的表现略优于 SARA。

3)信息管理:RRA 成功地执行了 13 个场景,其余的 4 个场景(B4,E5,J3,G3)是失败的;SARA 成功地执行了 16 个场景,其余的 B4 场景是失败的。RRA 的表现略逊于 SARA。

4)发布信息:RRA 成功地执行了 4 个场景,剩余的 1 个 G1 场景是失败的;SARA 成功地执行了 5 个场景。RRA 的表现逊于 SARA。

5)特定任务:RRA 成功地执行了 9 个场景;SARA 成功地执行了 6 个场景,其余的 3 个场景(E4,F1,G4)是失败的。RRA 的表现优于 SARA。

表 4 应用执行脚本执行情况

Table 4 Script execution

应用		执行情况				
美团	A1	√	√	√	√	×
	R.	√	√	√	√	×
淘宝	B1	√	√	√	×	√
	S.	√	√	√	×	√
谷歌相册	C1	√	√	√	√	√
	R.	√	√	√	√	√
QQ 音乐	D1	√	√	×	√	√
	S.	√	√	×	√	√
Keep	E1	√	√	√	√	×
	R.	√	√	√	×	√
支付宝	F1	√	×	×	√	√
	S.	×	×	×	√	√
微信	G1	×	√	×	√	√
	S.	√	√	√	×	√
QQ 邮箱	H1	√	√	√	√	√
	R.	√	×	√	√	×
Citymapper	I1	√	√	√	√	√
	S.	√	√	√	√	√
谷歌计算器	J1	√	×	×	×	√
	S.	√	√	√	√	√

虽然 RRA 整体的脚本成功率和 SARA 相当,但是 RRA 的学习成本和使用难度较低,更容易被用户接受且脚本可读性强。

首先,RRA 是一个界面简单的安卓应用,用户只需要了解少量的关键概念并将 RRA 安装到安卓终端中便可以用它进行录制回放;而 SARA 是一个桌面应用,用户需要具备编程基础,并且需要利用 ADB 将安卓终端与电脑进行连接,在连接的基础上进行录制回放,用户的学习成本和使用难度较高。

其次,RRA 主要依赖于安卓的可访问性服务,不需要对应用进行修改,不会影响应用的性能和稳定性,更容易被用户接受;而基于动态插桩技术实现的 SARA 需要在运行时对应用进行修改,可能会影响应用的性能和稳定性,同时对应用进行修改还可能引入不安全的代码,大部分用户对此并不能接受。

最后,RRA 会通过列表的形式展示所录制的脚本,并将操作类型、路径索引、参数化信息等脚本中的操作描述以表格的形式展现,让用户能够直观地获悉脚本信息;而 SARA 直接展示其根据用户需求录制得到的 Python 脚本,用户需要对脚本代码进行阅读才可获悉脚本信息,可读性较差。

在 RRA 失败的 10 个场景中,场景 A5,G1,G3,J2,J3,J4,F2 以及 F3 失败的原因是录制得到的应用执行脚本存在步骤的遗漏。具体遗漏的步骤如图 10 所示,场景 A5 缺少了点击预订按钮预订房间的步骤;场景 G1 缺少了点击按钮打开朋友圈内容编辑窗口的步骤;场景 G3 缺少了点击发起群聊按钮的步骤;场景 J2 缺少点击更多选项中的历史记录的操作;场景 J3 缺少了点击更多选项中的主题修改的步骤;场景 J4 缺少了点击更多选项中的帮助的步骤;场景 F2 缺少了点击购票的步骤;场景 F4 缺少了点击账单的步骤。如果对这些遗漏步骤的应用执行脚本进行手工修复,增加这些遗漏步骤的信息,那么 RRA 仍然能够正确地执行这些应用执行脚本。

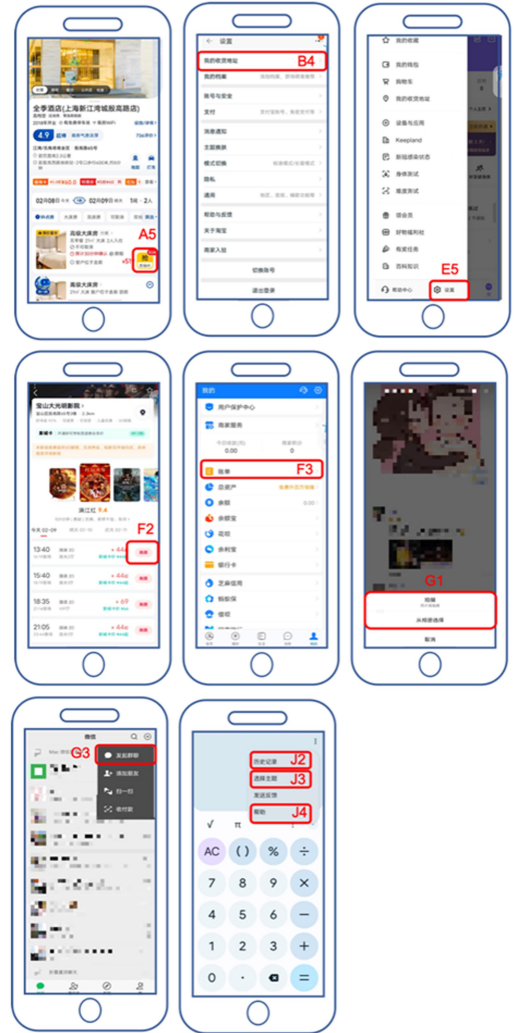


图 10 失败的步骤

Fig. 10 Failure steps

通常来说,点击操作一般会产生选择类型的无障碍事件或点击类型的无障碍事件,如果点击事件导致了界面变化,那么还会产生窗口状态变化类型的无障碍事件。通过对录制过程中的日志进行分析发现,场景 A5,G1,G3,J2,J3,J4,F2 和 F3 遗漏的步骤在录制过程中,产生了滚动类型的无障碍事件和窗口状态变化类型的无障碍事件,未产生选择类型的无障碍事件或点击类型的无障碍事件,导致本文的工具没有有效地捕获到这些点击操作,进而导致最终的应用执行脚本无法满足使用场景的要求。另外,场景 F2 的失败原因属于这类

失败场景中的极端情况,F2的点击购票的操作不仅没有产生点击类型的无障碍事件,也没有产生点击购票操作后触发界面的变化时预期的窗口状态变化类型的无障碍事件。

出现这类现象的主要原因是应用的开发者对可访问性服务进行了防御处理,例如通过利用 onTouch()方法替换 on-Click()方法、重写 performAccessibilityAction()方法等手段对点击事件或其他事件进行屏蔽。对于这类由于产生了非预期的无障碍事件或不产生无障碍事件导致的问题,可以通过优化无障碍事件接收时的处理逻辑尝试修复,例如利用事件序列建立反馈机制,遇到不合理的事件序列利用回滚、阶段性回放等方式进行修复。不过,可访问性服务面对这些“危机意识”较强的应用时处于天然的劣势,本文的方法在面对这类应用时存在一定的局限性。

在 RRA 失败的 10 个场景中,场景 B4 和 E5 失败的原因是录制过程中无法获取目标控件的路径索引。在场景 B4 的录制过程中,执行点击我的收货地址时,利用目标控件检索方法未能获取“我的收货地址”的路径索引,原因是该界面的无障碍节点信息对象树中没有包含该控件的无障碍信息对象。无独有偶,场景 E5 中需要点击侧边栏中的设置按钮,目标控件记录在侧边栏的无障碍节点信息树中,无法在界面的无障碍节点信息对象树中被检索到,因此利用目标控件检索方法也没有获取到“设置”的路径索引。

表 5 RQ2 实验场景

Table 5 Experimental scenario in RQ2

编号	应用	场景名称	场景描述	参数化说明
P1	美团	确认酒店 (A4)	请求搜索酒店,输入关键词搜索,从候选项中选择心仪的酒店	(1)为关键词搜索设置提示词“酒店名称”; (2)酒店选择设置为筛选操作
P2	支付宝	转账(F1)	选择消息图标,请求搜索转账目标,在候选列表中选择目标点击转账按钮,输入转账金额	(1)为转账目标搜索设置提示词“好友手机号”; (2)为目标金额按钮设置提示词“转账金额”
P3	微信	评论(G5)	选择发现图标,打开朋友圈,选择第一个内容进行评论,输入评论内容	为评论内容设置提示词“评论内容”
P4	QQ 邮箱	发邮件(H1)	请求搜索收件人,输入收件人信息进行搜索,选择目标收件人,点击收件人邮箱,输入邮件内容	(1)为收件人信息搜索设置提示词“好友信息”; (2)设置目标收件人选择为筛选操作; (3)为邮件内容输入设置提示词“邮件内容”
P5	Citymapper	设置家庭地址(I1)	请求设置家庭地址,输入家庭地址信息,从搜索结果中选择目标地址	(1)为家庭地址信息输入设置提示词“家庭地址”; (2)设置地址选择为筛选操作

为了确保鲁棒性实验的有效性,选择了极具代表性的应用,并且在不同的类别中选择了与日常生活最为贴近的场景以保证使用场景的多样性。首先,美团是生活中最常用的生活类应用,使用美团确认酒店(A4)是生活中较为常见的场景,属于信息获取类;其次,支付宝是国内移动支付的主流应用,转账(F1)是其中比较核心的使用场景,属于特定任务类;再次,微信是国内活跃用户最多的社交软件,在朋友圈中评论(G5)是日常生活中常见的社交方式,属于发布信息类;然后,得益于 QQ 的庞大用户群体,QQ 邮箱同样具备可观的使用人数,发邮件(H1)是该应用的核心业务,属于发布信息类;最后,Citymapper 是国外较为流行的公共交通应用,导航是目前日常生活中极为常见的用户需求,而设置家庭地址(I1)是导航回家必不可少的使用场景,属于信息管理类。除此之外,这 5 个场景中的一些操作可以进行参数化,同时操作内容变化的可能性较大,符合鲁棒性实验研究的要求。

这类由于无障碍节点信息对象树中不包含目标控件导致的问题,可以通过补充控件的索引标识和完善无障碍节点信息对象树的候选集合的方法尝试修复。其中,补充控件的索引标识可以通过增添控件的文本信息、资源描述符等内容作为辅助标识;完善无障碍节点信息对象树的候选集合可以将非界面的无障碍节点信息对象树纳入检索范围中,并动态替换当前有效的无障碍节点信息对象树。

综上所述,对于适用性的研究问题,本文总结出以下结论:面对移动用户生活中广泛的应用使用需求,除少数对安卓可访问性服务进行了防御处理的应用之外,大部分期望的使用场景都能够使用 RRA 构建出符合需求的应用执行脚本并借助 RRA 正确地执行。

5.3 鲁棒性(RQ2)

为了评估 RRA 的鲁棒性,本文从 RQ1 实验选取 5 个成功的应用使用场景(A4, B3, F1, G5, I1),使用实验设备 A 重新为这些使用场景录制相应的参数化应用执行脚本,测试应用执行脚本在不同参数输入的情况下是否都能正确地执行。首先,本文分别从信息获取、信息管理、发布信息和特定任务这 4 种类型的场景中各选择一个 RQ1 中实验成功的使用场景,即 A4, I1, H1 和 F1;然后从剩余的最常用的应用选取一个 RQ1 中实验成功的使用场景,即 G5;最后将这 5 个场景进行参数化设计,得到如表 5 所列的 5 个实验场景。

对于表 5 中的 5 个场景,本文根据场景的参数化说明录制了相应的应用执行脚本,然后对每个执行脚本都进行了 3 次回放实验且每个场景均能正确地回放。在 3 次回放实验中,第 1 次执行过程与录制过程一致,主要是为了验证应用执行脚本的正确性;第 2 次和第 3 次执行过程会根据场景需求变化,主要是为了对应用执行脚本的鲁棒性进行验证。

对于场景 P1,本文对 A4 步骤中的酒店搜索和酒店选择进行了参数化处理。第 1 次执行时,搜索酒店名为全季酒店,选择列表中的第 2 个选项“全季酒店(上海宝山吴淞国际邮轮码头店)”;第 2 次执行时,搜索酒店名为如家酒店,选择列表中的第 5 个选项“如家酒店(上海宝山万达广场共康路地铁站店)”;第 3 次执行时,搜索酒店名为汉庭酒店,选择列表中的第 9 个选项“汉庭酒店(上海长海医院国和路店)”。

对于场景 P2,本文对 F1 步骤中的好友搜索和转账金额进行了参数化处理。第 1 次执行时,好友手机号为

188xxxx1303, 转账金额为 100; 第 2 次执行时, 好友手机号为 187xxxx3198, 转账金额为 200; 第 3 次执行时, 好友手机号为 158xxxx5797, 转账金额为 300。

对于场景 P3, 本文对 G5 步骤中的评论内容输入进行了参数化处理。第 1 次执行时, 评论内容为第 1 次测试; 第 2 次执行时, 评论内容为第 2 次测试; 第 3 次执行时, 评论内容为第 3 次测试。

对于场景 P4, 本文对 H1 步骤中的收件人搜索和邮件内容输入进行了参数化处理。第 1 次执行时, 好友信息为 1, 选择列表中的第 1 个选项, 邮件内容为第 1 次测试; 第 2 次执行时, 好友信息为 2, 选择列表中的第 3 个选项, 邮件内容为第 2 次测试; 第 3 次执行时, 好友信息为 3, 选择列表中的第 5 个选项, 邮件内容为第 3 次测试。

对于场景 P5, 本文对 I1 步骤中的地址搜索和地址选择进行了参数化处理。第 1 次执行时, 家庭地址为 HKU, 选择列表中的第 1 个选项“香港大学 HKU”; 第 2 次执行时, 家庭地址为皇后大道, 选择列表中的第 3 个选项“皇后大道东”; 第 3 次执行时, 家庭地址为清水湾, 选择列表中的第 5 个选项“香港清水湾道”。

在场景 P1 中, 如图 11 所示, 第 2 次应用执行脚本回放时, 本文让 RRA 播报至酒店搜索结果中界面上可见的最后一个选项(第 5 个选项)时才进行确认, 此次场景的正确执行表明了参数化操作中筛选操作可以正确获取集合容器中的可见内容; 第 3 次应用执行脚本回放时, 本文让 RRA 在播报至酒店搜索结果中的 9 个选项时才进行确认, 此次场景的正确执行表明了参数化操作中筛选操作可以正确获取集合容器中所有候选项的内容。除了场景 P1 之外, 场景 P4 和场景 P5 中也包含了筛选操作, 这些场景的正确执行表明了筛选操作能够在多种应用中被成功识别和执行。



图 11 P1 两次执行的搜索结果

Fig. 11 P1 search results of two executions

在 RQ2 的每个实验场景中都包含参数化的输入操作, 这些场景的正确执行表明了 RRA 能够支持用户根据使用需求实时设置应用执行脚本中参数化的输入操作的输入内容。同时, 在 RQ2 的实验中, 参数化的输入操作使用了多种常见的形式进行参数设置。在场景 P1 中, 酒店名称以中文的形式输入; 在场景 P2 中, 好友手机号和转账金额以数字的形式输入; 在场景 P3 中, 评论内容以数字和中文相结合的形式输

入; 在场景 P4 中, 好友信息以数字的形式输入, 邮件内容以中文与数字相结合的形式输入; 在场景 P5 中, 家庭地址以英文或中文的形式输入。这些场景的正确执行结果表明了 RRA 所提供的语音交互能力能够支持常见的语言形式对输入操作的参数进行赋值。

通过上述实验说明与结果分析, 对于鲁棒性的研究问题, 本文总结出以下结论: RRA 能够支持附带提示词的输入操作和筛选操作这两类参数化操作, 利用这两类参数化操作构建的参数化的应用执行脚本允许用户实时设置参数内容并且保证应用执行脚本的正确执行, 让一个应用执行脚本能够满足用户多个相似的需求。

5.4 可迁移性(RQ3)

为了评估 RRA 的可迁移性, 本文将实验设备 A 上录制成功且回放成功的应用执行脚本迁移到实验设备 B 上, 然后在实验设备 B 上执行这些应用执行脚本, 测试这些场景是否能够被正确地执行。其中, 应用执行脚本的迁移直接通过数据库替换的方式进行, 即将实验设备 B 上 RRA 的数据库替换为实验设备 A 上 RRA 的数据库。

通过分析 RQ1 和 RQ2 中录制成功且回放成功的应用执行脚本, 本文从 RQ1 选取了除 C 类应用执行脚本(C1-C5)之外的所有成功的应用执行脚本, 然后从 RQ2 中选取了全部参数化的应用执行脚本, 最终得到了 RQ3 的实验数据集, 共 34 个应用执行脚本。

RQ3 的实验结果如表 6 所列, 在 34 个应用执行脚本中, 有 27 个正常的应用执行脚本和 5 个参数化的应用执行脚本能够被 RRA 迁移到设备 B 上正常执行, 有 2 个正常的应用执行脚本被 RRA 迁移到设备 B 上执行时发生错误。对 RRA 而言, 正常应用执行脚本的迁移成功率为 93.10%, 参数化应用执行脚本的迁移成功率为 100%, 整体的迁移成功率为 94.12%。另外, 在 29 个正常的应用执行脚本中, 有 26 个场景被 SARA 成功地回放, 有 3 个场景被 SARA 回放失败了, 迁移成功率为 89.66%。相比之下, RRA 的迁移效果更佳。

表 6 RQ3 实验结果

Table 6 RQ3 Experimental results

	A1	A2	A3	A4	B1
R.	✓	✓	✓	✓	✓
S.	✓	✓	✓	✓	✓
	B2	B3	B5	D1	D2
R.	✓	×	✓	✓	✓
S.	✓	×	✓	✓	✓
	D4	D5	E1	E2	E3
R.	✓	✓	✓	✓	✓
S.	✓	✓	✓	✓	✓
	F4	F5	G2	G5	H1
R.	✓	✓	×	✓	✓
S.	✓	✓	×	✓	✓
	H3	H4	I1	I2	I3
R.	✓	✓	✓	✓	✓
S.	✓	✓	✓	×	✓
	I4	I5	J1	J5	P1
R.	✓	✓	✓	✓	✓
S.	✓	✓	✓	✓	—
	P2	P3	P4	P5	
R.	✓	✓	✓	✓	
S.	—	—	—	—	

两个迁移执行失败的场景分别为管理购物车(B3)和打开链接(G2),失败的主要原因是录制时的上下文信息与回放时的上下文信息存在差异,如图 12 所示。其中,管理购物车的应用执行脚本回放失败的原因是,在进行回放实验时,购物车中内容发生了变化,导致部分操作不能获取到目标。打开链接的应用执行脚本回放失败的原因是,在进行回放实验时,当前微信的聊天窗口情况与录制时的微信的聊天窗口情况存在差异,未能找到目标链接进行操作。如果不使用 RQ1 实验中已经录制好的应用执行脚本进行回放,而是在 RQ3 实验过程中先使用设备 A 根据使用场景录制相应的应用执行脚本,那么管理购物车和打开链接的应用执行脚本均能在设备 B 上正确地回放。



图 12 RQ3 中执行失败的场景

Fig. 12 Execution failure scenario in RQ3

这两个由录制阶段和回放阶段的上下文信息的差异导致失败的场景暴露出了目前的录制回放技术比较致命的问题:录制的脚本只能按照一条固定的执行链路行进。虽然本文通过选取出链路上的一些操作进行参数化处理,在一定程度上提高了脚本的灵活程度,但是必须按照这条固定的执行链路行进的问题并未得到太大的改进。

综上所述,对于可迁移性的研究问题,本文总结出以下结论:在现有的实验场景中,通过 RRA 录制得到的执行脚本具有良好的可迁移性,可以根据需要将其放在不同的移动终端上执行。

6 相关工作

本文的工作主要与可访问性增强、录制回放的研究有关。在可访问性增强的研究中,现有的研究大致可以分为可访问性增强方法^[3,16-17,50]和可访问性影响因素^[18,23-25]两种类别。在录制回放的研究中,相关学者主要关注录制回放技术的优化与创新。根据录制回放技术的特征,可以分为插桩技术^[32,34,42-43]和非插桩技术^[33,44-46]两种类别。

6.1 可访问性增强

在可访问性增强方法中,Zhang 等^[16]提出了一种安卓应用的可访问性增强方法,开发者通过使用其提供的标记语言和应用运行时修复工具可以实现应用的可访问性增强;另外,Zhang 等^[17]在 2021 年提出了一种从屏幕像素推测出应用界面上组件可访问性元数据的方法,并将推测出的元数据提供给 IOS VoiceOver 以验证其方法对应用可访问性的增强。这两种方法本质上都是通过为应用提供可访问性标签,来完成应用可访问性的增强;而本文提出的方法利用录制回放的

方式,通过简化应用交互的过程并利用语音交互能力提供对话操作应用的交互方式,从而完成对应用可访问性的增强。Oney 等^[33]实现了一个网页应用的远程协作工具 Arboretum,可以让用户无缝地将浏览会话的控制部分转交给远程用户,通过“提议和接受/拒绝”机制保持对界面的控制,解决了视力有缺陷的人访问网页困难的问题。Lukić 等^[50]基于 IOS 提供的可访问性服务提出了一个远程控制工具,能够进行用户场景的录制回放、生成 Appium 的测试用例与应用的状态变化图。但是,这个工具面向的是 IOS 系统且需要两个设备同时在线才能进行远程控制,如果协作者当前无法在线,那么视力有缺陷的人访问网页的困难依然存在。与这两个研究工作相比,本文提出的方法利用录制回放的方式增强移动应用的可访问性,不同于远程协作的方式,录制回放的方式不要求双方同时在线,并且这种方式只需要进行一次录制即可多次回放。

在可访问性影响因素研究中,Alshayban 等^[18]分析了 1 000 多个安卓应用中可访问性问题的普遍性,调查了安卓应用可访问性问题出现的原因,分析了可访问性问题对应用评分和评论的影响;Fok 等^[23]编写爬虫程序构建包含 312 个应用程序 16 个月内的快照数据的数据集,利用该数据集纵向分析了图像元素可访问性影响因素;Vendome 等^[24]通过对程序设计领域的问答网站 Stack Overflow 上与安卓可访问性有关的帖子进行调查,确定了开发人员提高应用可访问性会遇到的问题并总结了经验教训;Eler 等^[25]提出了一个 MATE 工具去检测和定位安卓应用的可访问性问题。然而,这些现有的研究工作专注于分析与检测移动应用可访问性问题与影响因素,从开发者角度去修复移动应用的可访问性问题,增强移动应用的可访问性。上述这些可访问性影响因素研究的目的和本文不同,它们旨在帮助开发者解决与定位应用开发过程中的可访问性问题,而本文旨在解决现有移动应用的可访问性需求,增强现有移动应用的可访问性。除此之外,这些可访问性影响因素研究工作主要通过分析影响因素为应用开发者提供可访问性问题的修复建议与指导,间接增强应用的可访问性;而本文的方法是通过缩短交互路径、简化应用的使用过程并结合语音交互增加应用的交互手段,扩大应用可服务的用户群体,直接增强应用的可访问性。

6.2 录制回放技术

基于插桩技术的录制回放可以分为应用插桩和操作系统插桩两类。因为操作系统插桩在实践中更为耗时且修改后的操作系统可能与设备不兼容,所以大部分的研究工作把重点放在应用插桩上。在应用插桩的研究中,Sahin 等^[32]利用插桩技术实现了一个能够对安卓应用用户交互、网络事件、影响应用行为的随机数多种输入源进行录制并且能够跨设备回放的工具 RANDR。RANDR 利用反汇编技术修改目标应用,利用现有的插桩框架对 ART 中的 Java APIs 和本地依赖库进行修改。Guo 等^[34]采用动态插桩技术实现了基于组件进行录制且能够跨设备回放的桌面应用程序 SARA。但 RANDR 和 SARA 都利用了动态插桩技术,是一种侵入式的录制回放方法。Fazzini 等^[42]基于 Espresso^[38]框架实现了一个可以跨设备的录制回放工具 Barista。由于 Espresso 是对源码进行插桩,从而达到监控程序的执行及其生命周期的目的,

因此 Barista 只适用于开源的应用。在操作系统插桩的研究中, Qin 等^[43]实现了由一个运行在安卓移动设备上的客户端和一个运行在服务端的目标应用组成的客户端-服务器系统 MobiPlay。在 MobiPlay 中, 客户端实时显示服务器上目标应用的界面并将用户的输入数据转发给服务器, 服务器接收输入数据之后记录有效的数据, 操作目标应用。然而, MobiPlay 是基于坐标进行的录制回放, 因此无法进行跨设备的回放。

在基于非插桩技术的录制回放研究中, 主要是使用设备的 ROOT 权限进行录制回放。RERAN^[33]使用 ROOT 权限去执行 ADB 命令 `getevent` 和 `sendevent`。利用 `getevent` 从 `/dev/input/event *` 日志文件中记录用户的操作事件, 然后利用 `sendevent` 回放这些操作。但是, 这些操作事件与设备的分辨率、大小相关, 因此无法进行跨设备的回放。Bernal-Cárdenas 等^[44]提出了一种基于图像处理的数据回放方法, 利用 Faster R-CNN 找到视频帧中出现的触摸指示器的位置, 利用自定义的 Opacity CNN 识别触摸指示器的透明度, 然后根据触摸指示器的透明度进行分类, 编织回放脚本, 利用 `sendevent` 命令进行回放。但这种方法编织的回放脚本与设备的分辨率、大小相关, 无法进行跨设备的回放。Mosaic^[45]与 RERAN 类似, 使用 ROOT 权限去执行 ADB 命令 `getevent` 和 `sendevent`。为了实现跨设备的回放, Mosaic 使用一系列转置、缩放和归一化来映射不同设备之间的交互坐标。但是这要求目标元素能在特定方向上根据屏幕尺寸进行线性缩放。Appetizer 提供了一个能够录制屏幕触碰事件并跨设备回放的命令行工具 `replaykit`^[46]。但这个工具要求录制设备和回放设备的屏幕横宽比一致。与上文的数据回放研究相比, 本文方法能够在非侵入式的前提下录制脚本且跨设备回放脚本, 并在录制回放的基础上赋予了应用语音交互的能力。同时, 本文方法不需要获取设备的 ROOT 权限, 只需要用户开启可访问性权限即可。

结束语 在日常生活中, 使用复杂的移动应用会产生不少重复的交互操作, 使用录制回放技术能够帮助用户模拟这些重复操作, 缩短用户使用移动应用的交互路径, 提高移动应用的可用性。本文以录制回放为基础, 提出了一种基于录制回放的移动应用可用性增强方法。该方法利用安卓的可用性服务获取移动应用使用过程中的交互信息, 利用应用执行脚本的录制方法记录交互信息, 利用应用执行脚本的回放方法提取交互信息并执行交互操作。其中, 设计了基于无障碍节点信息对象的目标控件检索方法以保证应用执行脚本的可迁移性; 设计了基于语音交互的应用执行脚本参数化方法, 使得执行脚本能够满足用户的不同需求。

虽然本文方法在应用执行脚本的录制回放中取得了不错的实际效果, 为老年人和视觉障碍人士提供了一种按需使用移动应用的可能性, 但是本文的实验研究表明该方法仍存在一定的局限性。因此, 未来的一个工作方向可以针对现有的局限性进行, 即在无障碍事件反馈不充足的情况下确保录制回放的的有效性。同时, 未来可以尝试支持更多有价值的参数化操作, 增加应用执行脚本的可能性。另外, 本文方法具有较好的可迁移性, 未来可以尝试基于本文方法构建应用执行脚本的共享社区, 通过群智群力的

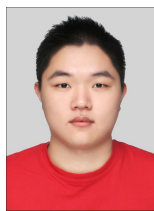
方式丰富移动应用的应用执行脚本。

参 考 文 献

- [1] IWARSSON S, STÅHL A. Accessibility, usability and universal design—positioning and definition of concepts describing person-environment relationships [J]. *Disability and rehabilitation*, 2003, 25(2): 57-66.
- [2] BIGHAM J P, LIN I, SAVAGE S. The Effects of “Not Knowing What You Don’t Know” on Web Accessibility for Blind Web Users[C]// *Proceedings of the 19th International ACM SIGACCESS Conference on Computers and Accessibility (ASSETS)*. New York, NY: ACM, 2017: 101-109.
- [3] ONEY S, LUNDGARD A, KROSNICK R, et al. Arboretum and arbority: Improving web accessibility through a shared browsing architecture[C]// *Proceedings of the 31st Annual ACM Symposium on User Interface Software and Technology (UIST)*. New York, NY: ACM, 2018: 937-949.
- [4] LAZAR J, ALLEN A, KLEINMAN J, et al. What frustrates screen reader users on the web: A study of 100 blind users[J]. *International Journal of human-computer interaction*, 2007, 22(3): 247-269.
- [5] YU J E, CHATTOPADHYAY D. “Maps are hard for me”: identifying how older adults struggle with mobile maps[C]// *Proceedings of the 22nd International ACM SIGACCESS Conference on Computers and Accessibility (ASSETS)*. New York, NY: ACM, 2020: 1-8.
- [6] ABOU-ZAHRA S, HENRY S L. Exploring web accessibility solutions in developing regions as innovations for the benefit of all [C]// *Proceedings of the 2010 International Cross Disciplinary Conference on Web Accessibility (W4A)*. New York, NY: ACM, 2010: 1-4.
- [7] HENRY S L, ABOU-ZAHRA S, BREWER J. The role of accessibility in a universal web[C]// *Proceedings of the 11th Web for all Conference (W4A)*. New York, NY: ACM, 2014: 1-4.
- [8] W3C Web Accessibility Initiative (WAI). Introduction to web accessibility [EB/OL]. (2022-03-31) [2022-11-25]. <https://www.w3.org/WAI/fundamentals/accessibility-intro/>.
- [9] Android Accessibility. Android accessibility overview [EB/OL]. [2022-11-25]. <https://support.google.com/accessibility/android/answer/6006564>.
- [10] Android Developers. Android Accessibility Developer Guidelines [EB/OL]. (2022-10-24) [2022-11-25]. <https://developer.android.com/guide/topics/ui/accessibility>.
- [11] MILNE S, DICKINSON A, CARMICHAEL A, et al. Are guidelines enough? An introduction to designing Web sites accessible to older people[J]. *IBM Systems Journal*, 2005, 44(3): 557-571.
- [12] MANKOFF J, FAIT H, TRAN T. Is your web page accessible? A comparative study of methods for assessing web page accessibility for the blind[C]// *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI)*. New York, NY: ACM, 2005: 41-50.
- [13] CLEGG-VINELL R, BAILEY C, GKATZIDOU V. Investigating the appropriateness and relevance of mobile web accessibility guidelines[C]// *Proceedings of the 11th Web for All Conference*.

- rence(W4A). New York, NY: ACM, 2014: 1-4.
- [14] ROSS A S, ZHANG X, FOGARTY J, et al. Epidemiology as a framework for large-scale mobile application accessibility assessment[C]//Proceedings of the 19th international ACM SIGACCESS Conference on Computers and Accessibility (ASSETS). New York, NY: ACM, 2017: 2-11.
- [15] YAN S, RAMACHANDRAN P G. The current status of accessibility in mobile apps [J]. ACM Transactions on Accessible Computing (TACCESS), 2019, 12(1): 1-31.
- [16] ZHANG X, ROSS A S, FOGARTY J. Robust annotation of mobile application interfaces in methods for accessibility repair and enhancement[C]//Proceedings of the 31st Annual ACM Symposium on User Interface Software and Technology (UIST). New York, NY: ACM, 2018: 609-621.
- [17] ZHANG X, DE GREEF L, SWEARNGIN A, et al. Screen recognition: Creating accessibility metadata for mobile applications from pixels[C]//Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems (CHI). New York, NY: ACM, 2021: 1-15.
- [18] ALSHAYBAN A, AHMED I, MALEK S. Accessibility issues in android apps: State of affairs, sentiments, and ways forward [C]//Proceedings of the ACM/IEEE 42nd International Conference on Software Engineering (ICSE). New York, NY: ACM, 2020: 1323-1334.
- [19] ANUPAM V, FREIRE J, KUMAR B, et al. Automating Web navigation with the WebVCR [J]. Computer Networks, 2000, 33(1/2/3/4/5/6): 503-517.
- [20] ALLEN J, CHAMBERS N, FERGUSON G, et al. Plow: A collaborative task learning agent[C]//Proceedings of the 22nd National Conference on Artificial Intelligence (AAAI). Menlo Park, CA: AAAI Press, 2007: 1514-1519.
- [21] LAU T, CERRUTI J, MANZATO G, et al. A conversational interface to web automation[C]//Proceedings of the 23rd Annual ACM Symposium on User Interface Software and Technology (UIST). New York, NY: ACM, 2010: 229-238.
- [22] LI I, NICHOLS J, LAU T, et al. Here's what I did: Sharing and reusing web activity with ActionShot[C]//Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI). New York, NY: ACM, 2010: 723-732.
- [23] FOK R, ZHONG M, ROSS A S, et al. A Large-Scale Longitudinal Analysis of Missing Label Accessibility Failures in Android Apps[C]//Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems (CHI). New York, NY: ACM, 2022: 1-16.
- [24] VENDOME C, SOLANO D, LINÁN S, et al. Can everyone use my app? an empirical study on accessibility in android apps [C]//Proceedings of the 2019 IEEE International Conference on Software Maintenance and Evolution (ICSME). Piscataway, NJ: IEEE Press, 2019: 41-52.
- [25] ELER M M, ROJAS J M, GE Y, et al. Automated accessibility testing of mobile apps[C]//Proceedings of the 2018 IEEE 11th International Conference on Software Testing, Verification and Validation (ICST). Piscataway, NJ: IEEE Press, 2018: 116-126.
- [26] MORAN K, LINARES-VÁSQUEZ M, BERNAL-CÁRDENAS C, et al. Fusion: A tool for facilitating and augmenting android bug reporting[C]//Proceedings of the 38th International Conference on Software Engineering Companion (ICSE). New York, NY: ACM, 2016: 609-612.
- [27] KONG P, LI L, GAO J, et al. Automated testing of android apps: A systematic literature review [J]. IEEE Transactions on Reliability, 2018, 68(1): 45-66.
- [28] BEKRAR S, BEKRAR C, GROZ R, et al. Finding software vulnerabilities by smart fuzzing [C]//Proceedings of the 2011 Fourth IEEE International Conference on Software Testing, Verification and Validation (ICST). Piscataway, NJ: IEEE Press, 2011: 427-430.
- [29] LINARES-VÁSQUEZ M, BERNAL-CÁRDENAS C, MORAN K, et al. How do developers test android applications? [C]//Proceedings of the 2017 IEEE International Conference on Software Maintenance and Evolution (ICSME). Piscataway, NJ: IEEE Press, 2017: 613-622.
- [30] JOORABCHI M E, MESBAH A, KRUCHTEN P. Real challenges in mobile app development [C]//Proceedings of the 2013 ACM/IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM). Piscataway, NJ: IEEE Press, 2013: 15-24.
- [31] LAM W, WU Z, LI D, et al. Record and replay for android: Are we there yet in industrial cases? [C]//Proceedings of the 2017 11th Joint Meeting on Foundations of Software Engineering (ESEC/FSE). New York, NY: ACM, 2017: 854-859.
- [32] SAHIN O, ALIYEVA A, MATHAVAN H, et al. Randr: Record and replay for android applications via targeted runtime instrumentation [C]//Proceedings of the 34th IEEE/ACM International Conference on Automated Software Engineering (ASE). Piscataway, NJ: IEEE Press, 2019: 128-138.
- [33] GOMEZ L, NEAMTIU I, AZIM T, et al. Reran: Timing- and touch-sensitive record and replay for android [C]//Proceedings of the 2013 International Conference on Software Engineering (ICSE). Piscataway, NJ: IEEE Press, 2013: 72-81.
- [34] GUO J, LI S, LOU J G, et al. Sara: self-replay augmented record and replay for Android in industrial cases [C]//Proceedings of the 28th ACM Sigsoft International Symposium on Software Testing and Analysis (ISSTA). New York, NY: ACM, 2019: 90-100.
- [35] Google Play. HiroMacro Auto-Touch Macro [EB/OL]. (2017-08-20) [2023-01-28]. <https://play.google.com/store/apps/details?id=com.prohiro.macro&pli=1>.
- [36] Uptodown. RepetiTouch Free [EB/OL]. (2021-02-24) [2023-01-30]. <https://repetitouch-free.cn.uptodown.com/android>.
- [37] Github. culebra [EB/OL]. (2018-06-01) [2023-01-30]. <https://github.com/dtmilano/AndroidViewClient/wiki/culebra>.
- [38] Android Developers. Create UI tests with Espresso Test Recorder [EB/OL]. (2023-03-08) [2023-03-10]. <https://developer.android.com/studio/test/other-testing-tools/espresso-test-recorder>.
- [39] Android Developers. monkeyrunner [EB/OL]. (2022-12-09) [2023-01-30]. <https://developer.android.com/studio/test/monkeyrunner/index.html>.
- [40] Ranorex. Mobile Testing Tools [EB/OL]. [2023-01-30]. <https://www.ranorex.com/mobile-automation-testing>.

- [41] HU Y, AZIM T, NEAMTIU I. Versatile yet lightweight record-and-replay for android[C]//Proceedings of the 2015 ACM SIGPLAN International Conference on Object-Oriented Programming, Systems, Languages, and Applications (OOPSLA). New York, NY: ACM, 2015: 349-366.
- [42] FAZZINI M, FREITAS E N D A, CHOUDHARY S R, et al. Barista: A technique for recording, encoding, and running platform independent android tests[C]//Proceedings of the 2017 IEEE International Conference on Software Testing, Verification and Validation (ICST). Piscataway, NJ: IEEE Press, 2017: 149-160.
- [43] QIN Z, TANG Y, NOVAK E, et al. Mobiplay: A remote execution based record-and-replay tool for mobile applications[C]//Proceedings of the 38th International Conference on Software Engineering (ICSE). New York, NY: ACM, 2016: 571-582.
- [44] BERNAL-CÁRDENAS C, COOPER N, MORAN K, et al. Translating video recordings of mobile app usages into replayable scenarios[C]//Proceedings of the ACM/IEEE 42nd International Conference on Software Engineering (ICSE). New York, NY: ACM, 2020: 309-321.
- [45] HALPERN M, ZHU Y, PERI R, et al. Mosaic: cross-platform user-interaction record and replay for the fragmented android ecosystem[C]//Proceedings of the 2015 IEEE International Symposium on Performance Analysis of Systems and Software (ISPASS). Piscataway, NJ: IEEE Press, 2015: 215-224.
- [46] Github. replaykit [EB/OL]. (2018-06-16)[2022-11-27]. <https://github.com/appetizerio/replaykit>.
- [47] TALEBPOUR S, ZHAO Y, DOJCILOVIĆ L, et al. UI test migration across mobile platforms[C]//Proceedings of the 36th IEEE/ACM International Conference on Automated Software Engineering (ASE). Piscataway, NJ: IEEE Press, 2021: 756-767.
- [48] QIN X, ZHONG H, WANG X. Testmig: Migrating gui test cases from ios to android[C]//Proceedings of the 28th ACM SIGSOFT International Symposium on Software Testing and Analysis (ISSTA). New York, NY: ACM, 2019: 284-295.
- [49] YU S, FANG C, YUN Y, et al. Layout and image recognition driving cross-platform automated mobile testing[C]//Proceedings of the 43rd International Conference on Software Engineering (ICSE). Piscataway, NJ: IEEE Press, 2021: 1561-1571.
- [50] LUKIĆ N, TALEBPOUR S, MEDVIDOVIĆ N. Remote control of ios devices via accessibility features[C]//Proceedings of the 2020 ACM Workshop on Forming an Ecosystem Around Software Transformation (FEAST). New York, NY: ACM, 2020: 35-40.
- [51] AOSP. Android 4.0 Compatibility Definition Document [EB/OL]. (2013-04-21)[2023-02-28]. <https://developer.android.com/guide/topics/ui/accessibility/service>.



LI Xiangmin, born in 1998, postgraduate. His main research interests is intelligent software engineering and systems.



SHEN Liwei, born in 1982, Ph.D, associate professor. His main research interests include cyber physical social, ubiquitous computing and cloud computing.

(责任编辑:杨雪敏)