通过概率形式化建模揭示并解读交互风格

Oana Andrei、Muffy Calder、Matthew Chalmers 和 Alistair Morrison

格拉斯哥大学计算机科学学院，英国
firstname.lastname@glasgow.ac.uk

arXiv:2305.01656v1 [cs.HC] 1 May 2023

摘要。 我们介绍了一项研究，该研究采用了新的计算方法，基于用于推断混合隐马尔可夫模型的机器学习与概率模型检验的创新性结合，以揭示移动应用中的交互风格。随后，我们利用这些风格为重新设计提供依据，并实施和部署了重新设计方案，之后再次采用相同的方法进行分析。数据集为记录的用户轨迹，收集自每个版本为期六个月的两次部署，涉及数千名用户，并按照不同的时间区间进行分段。这些方法不预先假设任务或参与度测量等绝对指标，而是通过对簇进行无监督推断，并结合概率时序逻辑分析来揭示交互风格。对于两个版本，我们都发现，用户在使用的第一天、第一周或第一个月所采用的交互风格，与第二个月和第三个月所采用的风格之间存在明显差异，而这一结果超出了我们的预期。

1 引言

菜单驱动界面的设计通常旨在满足人们所感知或预期的终端用户需求和交互风格，但在实践中，用户可能会采用多种交互风格，其中一些是设计人员未曾预料到的。这可能由多种因素造成，包括软件挪用、 [18,19,44],开始使用软件至今的时间（例如，第一周或使用六个月后）、对特定用途的意图（例如，快速使用或长时间、深入地使用），或心理状态（例如，心不在焉或专心致志）等。因此，交互风格可能因用户而异， [52] 也可能随时间推移而变化：同一用户在不同交互会话之内及会话之间，都可能采用不同的风格。因此，设计人员可能希望根据用户长期以来使用系统的方式重新设计界面。

研究交互的现有工具包括访谈、出声思维法和直接观察等定性方法，这些方法有助于揭示用户的行为和偏好，但要在庞大的用户总体中实施这些程序，成本很高。若将样本量限制在较小范围内，可能会遗漏某些活动形式，也可能产生文化偏差，例如，样本仅来自设计人员所在地区的用户。定量方法可以涵盖更多用户——甚至可能是所有 用户——但大多数现有方法都依赖预先设定的建模假设，例如假定用户会执行某些任务，或采用相对简单的指标。例如，这些方法可能关注应用使用时长和访问的屏幕等基本特征的发生统计，或采用基于任务的测量 [21]。

我们介绍了一项关于移动应用以用户为中心的重新设计的研究，该研究采用了一种研究交互的新型定量方法。数据是记录的用户交互轨迹，从应用用户总体中提取（在我们的案例中为所有用户）。我们根据时间区间将轨迹分为若干数据集：使用的第1天、第1周、第1个月、第2个月和第3个月，从而观察交互风格是否与参与时长相关。我们没有预先假定任务，而是记录了所有会改变用户视图的交互。基于这些数据集，我们使用机器学习（ML）无监督聚类方法推断计算模型。我们方法的一项创新在于，使用概率时序逻辑性质询问所推断的模型，然后通过归纳式编码解释结果，以评估设计对我们所发现的交互风格的支持程度。这些结果为界面的重新设计提供了依据，随后我们以相同的方式部署并研究了重新设计后的界面。

该研究包含层级菜单界面的两轮设计迭代，适用于 AppTracker [37]，它允许用户跟踪其设备的使用情况，也可被视为一种用于个人信息学 [51,32]的工具。每种设计（我们分别称为 AppTracker 1 和 AppTracker2）都至少部署了6个月，涉及数千名用户。本文介绍的工作代表了一个罕见的、长期的多年合作项目，参与者来自人机交互评估、移动应用设计、机器学习和模型检验等领域。

本文的组织结构如下。下一节概述 AppTracker 和研究设计。第 3 节介绍主要结果，包括在 AppTracker1 中发现的交互风格、为 AppTracker2 重新设计的界面、在 AppTracker2 中发现的交互风格，以及对在 AppTracker1 和 AppTracker2 中发现的交互风格进行比较。第 4 节探讨我们的研究发现以及机器学习的作用，相关工作见第 5 节，结论和未来工作见第 6 节。

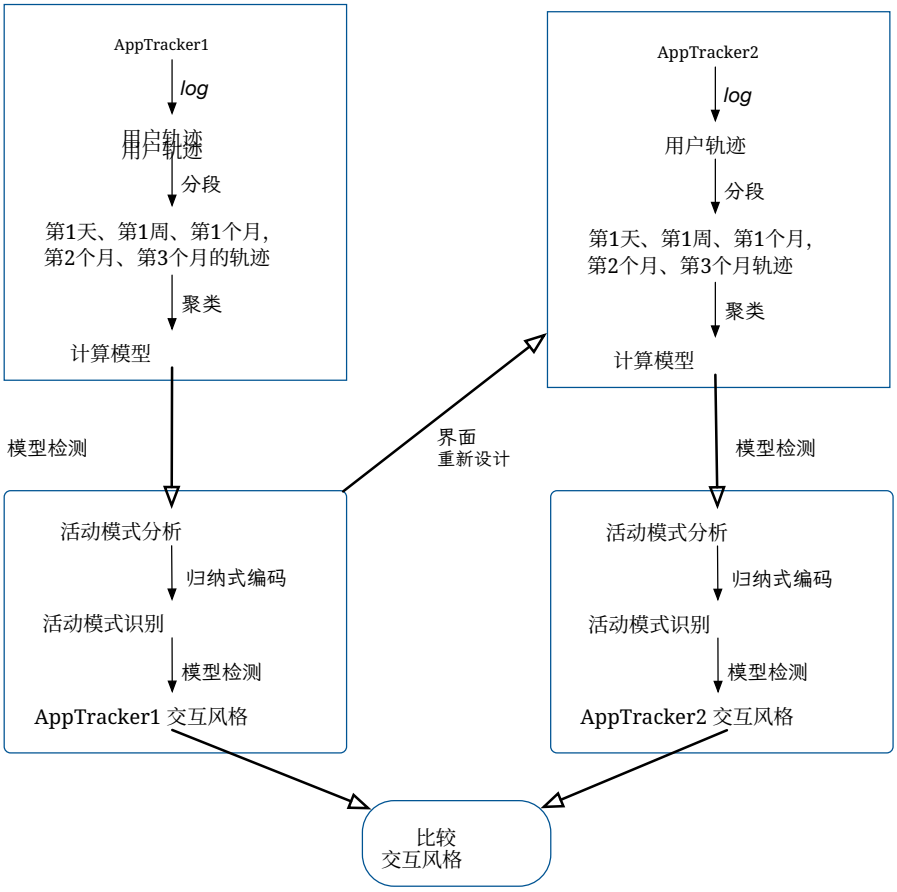


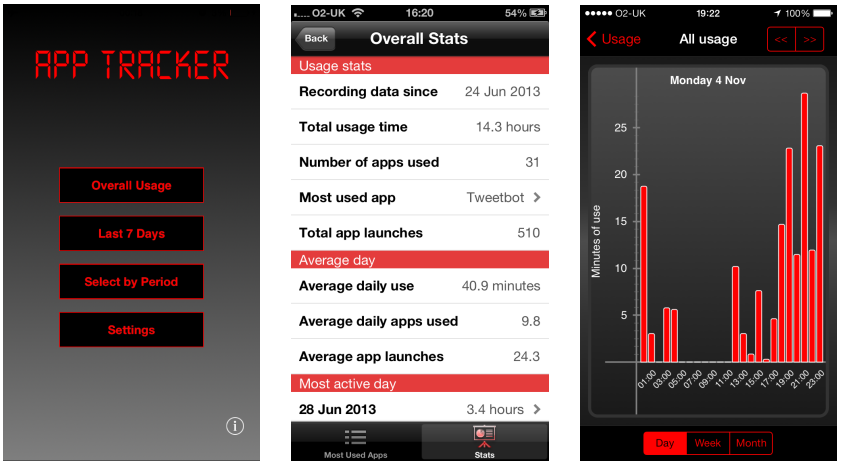
图 1. 研究设计

2 研究设计

AppTracker 让用户能够跟踪并查看其 iPhone 或 iPad 设备上所有移动应用的使用情况。它在后台运行，监控设备上安装的所有应用的打开和关闭情况，以及设备每次锁定或解锁的情况。它会收集这些数据，并以一系列图表和统计数据的形式呈现，让用户了解自己的使用行为，例如每天使用设备的时长，或一段时间内最常使用的应用。用户界面基于分层菜单，用户可通过菜单结构导航，以查看图表和汇总统计数据。AppTracker 会生成两种不同类型的数据供研究：i）用户对设备的使用情况，包括每次启动应用的记录；ii）用户在 *AppTracker* 应用本身中的交互，例如按钮点击和屏幕切换。前一种数据已在先前的研究 [37], 中得到分析；本研究基于后一种数据。

2.1 概述

图 1 概述了本研究的流程。我们记录了大量用户群体的 AppTracker1 用户交互，将用户轨迹分段为时间序列数据集，并使用机器学习聚类算法从这些数据集中推断混合马尔可夫模型。每个马尔可夫模型的一个关键概念是推断出的活动模式及其之间的转移概率；这些内容共同概括了一组记录用户轨迹所共有的典型观测时间行为。每个活动模式都是一个离散时间马尔可夫链，其中观测变量标记 AppTracker 状态，而每个模式对应于涵盖全部交互行为的混合马尔可夫模型中的一个隐状态。我们使用一组（参数化的）概率时序逻辑性质和模型检验 [5] 对模型的活动模式进行了分析。我们使用归纳式编码 [46] 对结果进行分类，例如到达某一状态所需的步数、会话时长、主要状态，然后对结果进行解释，以识别这些活动模式。我们使用另一组时序逻辑性质，生成活动模式之间的长期似然和转移概率；所有这些结果共同构成了对交互风格的描述。我们考察了 AppTracker1 对用户交互风格的支持程度，并在重新设计中解决其不足，将成果用于下一版本 AppTracker2，随后该版本得以部署。我们基于 AppTracker2 用户轨迹再次进行分析，最后通过比较 AppTracker1 与 AppTracker2 中的交互风格结束了本研究。



(a) 主菜单 (b) 统计 (c) 每日设备使用情况

图2. AppTracker1 的屏幕截图：(a) 主菜单视图对应状态“主界面”；(b) 统计视图对应状态“统计”；(c) 所选时段为11月4日星期一时，每日设备使用情况视图对应状态“总体图表”。

2.2 时间序列数据

每条记录的用户轨迹都是一系列事件标签。每条轨迹由许多用户交互会话组成，这些会话在应用程序启动或被调至前台时开始（以事件 startS 表示），并在应用程序关闭或置于后台时结束（以事件 stopS 表示）。我们按时间间隔 $[t_1, t_2]$ 对用户轨迹进行分段，使得该分段的第一个会话发生在时间戳 t_1 或之后，而片段的最后一个会话发生在时间戳 t_2之前；整条轨迹可能延伸至时间戳 t_2之后。

2.3 AppTracker1 界面

启动 AppTracker1 后，主菜单界面提供四个主要选项（图 2(a)）；从上到下依次为：

– 总体使用情况汇总自安装 AppTracker1 以来记录的所有数据，并打开“总体使用情况”和“统计”视图（图 2(b)）。– 最近7天 打开“最近7天”视图，并显示用户过去 7 天内使用最多的 5 个应用的活动图表。– 按时间段选择 打开“选择时段”视图，并显示所选时间段的统计信息，例如自星期五以来使用最多的应用、过去一个月每天使用 Facebook 的时长，以及星期一的每小时设备使用情况（图 2(c)）。– 设置 允许用户启动和停止跟踪器，或重置记录的数据。

共有 16 个用户触发事件可在不同视图之间切换（事件名称即切换后的视图名称），参见图 3 中的状态图。状态分为摘要状态（用于查看摘要或总体使用数据）、具体状态（用于查看深入的特定数据）、标记会话开始和结束的会话相关状态，以及其余状态。请注意，根据使用情境，状态“最近7天”既可以是摘要状态，也可以是具体状态。

记录的交互数据使用 SGLog 框架 [25]存储在 MySQL 数据库中，并通过 JavaScript 处理，以 JSON 格式获取用户轨迹。AppTracker1 于 2013 年 8 月首次发布，下载次数超过 35,000 次。我们的数据集取自 2013 年和 2014 年的 322 条用户轨迹样本。所有轨迹中的最大会话数量为 129，最小值限定为 5。

2.4 计算方法

我们使用机器学习聚类方法来推断混合马尔可夫模型，该模型首次在 [1], 中基于一阶自回归隐马尔可夫模型（AR-HMM） [38]进行了定义。混合模型允许交错变化，因此我们能够对可能在会话内部和会话之间切换交互风格的用户进行建模。为确保内容完整，我们在此给出一些定义。

离散时间马尔可夫链（DTMC）是一个元组 $\mathcal{D} = (S,\ s_0,\ \mathrm{P}\ \mathcal{L})$，其中： S是状态集合； $s_0 \in S$是初始状态；P： $S \times S \to [0,1]$ 是转移概率函数，使得对于所有状态 $s \in S$，我们都有

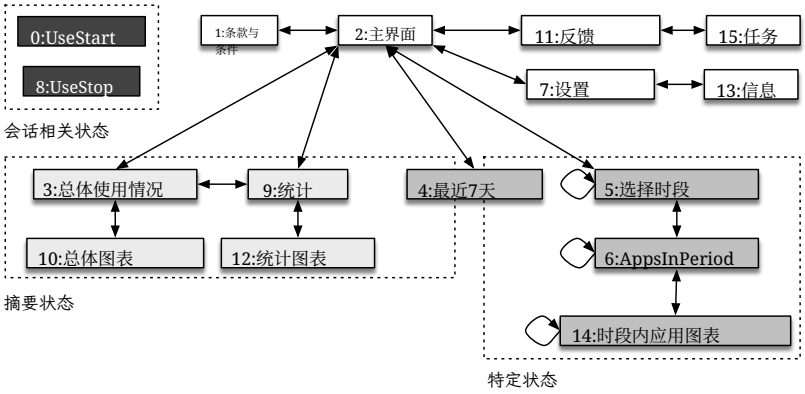


图 3。 AppTracker1 状态图。菜单的垂直布局对应于虚线框中状态从左到右的遍历。

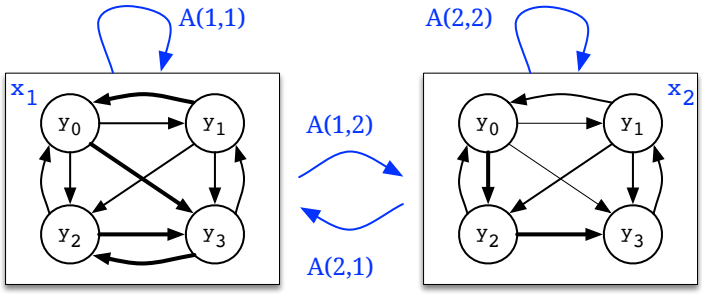


图 4。 包含隐状态 x_1 和 x_2的 GPAM(2) 的图示表示。两个活动模式分别是每个框中的离散时间马尔可夫链；共有四个观测状态 y_0、 y_1、 y_2、 y_3。转移概率通过转移线的粗细表示。

$\sum_{s' \in S} \mathrm{P}(s,\ s') = 1$；并且 $\mathcal{L}: S \to 2^{\mathcal{A}}$ 是一个标记函数，将每个状态 s 映射到集合 S 中的一组有效原子命题，这些命题来自集合 $\mathcal{A}$。离散时间马尔可夫链的一条路径（或执行）是一个非空序列 $s_0 s_1 s_2 \ldots$，其中 $s_i \in S$ 且对于所有 $i \geq 0$，$\mathrm{P}(s_i,\ s_{i+1}) > 0$。转移也称为时间步。

一个一阶自回归隐马尔可夫模型 （AR-HMM） [38] 是一个元组（$\mathcal{X}$、 $\mathcal{Y}$、 π、 A、 B），其中： $\mathcal{X}$是隐状态（或潜在状态） $\mathcal{X} = \{1, \ldots, K\}$的集合； $\mathcal{Y}$是由隐状态生成的观测状态集合；π： $\mathcal{X} \to [0,1]$ 是一个初始分布，其中 $\sum_{x \in \mathcal{X}} \pi(x) = 1$； A： $\mathcal{X} \times \mathcal{X} \to [0,1]$ 是转移概率矩阵，使得对于所有 $x \in \mathcal{X}$，都有 $\sum_{x' \in \mathcal{X}} A(x,\ x') = 1$； B： $\mathcal{X} \times \mathcal{Y} \times \mathcal{Y} \to [0,1]$ 是观测概率矩阵，使得对于所有 $x \in \mathcal{X}$和 $y \in \mathcal{Y}$，都有 $\sum_{y' \in \mathcal{Y}} B(x,\ y,\ y') = 1$。

现在，设 $\mathcal{P}$ 是一个包含 M 条用户轨迹的总体，这些轨迹包含 n 种不同类型的事件标签； $\mathcal{A}$ 是 $\mathcal{P}$ 中所有事件的标签集合， K 是一个正整数。对于用户轨迹总体 $\mathcal{P}$，含有 K 个成分的广义总体混合模型，或称 GPAM(K)，是一个元组 $(\mathcal{X}, \mathcal{Y}, \pi, A, B, \mathcal{L})$，其中 $(\mathcal{X}, \mathcal{Y}, \pi, A, B)$ 是一个具有 $|\mathcal{X}| = K$ 和 $|\mathcal{Y}| = n$ 的 AR-HMM，且 $\mathcal{L}: \mathcal{Y} \to \mathcal{A}$ 是一个标记函数，为 $\mathcal{Y}$ 中每个观测到的状态映射一个唯一的事件名称。图 4 给出了 GPAM(2) 的示意图。对于任意 GPAM $(\mathcal{X}, \mathcal{Y}, \pi, A, B, \mathcal{L})$ 和一个隐状态 $x \in \mathcal{X}$，元组 $(\mathcal{Y}, \mathcal{L}^{-1}(\text{startS})\ B(x)\ \mathcal{L})$ 是一个称为活动模式的离散时间马尔可夫链（DTMC）。

对于每个包含用户轨迹的输入数据集 $\mathcal{P}$，我们为每条轨迹 α 计算转移发生矩阵 $n \times n$，使得矩阵位置 (i, j) 的值表示子序列 $\alpha_i \alpha_j$ （即两个相邻的事件标签）在 α 中出现的次数。所得的转移发生矩阵是机器学习的输入数据；我们采用 Baum—Welch 聚类算法 [49], ，该算法使用局部非线性优化的期望最大化（EM）算法 [17], 以求得观察每条轨迹的最大似然参数，并在对数似然存在多个局部极大值时重新启动算法。

我们使用概率模型检验器 PRISM [31] 分析以 rPCTL 表示的时序属性。rPCTL 是带奖励的概率计算树逻辑 PCTL* [31,5] 的扩展；概述请参见附录 B。

我们定义并使用的参数化属性集合见表 1。这些时序属性描述活动模式内的行为，或涉及多个活动模式的行为。后一类属性是 StateToPattern 和 LongRunPattern，我们仅用它们对 GPAM 进行模型检验。其余属性则用于作为离散时间马尔可夫链的活动模式。首先，我们对所有状态，针对 K 的不同递增值，对时序属性 VisitProbInit、StepCountInit、VisitCountInit、SessionLength 和 SessionCount 进行模型检验。视情况而定，我们还会使用属性 VisitProbBtw 和 StepCountBtw，条件是

表 1. 基于奖励的概率计算树逻辑（rPCTL）属性由以下参数确定： N 表示时间步数的正整数值； j， j_1， j_2 表示观测状态标识符的正整数值； i， i_1， i_2 介于 1 和 K 之间的整数值，表示隐藏状态标识符（活动模式）； $p \in [0,1]$ 一个概率。

Name	公式及非正式描述
VisitProbInit	$P_{=?}[\text{true } U^{\leq N}(y=j)]$： 从初始状态开始，在 N 步内到达 j 的概率
StepCountInit	$R_{\texttt{rSteps}=?}[F\ (y=j)]$： 从初始状态开始到达 j 所需的预期步数
VisitCountInit	$R_{\texttt{rStatej}=?}[C^{\leq N}]$： 在 N 步内访问 j 的预期访问次数
SessionLength	$R_{\texttt{rSteps}=?}[F(y=\text{stopS})]$： 直到会话结束的预期步数。
会话次数	$R_{\texttt{rStatestopS}=?}[C^{\leq N}]$： 在 N 步内的预期会话次数
VisitProbBtw	*filter*(状态，$P_{=?}[(\neg\text{stopS})\ U^{\leq N}(y=j_2)], (y=j_1)$)：从 j_1 到达观测状态 j_2 的概率在同一会话内
StepCountBtw	*filter*(状态，$R_{\texttt{rSteps}=?}[F\ (y=j_2)]$,$(y=j_1)$)：从 j_1到达状态 j_2 所需的预期步数
StateToPattern	$P_{\geq 1}[F\ (x=i_1 \wedge y=j)] \wedge P_{\geq 1}[G\ ((x=i_1 \wedge y=j) \Rightarrow P_{>p}[(x=i_1 \wedge \neg\text{stopS})\ U\ (x=i_2)])]$：以下内容的可能性观测到状态 j 处于活动模式 i_1，从而导致活动模式更改为 i_2，且发生在同一会话
StateToStop	$P_{\geq 1}[F\ (x=i \wedge y=j)] \wedge P_{\geq 1}[G\ ((x=i \wedge y=j) \Rightarrow P_{>p}[(x=i)\ U\ \text{stopS}])]$： 观测到的状态的可能性 j 在活动模式 i 中导致会话结束，但不改变活动模式
LongRunPattern	$S_{=?}[x=i]$： 长期处于活动模式 i 的概率

因此，解释活动模式其他属性所产生的结果过于困难。随后，StateToPattern 和 LongRunPattern 属性分别通过考虑活动模式是否会在一个会话内发生变化，以及每种活动模式的长期概率，帮助我们识别更细致的特征。我们还考虑了其他 PTCL* 属性，但未将其纳入本文。这些属性帮助我们分析某些特定状态是否比其他状态更少经过几个步骤就导致会话结束，以及状态公式之间是否存在相关性（跨状态和/或模式）。

根据观测状态在应用程序中的预期功能和用途进行分组，例如，AppTracker1 中的摘要和具体。分组以迭代方式完成：首先由设计人员提出初步分组；随后分析人员和设计人员分析每种活动模式中的时序属性结果后，可共同修订分组。请注意，此类状态组可能会重叠。定量分析的源代码可在线获取 [1]。

2.5 归纳式编码

虽然 PRISM 模型检验的结果是定量的，但对这些结果的解释具有主观性，因此与定性数据的解释类似。我们采用一般归纳法分析定性评估数据 [46]，旨在让研究发现从数据中自然呈现，而非采用检验假设的演绎法。

两位评估人员（作者 Andrei 和 Calder）独立进行编码，随后进行了多次修改和完善，再由四位作者检查其清晰度，并由利益相关者（担任设计人员的 Morrison 和 Chalmers）进行检查。我们注意到，为标签选择有意义的表述耗费了许多时间。由于标签本身具有内在含义，随着我们对活动模式各种细微差异的理解不断加深，在整个研究过程中，标签也经过了数次更改。我们认为，这既是我们方法的优势，也是其不足。

3 研究结果

我们按照图 1 所示的研究流程概述结果。为简洁起见，我们仅报告 GPAM(2)模型的结果；GPAM(3)模型的详情见附录 C。我们将来自第1天、第1周和第1个月数据集的模型结果称为早期使用结果，将第2个月和第3个月的结果称为熟练用户使用。

为了推断模型，我们使用 Java 实现了鲍姆–韦尔奇算法，并在 2.8GHz 英特尔至强处理器上运行（单线程、单核）；算法重新启动了 200 次，每次重启的最大迭代次数为 100 次。以性能为例，对于由第一个月使用情况构成的数据集，算法为 $K=2$ 耗时 3.2 分钟，为 $K=3$ 耗时 4.1 分钟，为 $K=4$ 耗时 5.3 分钟。

我们首先对所有状态的 VisitProbInit、StepCountInit、VisitCountInit、SessionLength 和 SessionCount 属性进行模型检验，$K=2$ 针对所有状态。我们为 N 选取一个单一值，通常介于 10 和

[1] https://github.com/oanaandrei/temporalanalytics

150；根据我们的经验，50 是一个不错的起始值。请注意，由于状态不可达性、过滤器不满足任何状态，或迭代方法在 100,000 次迭代内未能收敛（该上限由 PRISM模型检查器设定），模型检查器的结果可能不是一个数值（概率或正实数）。在这种情况下，结果显示为“—”。为了便于解读，结果按从“最佳”到“最差”的顺序排列：对于 `VisitProbInit`、`VisitProbBtw` 和 `VisitCountInit`，按数值从大到小排列；对于`StepCountInit` 和`StepCountBtw`，则按数值从小到大排列。这种排序体现了以下判断：访问某个状态的概率越高、状态访问次数越多，以及到达某个状态所需的步数越少，都表明用户对某个状态或一对状态更感兴趣。我们用颜色直观地表示这一排序：用蓝色 表示“最佳”结果，用紫色 表示“最差”结果。

3.1 AppTracker1 交互风格

表格 2 和 3 列出了针对从不同使用时间间隔推断出的 GPAM(2) 模型分析 VisitProbInit、VisitCountInit 和 StepCountInit 时序属性所得的活动模式结果。

如果满足以下条件，我们就称某个状态在某个活动模式中是主要的：(i) 使用 VisitProbInit 计算出的到达该状态的概率大于 0.5，使用 VisitCountInit 计算出的访问次数大于 1，并且使用 StepCountInit 计算出的到达该状态所需时间步数小于为时序属性选定的时间上限 $N = 50$；(ii) 同一模型中不存在其他活动模式，其 VisitCountInit 和 StepCountInit 的得分至少高出三倍。当某组状态（或这些状态所属的状态分组）是该活动模式中的主要状态时，该活动模式便以其为中心 。活动模式根据两个维度进行标记：使用强度 和主要状态。使用强度类型的取值基于对会话频率和长度的抽象，以及 VisitProbInit、VisitCountInit、StepCountInit、VisitProbBtw 和 StepCountBtw 属性针对主要状态所返回的结果，并结合使用强度类型加以确定。

表 4 展示了我们通过归纳式编码过程得出的结果：

– 首先，我们考察会话特征，并使用 Jenks自然断点优化方法 [28] 来确定将会话数量值划分为三类的最佳方式；对于会话长度，我们也采用同样的方法。– 其次，我们根据会话特征和主导状态，对活动模式进行初步分类。我们可以立即看到，较少/较长的会话与较多/较短的会话之间存在相关性。我们注意到，多/长会话和少/短会话并未出现。每种活动模式所围绕的状态标识符按照其结果从高到低列出：某一状态在某种模式中的结果越好（到达该状态的概率越高、访问次数越多、到达所需的步数越少），该状态相较于其他状态的主导程度就越高。我们将一组状态称为 TopLevelMenu，这些状态平均在一到两次按钮按压内即可到达（对应于相应离散时间马尔可夫链中的时间步）。– 第三，我们得出结论：使用强度类型由三个值组成，分别称为 Browsing、Glancing 和 Focussing。– 最后，我们结合使用强度和主导状态组类型，分配四个活动模式标签。请注意，其中有两种可能的组合并未出现：GlancingSummary 和 BrowsingSpecific。

We 可以清楚地看到，早期使用和经验丰富的使用者在活动模式上存在明显差异。 e.

活动模式标注完成后，我们进一步通过模型检验，利用时序属性 LongRunPattern 和 StateToPattern 分析这些模式之间的关系，从而进一步了解交互风格。活动模式的长期概率见图 10(a)，而对于给定的活动模式，某个状态会导致活动模式发生变化（在会话内）的概率见图 6(a)。后者显示，在早期使用阶段，所有状态下从特定浏览一瞥转为浏览摘要行为的概率都很高（接近 1），反向转变的概率则较低（介于 0.5 和 0.75 之间）；Stats 除外，其概率约为 0.2，而总体图表和时段内应用图表的概率更低。这两个例外与 StateToStop 分析（此处未展示）中强调的结果相符：在 Stats、总体图表和时段内应用图表这几种状态下，一旦进入浏览摘要，就不太可能在会话结束前转到其他活动模式。在经验丰富的用户模型中，从聚焦摘要转到 FocussingSpecific 比反向转变更有可能，但总体而言，在一次会话内，这两种模式之间不太可能发生转换。

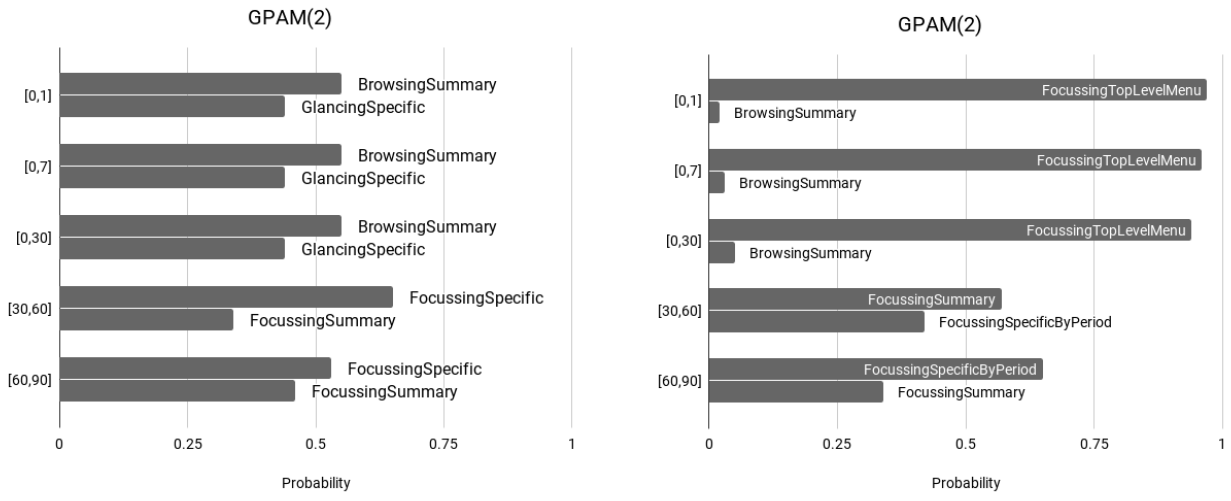
总而言之，基于 GPAM(2)模型的 AppTracker1 交互风格总结如下。早期使用包含浏览和浏览两种模式，后者可能是因为用户会探索应用提供的所有屏幕和功能。对于经验丰富的用户，使用方式是聚焦的，其中具体聚焦略微更为常见。浏览模式的会话在所有模式中屏幕查看次数最少，同时似乎也是数量最多的会话。我们注意到，这些浏览模式与 [22], 中讨论的微型使用相似，后者被定义为“与应用进行的短暂交互”。

表 2。AppTracker1 活动模式分析。属性`VisitProbInit`、`VisitCountInit`和`StepCountInit` 以 $N = 50$实例化后，在五个时间区间内，状态总体使用情况、最近7天、选择时段、统计、`AppsInPeriod`以及我们称为 AP1 和 AP2 的两种活动模式在 GPAM(2)模型上的结果。每一行对应于从相应时间区间的分段数据集中学习得到的一个 GPAM(2)模型。蓝色表示最佳结果，紫色表示某个时间区间的 GPAM(2)模型中，某一状态在各活动模式下的最差结果。

Prop.	时间 / 间隔	总体使用情况 AP1	总体使用情况 AP2	最近7天 AP1	最近7天 AP2	选择时段 AP1	选择时段 AP2	统计 AP1	统计 AP2	AppsInPeriod AP1	AppsInPeriod AP2
VisitProbInit	[0,1]	0.94	0.99	0.80	0.89	0.80	0.42	0.81	0.99	0.45	0.13
	[0,7]	0.67	0.99	0.89	0.88	0.88	0.48	0.55	0.99	0.67	0.21
	[0,30]	0.59	0.99	0.91	0.92	0.90	0.56	0.46	0.98	0.76	0.29
	[30,60]	0.87	0.99	0.98	0.31	0.93	0.00	0.45	0.96	0.77	0.00
	[60,90]	0.91	0.99	0.97	0.02	0.96	0.10	0.56	0.91	0.83	0.09
VisitCountInit	[0,1]	3.54	14.58	1.63	2.24	1.92	0.72	1.74	5.77	0.95	0.28
	[0,7]	1.19	15.25	2.21	2.09	2.75	0.87	0.84	5.27	2.11	0.48
	[0,30]	0.89	15.55	2.39	2.52	2.62	1.19	0.65	4.75	1.95	0.69
	[30,60]	2.28	14.27	5.06	0.40	4.29	0.01	0.80	4.04	4.39	0.01
	[60,90]	3.00	14.73	4.48	0.02	4.61	0.10	1.28	3.63	5.64	0.82
StepCountInit	[0,1]	16.55	4.53	30.27	22.75	30.45	90.36	29.82	12.01	83.40	332.40
	[0,7]	44.55	3.63	22.14	23.24	23.27	<escape>75.96 63.2412.58 45.55</escape>				210.12
	[0,30]	56.55	3.44	20.07	19.28	21.53	59.94	81.13	13.68	35.54	145.35
	[30,60]	23.41	2.15	9.02	137.09	18.90	5483.99	85.45	15.97	34.45	25915.18
	[60,90]	19.55	2.23	10.46	2269.78	15.49	483.09	61.19	21.01	28.65	532.74

表3. AppTracker1。GPAM(2) 中会话的特征：针对状态 UseStop，VisitProbInit 计算到达会话结束的概率，而 SessionCount 和 SessionLength 分别计算预期会话次数和每个会话中的预期时间步数（按钮点击）。

时间 / 间隔	VisitProbInit AP1	VisitProbInit AP2	会话次数 AP1	会话次数 AP2	SessionLength AP1	SessionLength AP2
[0, 1]	0.99	0.31	10.13	0.37	3.86	130.96
[0, 7]	0.99	0.43	10.20	0.54	3.81	87.76
[0, 30]	0.99	0.38	10.82	0.47	3.51	102.07
[30, 60]	0.99	0.99	6.17	7.55	7.09	5.36
[60, 90]	0.99	0.99	5.43	7.35	8.28	5.56



(a) AppTracker1 (b) AppTracker2

图 5. AppTracker1 和 AppTracker2。不同时间区间下，GPAM(2)模型中长期处于各活动模式的概率可视化；每个时间区间内的条目按降序排列。这些结果基于对每种活动模式的时序属性 LongRunPattern 进行模型检验。

一个值得探讨的重要问题是：我们发现的潜在结构是否只反映了顶层菜单结构？为了回答这个问题，我们研究了 GPAM(3)模型（见附录 C），因为主菜单有三个

表 4。 AppTracker1。GPAM(2)。归纳式编码：对平均会话次数和长度进行分类，随后将其与主要状态一同用于活动模式的初步分类，并描述使用强度。状态标识符（id）如下：3 表示总体使用情况，4 表示最近7天，5 表示选择时段，6 表示时段内应用，9 表示统计，10 表示总体图表，12 表示统计图表，14 表示时段内应用图表。状态标识符以摘要状态和特定状态突出显示。状态 4 根据上下文具有双重角色，可作为摘要状态或特定状态。

会话次数类别		
few	**mid**	**many**
$\{0.37, 0.47, 0.54\}$	$\{5.43, 6.17, 7.35, 7.55\}$	$\{10.13, 10.82\}$

会话时长类别		
简短	**mid**	**long**
$\{3.51, 3.81, 3.86\}$	$\{5.36, 5.56, 7.09, 8.28\}$	$\{87.76, 102.07, 130.96\}$

Time 间隔	**AP1**		**AP2**	
	会话特征	**主要状态**	**会话特征**	**主要状态**
$[0, 1]$	多/短	5,	少/长	3,
$[0, 7]$	多/短	5,	少/长	3,
$[0, 30]$	多/短	5,	少/长	3,
$[30, 60]$	中/中	4,	中/中	3,
$[60, 90]$	中/中	6,	中/中	3,

使用强度	**使用强度特征**
	浏览：少/长会话，集中于顶层菜单中的至少三个状态，有时来自不同的状态组。浏览：多/短会话，集中于一个或两个状态。聚焦：中/中会话，集中于同一组中的状态。

时间间隔	**AP1 标签**	**AP2 标签**
$[0, 1]$	特定浏览一瞥	浏览摘要
$[0, 7]$	特定浏览一瞥	浏览摘要
$[0, 30]$	特定浏览一瞥	浏览摘要
$[30, 60]$	FocussingSpecific	聚焦摘要
$[60, 90]$	FocussingSpecific	聚焦摘要

选项（不包括“设置”）。如果活动模式只是反映菜单结构，那么我们本应预期 GPAM(3) 中的每种模式都以上述某个状态为中心，且这些状态两两之间的相关性极低。但事实并非如此。在全部五个 GPAM(3) 模型中，我们都发现了使用强度为“聚焦”或“浏览”的活动模式，它们以“最近7天”以及“选择时段”或“AppsInPeriod”为中心；没有任何模型包含这样两种模式：一种以“选择时段”或“AppsInPeriod”为中心、但不包含“最近7天”，另一种以“最近7天”为中心、但不包含“选择时段”或“AppsInPeriod”。

3.2 为熟练用户使用重新设计界面

显然，AppTracker1 的顶层菜单并不适合用户的交互风格。例如，我们发现了浏览行为：用户会快速查看应用，以了解自己的使用行为，但这与菜单结构不太契合。用户可以在第一个菜单项中浏览历史上使用最多的应用（总体使用状态），但如果用户已经使用 AppTracker1 很长时间，这份列表就很可能逐渐变得一成不变。相反，虽然用户可以查看近期（例如今日）的使用情况，但这需要经过菜单中的多个步骤，才能看到更详细的信息。

我们选择专注于面向熟练使用的重新设计；这些用户是在一段时间内自愿持续使用该应用的人，因此我们也预计能观察到更稳定的交互风格，此时用户的初期学习过程已逐渐结束。另一种同样合理的选择是关注使用早期阶段的用户，例如解决一些问题，以改善用户在初期体验之后的留存率。

我们的目标并非改变 AppTracker 的总体用途或添加新功能，而是通过添加、删除或移动状态来重新配置菜单结构，以更好地支持高效地浏览摘要状态和聚焦特定状态。这些行为与现有菜单布局并不十分契合。例如，我们发现 AppsInPeriod（位于具体子菜单中）会出现在浏览活动模式中，因为用户可能会深入浏览具体菜单，以查看当天的活动。表 4 也

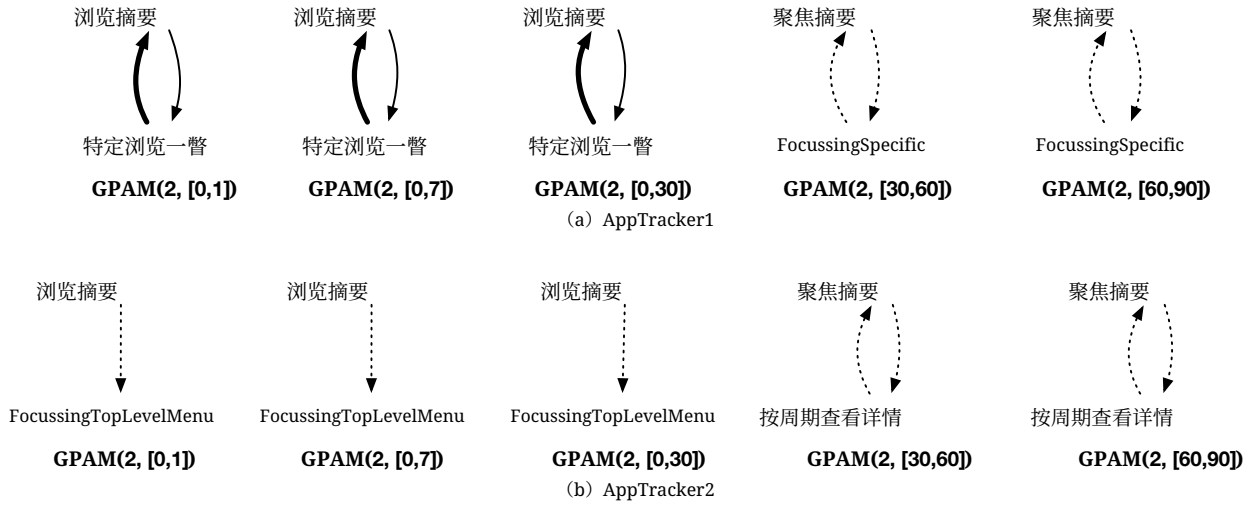


图 6. AppTracker1 和 AppTracker2。在 GPAM(2)模型中，针对不同时间区间，展示会话中从某一状态转变为某一活动模式的平均概率。除统计和统计图表状态外（这些状态对应的概率低于平均值），所有状态对应的概率都非常相似。虚线箭头表示 $[0.02, 0.2)$ 中的概率，细箭头表示 $[0.5, 0.8)$ 中的概率，粗箭头表示 ≥ 0.8。低于 0.02 的概率不予显示。这些结果基于对多种状态与活动模式组合的时序属性 StateToPattern 进行模型检验得出。

显示了熟练用户使用期间围绕摘要状态和具体状态的活动聚类，但效率不高；用户似乎聚焦于摘要状态，而这些信息本可以通过一种便于浏览的方式呈现，让用户更快地获取所需信息。

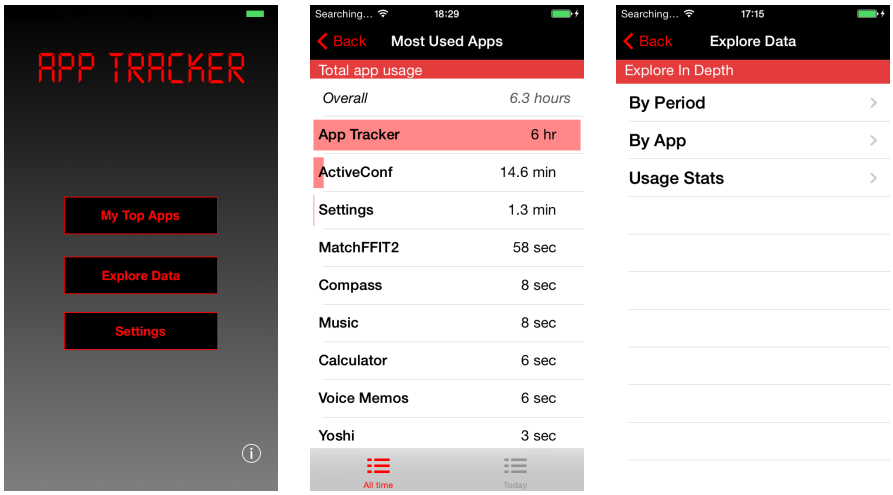
因此，我们调整了顶层菜单结构，将原先的三个选项改为两个（另加“设置”）。这两个选项分别对应浏览使用和聚焦使用，并分别命名为 我的热门应用和 探索数据。对应的两个状态组为摘要状态和特定状态（后者是 SpecificMenu、按周期查看详情、按应用查看详情和 SpecificStats 状态组的并集）。因此，层级结构中更深处的若干状态被移动或拆分，即将活动模式围绕其展开的状态移至与该活动模式对应的选项附近。为了支持浏览使用行为，我的热门应用 仅包含表格，显示：（i）自安装 AppTracker 以来用户使用最多的应用；以及（ii）当天使用最多的应用。通过这种方式，重新设计后的菜单旨在让用户能更轻松地从顶层菜单访问当天的使用情况。

这些变化如图 8 所示，并总结如下：

– 对应用菜单进行了重新组织，以专门支持浏览和聚焦这两种主要使用方式，因此仅保留两个主菜单选项，让这些行为更加鲜明。– 屏幕视图“总体使用情况”被一个更适合浏览的新视图“OverallAll”取代，该视图不支持深入查看详细使用情况。– 菜单的摘要部分新增了适合浏览的屏幕视图“今日应用”。– 屏幕视图“最近7天”已移除。– 菜单中的“特定状态”组被拆分为更细致的子成分：“按周期查看详情”、“按应用查看详情”和“SpecificStats”。– 屏幕视图“总体使用情况”被移至“具体”部分，并重命名为“按应用查看总览”，与“按应用查看总览图表”并列；这两个新状态归入“按应用查看详情”。– 屏幕视图“统计”被移至菜单的“具体”部分，与“统计图表”并列；这两个新状态归入“SpecificStats”。

AppTracker2 的主菜单界面提供了三个主要选项（图 7(a)），并包含 18 个用户触发事件。新增状态如下：

- OverallAll：自安装 AppTracker2 以来设备使用情况的汇总统计数据，
- ExploreData：提供三个选项，用于更深入地探索所有记录的数据，
- 今日应用：当天应用使用情况的摘要，
- 按时段查看应用：按所选时间段查看各种应用的使用统计，
- ChartAppsInPeriod：从按时段查看应用中选择后显示的详细应用使用情况，



(a) 主菜单视图 (b) 我的热门应用子-menu (c) 探索数据子-menu

图 7. AppTracker2 的屏幕截图：(a) 主菜单视图对应状态 Main； *My Top Apps* 选项可将用户带至状态 OverallAll, *Explore Data* 可将用户带至状态 ExploreData， *Settings* 可将用户带至状态 Settings。(b) *My Top Apps* 视图（对应状态 OverallAll）显示自安装应用以来设备的总体使用情况，并提供查看今日使用情况的选项——右下角的选项 *Today*。(c) *Explore Data* 视图提供了探索数据的选项： *By Period* （状态 SelectPeriod）、 *By App* （状态 OverallbyApp），以及查看*Usage Stats* （状态 Stats）。

- OverallbyApp：所选应用的使用摘要，
- 按应用查看总览图表：从按应用查看总览中选中时，显示详细的应用使用情况。

根据其在整个 AppTracker2 菜单中的位置、从 AppTracker1 交互风格中获得的见解，以及 AppTracker2 的设计意图，对状态进行分组。AppTracker2 于2016年发布，我们的数据集取自为期六个月的600条用户轨迹样本。

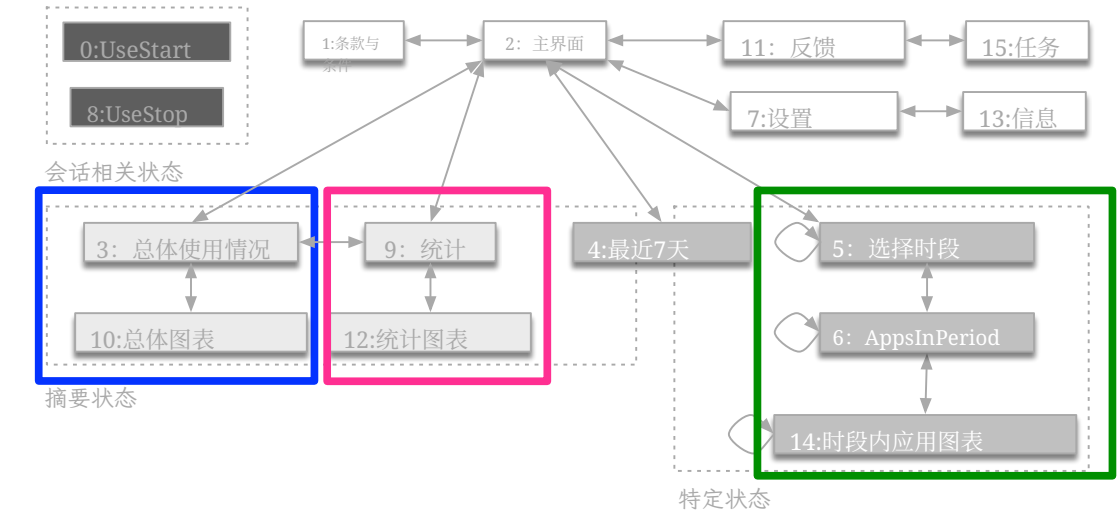
3.3 AppTracker2 交互风格

为简洁起见，我们仅展示表5中 GPAM(2)模型归纳式编码过程的具体步骤。各标签的含义不言自明，但需要注意的是，摘要状态的分组方式与 AppTracker1 中的分组有所不同，参见图 8。特别是，新增的状态 AppsToday（状态 8）出现在除 FocussingSpecificByPeriod 之外的所有活动模式中。主要结果再次表明，存在四种不同的活动模式，并且早期使用与熟练用户使用这两种风格之间存在明显的区分。长期概率见图 10(b)，而对于给定活动模式，某一状态导致活动模式发生变化（在同一会话内）的概率则见图 6(b)。

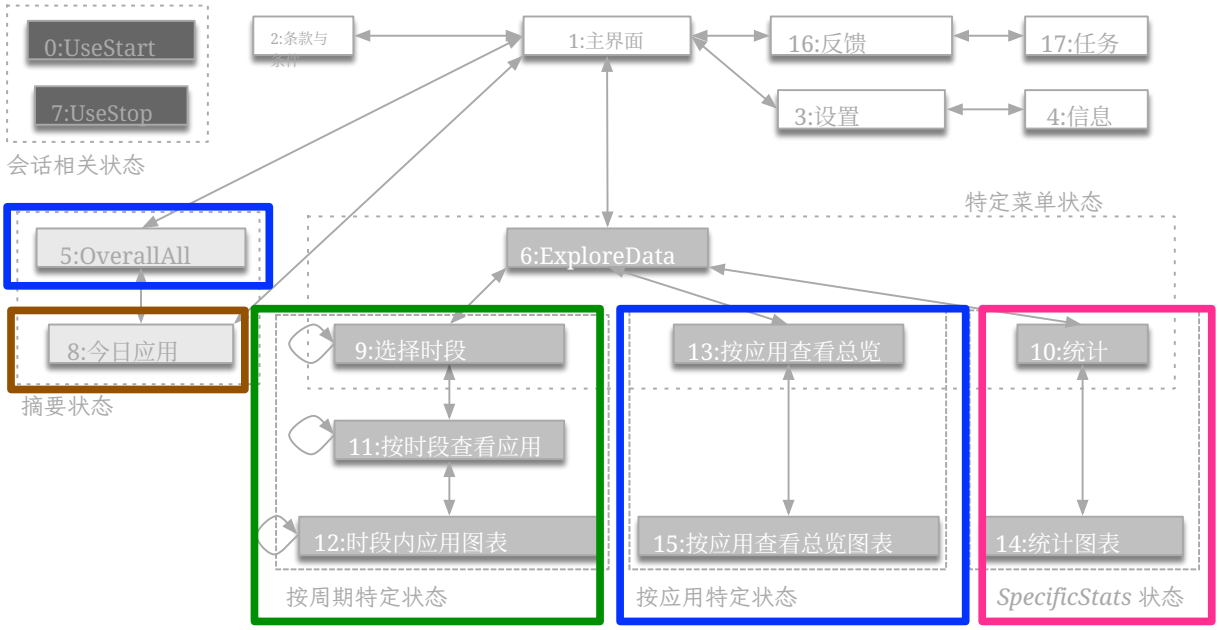
AppTracker2 使用情况的 GPAM(2)模型所呈现的交互风格总结如下。总体而言，会话更少、时长更短，不存在短会话，也不存在中/中组合。聚焦行为明显更多。在早期几天的模型中，浏览行为很少，这可能表明 App- Tracker2 易于使用，也可能表明大多数用户已经熟练掌握了这个应用的第二个版本。所有模型中的转移概率都很低。

3.4 比较 AppTracker1 和 AppTracker2 的交互风格

将 AppTracker2 与 AppTracker1 的交互风格进行比较，可以发现活动模式之间有许多相似之处，这也反映在标签名称上。考虑到此次重新设计涉及菜单重组，而非全面的应用程序重新设计，这并不令人意外。图 5 和图 6 展示了这些风格的主要差异。我们将此次重新设计对交互风格的影响概括如下：AppTracker2 中没有短会话；所有模型中的聚焦都更多，其中有经验的用户更可能聚焦特定状态；浏览行为很少且发生概率较低；只有在早期使用模型中，新的摘要状态“今日应用”才会大量出现；此外，用户离开某个活动模式的可能性也更低。最后一点或许表明，应用程序与用户意图更加一致，尤其是与有经验用户的意图更加一致；也可能是因为大多数用户已经熟练掌握了这个第二版应用。有人可能会认为，在早期使用中观察到的 FocussingTopLevelMenu 是一种持续时间更长、但仍专注的浏览。值得注意的是，尽管此次重新设计主要面向有经验的用户，但它对早期用户产生了显著的积极影响。



AppTracker1



AppTracker2

图 8. AppTracker1 和 AppTracker1 的界面设计。两个应用版本采用相同的颜色以表示对应关系。

3.5 至少包含三个簇的模型所揭示的洞见

将簇的数量增加到 K 3 后，我们发现用户对选定状态或状态集合的兴趣更加细致，也发现了 AppTracker2 中出现极短会话的可能性。例如，在 GPAM(3) 中，我们再次发现了熟练用户使用中的浏览行为，但它完全聚焦于主界面菜单或（新增的）摘要状态。附录 C 中对 AppTracker1 和 AppTracker2 的 GPAM(3)模型进行了简要分析。

我们观察到，随着模型成分数量增加，我们会发现对所选状态或状态组更细致的关注（例如 GPAM(4) 和 GPAM(5) 模型中的 FocussingSpecific），而当 K 较小时，这种关注并不那么明显。对于 $K \geq 4$，我们会看到更多以各个状态为中心的活动模式。一般而言，K 并不存在最优值。它是一种探索性工具，其用途取决于状态组数量、状态图，以及对重新设计有帮助的细粒度程度。在本研究中，我们发现 2-3 个簇最为有用。请注意，我们至少需要研究 $K = 3$，以调查活动模式是否与顶层菜单选项相吻合。此外，分析 GPAM(4) 和 GPAM(5) 模型也很重要，以确认它们并未揭示出显著不同或对重新设计更有用的活动模式；此处未列出 GPAM(4) 和 GPAM(5) 分析的详细信息。

表5. AppTracker2、GPAM(2)模型。对平均会话次数和长度进行分类，随后用于活动模式的初步分类，并结合主导状态进行描述，同时说明使用强度标签。状态标识符（ID）如下：5 为 OverallAll，6 为 ExploreData，8 为 今日应用，9 为 选择时段，10 为 统计，11 为 按时段查看应用，12 为 时段内应用图表，13 为 按应用查看总览，14 为 统计图表，15 为 按应用查看总览图表。状态标识符以摘要状态和特定状态的形式突出显示。

会话次数类别	
few	**many**
$\{1.40, 2.10, 3.04\}$	$\{5.35, 5.86, 5.87, 6.23, 6.39, 6.52, 7.02\}$

会话时长类别	
mid	**long**
$\{6.14, 6.57, 6.68, 6.88, 7.52, 7.84, 8.31\}$	$\{17.61, 25.08, 38.84\}$

Time int.	**AP1**		**AP2**	
	会话	**状态 ID**	**会话**	**状态 ID**
$[0, 1]$	少/长	5, 8 ,	多/中	5 , 6 ,
$[0, 7]$	少/长	5、8、	多/中	5， 6，
$[0, 30]$	少/长	5、	多/中	5 , 6 ,
$[30, 60]$	多/中	9， 6，	多/中	5, 8, 6 ,
$[60, 90]$	多/中	6, 9, 1	多/中	5, 8 ,

使用强度	**使用强度特征**
浏览	少/长会话为中心 顶层菜单中，有时来自不同的状态组 聚焦于以同一组中的状态为中心的多/中会话
	在至少三个状态上

时间间隔	**AP1 标签**	**AP2 标签**
$[0, 1]$	浏览摘要	FocussingTopLevelMenu
$[0, 7]$	浏览摘要	FocussingTopLevelMenu
$[0, 30]$	浏览摘要	FocussingTopLevelMenu
$[30, 60]$	按周期查看详情 聚焦摘要	
$[60, 90]$	FocussingSpecificByPeriod 聚焦摘要	

4 讨论

我们探讨了使用数据驱动的机器学习模型和计算方法所带来的一些启示。

4.1 数据可靠性与分段

我们发现并随后修复了日志中的若干错误，例如缺少会话的开始或结束记录、缺少事件以及时间戳异常。这些问题均由日志记录框架 SGLog 与 iOS 之间的交互导致，例如应用在数据写入文件前崩溃，或数据传输至我们的分析服务器时发生网络故障。由于我们的研究重点是长期参与，因此我们舍弃了会话数少于五次的用户轨迹，因为这类短用户轨迹只会给数据集增添更多噪声。将五次会话设为用户轨迹最短长度的门槛是基于经验作出的选择，我们无法排除采用不同的最低门槛会得出不同结果的可能性。

我们选择了不同的时间区间，涵盖从初始参与（首日）到持续参与（数月）的使用体验。对于这一应用而言，这些区间值得关注，也符合我们的研究目的。更普遍地说，设计人员可以选择在不同时间关注不同的区间。例如，在初始设计发布后不久，他们可能会聚焦于根据首日使用数据（甚至更短的时间，例如最初五分钟）推断出的模型；但在后续设计中，他们可能只考虑根据长期参与用户的使用数据推断出的模型。

4.2 数据集偏差

AppTracker2作为AppTracker1的一次更新版发布（两者仅可在越狱iOS设备上的Cydia应用商店中获取）。因此，AppTracker2用户可能是安装了更新的现有用户，也可能是刚开始使用该应用的新用户。这可能会影响交互风格，因为现有用户

会更熟悉系统，也已有记录的信息，因此会看到已经填充数据的图表和表格；而新用户在最初几天看到的则是空白屏幕。更广泛地说，区分新用户可能颇具挑战，因为用户可以重新安装应用或购买新设备。例如，苹果公司曾更改应用可访问的唯一设备标识符类型，以便跨安装跟踪用户。后来，苹果公司完全禁止了这种做法，因此，如果用户卸载某个开发者的所有应用后再重新安装，系统就会生成一个全新的标识符，且不会提供任何信息来关联第二次安装的用户与第一次安装的用户（反之亦然） [9,40]。这些问题是开展应用真实世界部署而非进行限制更多的实验室式研究时固有的。我们认为，成功招募大量用户带来了显著收益；此外，用户通过自主选择下载我们的应用，而非被明确招募为试验参与者，也提升了研究的外部效度。随后，我们研究了人们如何在日常生活中使用安装于自己设备上的应用。

4.3 状态选择、数据量和可扩展性

我们的研究受益于拥有大量日志数据，这为可靠地应用机器学习方法提供了条件。需要注意的是，日志数据的数量和用户轨迹的数量都与概率模型检验无关；只有观测状态的数量决定其复杂度。AppTracker 研究涉及 16 至 18 种观测状态，这一规模完全可以应对，但模型检测在状态数量增加一个数量级时就会遇到困难。菜单选项丰富得多的应用，或进一步纳入位置、人口统计信息、事件持续时间、一天或一周中的时间等分类变量，都可能导致这种情况。

4.4 用户熟练度与专注度随时间的变化

较简单的分析方法可能会追求绝对指标，例如在应用中花费更长时间或更频繁地启动应用，并将其作为参与度的衡量标准 [4]。我们没有将使用强度与使用经验和/或熟练程度混为一谈。我们的目标是支持观察到的风格，并指出，熟练用户使用模型中同时存在聚焦型和浏览型风格。

4.5 进一步的重新设计可能性

新用户。 如果新用户与长期用户之间存在差异（例如，使用第一天/周/月的用户与使用第二/第三个月的用户），则添加支持应用程序“新手引导”并平稳转移到另一种交互风格的新功能。

微型使用。 关注那些在极短时间内完成的活动，这类活动可能更适合通过小组件来实现，而非导航至完整的应用程序。这里所说的小组件，是指设备图形界面中的一种简单附加元素，通常作为该设备上运行的某个应用程序的补充。

快捷方式。 识别活动模式中最受欢迎的初始状态，并为这些状态实现快捷方式。

拆分应用程序。如果每个会话中（几乎）始终只有一种活动模式，且不同模式之间不会重叠，那么就将该应用程序拆分为两个（或更多）独立的应用程序。

5 相关工作

5.1 我们之前的工作

多年来，我们不断完善分析方法：改进马尔可夫模型、时序属性以及数据集分段。我们的首项研究 [3] 涉及 iOS 多人游戏 Hungry Yoshi [35], 一个数据集、一个简单的推断模型、简单的时序逻辑性质，以及 $K = 2$。与 AppTracker 不同，该应用有明确的（用户）目标状态；除了用户交互之外，设备扫描到 Wi-Fi 接入点时还会产生外部事件。我们发现了两种代表不同游戏策略的交互风格，但由于苹果公司的 iOS 发生了变化，我们无法再扫描 Wi-Fi 接入点，因此无法实施重新设计。该应用已不再开发。我们对 AppTracker 的初步分析 [2]采用了不同的推断混合模型、一组受限的通用时序属性，并对数据集进行了分段。该研究没有采用归纳式编码，也没有重新设计菜单。GPAM 模型在 [1],中定义；我们还在其中引入了针对潜变量的逻辑公式，但没有考察活动模式之间转移的概率。值得注意的是，我们早期的工作都没有涉及具体的设计建议、新设计的实施与部署，也没有分析新设计中的交互风格。

5.2 其他相关工作

我们最初的建模方法受到一项关于使用单纯形混合模型对网页浏览和电话使用进行建模的实证研究的启发 [24]。值得注意的是，同年，Bowring 等人 [11] 在提出未来可使用隐马尔可夫模型进行软件自动分类时，引用了 Girolami 等人的研究 [24] 。据我们所知，尚无人研究使用混合模型对交互行为进行建模；然而，近期也有研究指出，大型移动应用用户群体中存在不同的用户群体或用户类型 [15,29,52] ，并提出了混合模型。

近 30 年前，Whittaker 和 Poore 提出了将马尔可夫链作为软件使用模型的方法 [50]，其中转移概率是根据执行轨迹中所有二元组的频数估算得出的。我们可以将该模型视为只有一个组成部分的 GPAM，即 GPAM(1)。马尔可夫链模型也被用于描述软件行为，见 [16,45] ，近期则见 [23,12,34,30,48,20]。 [30], 使用统计方法揭示使用风格，但并未将其视为隐马尔可夫模型变体中的潜在状态。 [50] 或 [45] 所用的分析技术，是对转移矩阵进行经典数学运算，例如计算处于某一状态的长期概率，或首次到达某一状态所需转移的预期状态数（平均首次通过时间）；而我们使用概率时序逻辑性质，这使得我们能够提出表达能力更强的性质，并通过概率模型检测工具自动分析这些性质。一阶马尔可夫模型已用于许多相关问题的建模，例如：人类网页导航，其中状态对应访问过的网页 [10,14,42,41,23],；更具体地说，是点击流数据 [36,33,8],；可用性分析，其中状态通常对应按钮按压 [45], ；移动应用，其中状态对应设备屏幕事件 [3,30], ；以及人与搜索引擎的交互 [47]。

Banovic 等人 [6] 对移动应用使用情况的研究，提供了有关持续时间以及瞥视、查看和参与等交互类型的证据，并据此描述会话。我们也识别出三种使用强度类型（瞥视、浏览和聚焦），但它们是根据会话内交互的数量和会话频率来描述的。我们注意到，我们所说的瞥视和 [6] 中的瞥视都涉及微型使用。我们认为，我们对瞥视行为的描述更接近于 [39] 中的查看习惯，即快速查看设备上易于访问的动态内容，这种查看简短且重复。BEAR工具 [23]与此类似：它从日志中推断离散时间马尔可夫链，并使用概率时序逻辑性质和 PRISM 查询模型；但关键区别在于，用户根据静态属性进行分类，例如设计者预先定义的时区或操作系统，而且假定行为是同质的，也就是说，不存在类别内差异，也无法表达或检测混合行为。 [12], 也提出了这一问题，指出用于划分日志数据的筛选器可能会显著影响最终模型及后续分析。

我们还注意到，离散时间马尔可夫链是用于描述 [43], 中使用模式的模型；在该场景中，研究人员基于离散时间马尔可夫链的交互式图形表示的可视化，以及不同长度的聚类序列来分析移动健康应用。如上所述，概率时序逻辑为分析离散时间马尔可夫链提供了额外的分析工具，可带来超越图形表示所获得见解的分析洞见。

其他运用计算交互方法对人类行为建模的途径，会利用用户触发事件日志，包括马尔可夫决策过程（MDP） [7] （识别不同个体和总体中的惯例）以及部分可观测马尔可夫决策过程（POMDP） [13,26]。我们可以通过概率模型检验分析这类马尔可夫模型的性质，从而运用时序逻辑的强大功能，并按照我们的方法解读结果。

6 结论

交互重设计与数据密不可分，但过去我们缺乏定量工具，无法在大规模场景下发现比任务更细致入微的交互风格。本文展示了如何利用基于无监督机器学习聚类和概率时序逻辑的新型计算方法，为研究和解读交互风格提供新的定量工具。

我们推断出的混合马尔可夫模型体现了用户希望如何使用 App- Tracker，以及他们实际上如何使用 AppTracker 的方式。这项研究的结果令我们大开眼界：对于早期使用与熟练用户使用之间可能存在的差异，以及我们会在 AppTracker1 和 AppTracker2 中发现何种活动模式，我们此前都没有任何预设。我们发现，AppTracker1 的顶层菜单并不适合用户与应用交互的方式。例如，我们识别出一些熟练用户使用风格，主要由浏览和聚焦模式构成，分别以 AppTracker1 的摘要状态和具体状态为中心。这些风格与现有的菜单布局并不匹配。例如，AppsInPeriod（位于具体子菜单中）会出现在浏览活动模式中，因为用户会深入具体菜单，查看当天的活动。反过来，我们也发现用户会聚焦于摘要状态，而相关信息本可以通过重新组织呈现方式，让用户以浏览的方式获取。因此，我们重新设计了界面，将三个主要选项减少为两个，并将活动模式所围绕的状态移至

与该选项相近的位置。我们在 AppTracker2 中发现的交互风格表明，它实现了重新设计的目的：没有短会话，所有模型中的聚焦行为都更多；而在 AppTracker1 早期使用中普遍存在的浏览行为几乎消失了，在早期使用模型中的可能性也很低。总体而言，活动模式之间转移的可能性降低了，这表明用户能更快找到合适的风格。只有通过使用包含潜变量的混合（而非混合）马尔可夫模型及 StateToPattern 属性，才能获得这些洞见。

关于将机器学习和计算方法用于交互设计研究的普遍担忧，即使不是全部，也包括数据偏差、数据可靠性以及模型的时间有效性。前两者是现实世界部署的结果，而非基于实验室的研究。第一项担忧包含一种我们无法消除的偏差：现有用户可能对第二次设计产生影响。在绝对确定的情况下无法区分现有用户和新用户，是现实世界应用部署的一个方面，也因此是使用计算方法时需要考虑的因素。第二项同样是现实世界部署的一个方面，因为不同系统之间的交互（包括通信和数据收集系统）会影响数据可靠性。最后一项与重新设计有关——从新的数据集推断模型的频率，取决于底层数据变化的速度和程度。在应用计算方法时，有时很容易把“数据集”称为一个单一整体，仿佛它是一个不可分割的实体。本研究突出了数据分段（在我们的案例中是时间分段）对模型以及后续纵向分析和决策的影响。

我们强调，数据本身并不会自动说明任何问题，分析人员需要提出有意义的问题并进行可视化。对于分析人员而言，在提出这些问题时，一个至关重要的工具是基于潜变量的概率时序逻辑公式——即通过机器学习推断所发现的活动模式。我们需要一种 时序 逻辑来推理计算路径，并表达会话中观测状态与行为之间的关系。我们发现，关于活动模式变化可能性（StateToPattern）和长期行为（LongRunPattern）的性质，是我们分析中最强大、最有用的方面，因为这些性质使我们能够了解在特定时间区间内，哪些活动模式更受欢迎，哪些只是暂时出现的，从而进一步深入了解不同的交互风格，为重新设计提供依据。

作为未来的工作，我们可以在每个事件之前添加停留时间类别和静态用户属性。我们还可以引入定性方法，例如用户访谈，以深入了解某种活动模式背后的意图。一种新颖的在线方法是在用户持续一段时间采用某种具体活动模式，或开始采用某种具体模式时，弹出问卷；这将扩展 [27] 中提出的情境触发式体验采样方法。另一种互补且更为被动的在线方法是有选择地启动更频繁、更精细和/或不同类型的跟踪与感知，以便更深入地了解用户情况，同时以临时方式开展，并充分考虑到这可能给用户带来的成本，例如增加电池消耗和数据传输费用。我们还可以尝试使用来自不同子群体的数据集，例如比较长期参与的用户与短时间后便停止参与的用户的活动模式。最后，我们可以实现可视化的自动化，并允许用户与分析结果进行交互，例如点击长期概率图表中的某个单独柱形，以显示相应的主导状态。

致谢。本研究得到 UKRI-EPSRC 项目资助，项目编号为 EP/J007617/1（泛在计算系统设计的人口方法）和 EP/N007565（传感器系统软件科学）。

参考文献

1. O. Andrei 和 M. Calder。交互式软件使用的数据驱动建模与概率分析。 *Journal of Algebraic and Logical Methods in Programming (JLAMP)* 100 195 214, 2018 , : – 。2. O. Andrei、M. Calder、M. Chalmers、A. Morrison 和 M. Rost。用于指导重新设计的应用使用情况概率形式化分析。载于 E. ´Abrah ´ 9681 *Lecture Notes in Computer* am 和 M. Huisman（编）， *Science* 115 129 2016 *Proc. of iFM'16*，第 卷，页 – 。Springer，。3. O. Andrei、M. Calder、M. Higgs 和 M. Girolami。用户活动模式离散时间马尔可夫链模型的概率模型检验。载于 G. Norman 和 W. H. Sanders（编）， *Proc. of QEST 2014*，第 8657 *Lecture Notes in Computer* 卷，页 – 。Springer，。 *Science* 138 153 2014 4. S. Attfield、G. Kazai、M. Lalmas 和 B. Piwowarski。迈向用户参与度科学（立场论文）。载于 *Pre-proceeding of the ACM WSDM Workshop on User Modelling for Web Applications*，第 9–12 页，2011年。5. C. Baier 和 J.-P. Katoen。 *Principles of Model Checking*。麻省理工学院出版社，2008年。6. N. Banovic、C. Brant、J. Mankoff 和 A. K. Dey。ProactiveTasks：移动设备使用会话概述。载于 A. J. Quigley、S. Diamond、P. Irani 和 S. Subramanian（编） *Proceedings of the 16th international conference on Human-computer interaction with mobile devices & services (MobileHC'14)* 243 252 2014 ，页 – 。ACM，。7. N. Banovic、T. Buzali、F. Chevalier、J. Mankoff 和 A. K. Dey。人类日常行为的建模与理解。载于 J. Kaye、A. Druin、C. Lampe、D. Morris 和 J. P. Hourcade（编）， *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems* 248 260 2016 ，页 – 。ACM，。8. F. Benevenuto、T. Rodrigues、M. Cha 和 V. A. F. Almeida。在线社交网络中用户行为的特征分析。载于 A. Feldmann 和 L. Mathy（编）， *Proceedings of the 9th ACM SIGCOMM Internet Measurement Conference (IMC'09)* 49 62 2009 ，页 – 。ACM，。

9. H. Bojinov、Y. Michalevsky、G. Nakibly 和 D. Boneh。通过传感器指纹识别移动设备。 *CoRR*, abs/1408.1416，2014年。10. J. Borges 和 M. Levene。用户导航模式数据挖掘。载于 B. Masand 和 M. Spiliopoulou（编），*Web Usage Analysis and User Profiling: International WEBKDD'99 Workshop*，第 92–112 页。Springer，2000年。11. J. F. Bowring、J. M. Rehg 和 M. J. Harrold。通过主动学习自动分类软件行为。载于 G. S. Avrunin 和 G. Rothermel（编） *Proceedings of the International Symposium on Software Testing and Analysis (ISSTA'04)* 195 205 2004 ，页 – 。ACM，。12. N. Busany 和 S. Maoz。具有统计保证的行为日志分析。载于 *Proceedings of the 38th International Conference on Software Engineering (ICSE'16)* 877 887 2016 ，第，页 – 。ACM，。13. X. Chen、G. Bailly、D. P. Brumby、A. Oulasvirta 和 A. Howes。交互行为的涌现：理性菜单搜索模型。载于 B. Begole、J. Kim、K. Inkpen 和 W. Woo（编），*Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems (CHI'15)* 4217 4226 2015 ，页 – 。ACM，。14. F. Chierichetti、R. Kumar、P. Raghavan 和 T. Sarl´os。网络用户真的具有马尔可夫性吗？载于 A. Mille、F. L. Gandon、J. Misselis、M. Rabinovich 和 S. Staab（编），*Proceedings of the 21st World Wide Web Conference 2012 (WWW'12)* 609 618 2012 ，页 – 。ACM，。15. K. Church、D. Ferreira、N. Banovic 和 K. Lyons。理解手机使用数据面临的挑战。载于 S. Boring、F. Rukzio、H. Gellersen 和 K. Hinckley（编），*Proceedings of the 17th International Conference on Human-Computer Interaction with Mobile Devices and Services (MobileHCI'15)* 504 514 2015 ，页 – 。ACM，。16. J. E. Cook 和 A. L. Wolf。从基于事件的数据中发现软件过程模型。 *ACM Trans. Softw.* , ( ): – 。17. A. P. Dempster、N. M. Laird 和 D. B. Rubin。通过期望最大化算法从不完整数据中求最大似然。 *Eng. Methodol.* 7 3 215 249, 1998 *Journal of the Royal Statistical Society. Series B (Methodological)*, 39(1):1–38，1977年。18. A. J. Dix。面向挪用的设计。载于 T. C. Ormerod 和 C. Sas（编），*Proceedings of the 21st British HCI Group Annual Conference on People and Computers: HCI...but not as we know it - Volume 2* 07, ，BCS-HCI' 第 27–30 页。英国计算机学会，2007年。19. P. Dourish。交互技术的挪用：从无地点文档中获得的一些经验。 *Computer Supported Cooperative Work (CSCW)* 12 4 465 490, 2003 , ( ): – 。20. S. S. Elbaum、A. Gopinath、G. Rothermel、C. Kanduri 和 J. Miller。从软件执行中推断扩展概率有限状态自动机模型。 *ACM Trans. Softw. Eng. Methodol.*, 27(1):4:1–4:39，2018年。21. X. Ferre、E. Villalba、H. Julio 和 H. Zhu。扩展移动应用分析以记录可用性测试。载于 *IFIP Conference on Human-Computer Interaction INTERACT 2017* 114 131 2017 ，页 – 。Springer，。22. D. Ferreira、J. Gonçalves、V. Kostakos、L. Barkhuus 和 A. K. Dey。对移动应用微观使用进行情境体验抽样。载于 A. J. Quigley、S. Diamond、P. Irani 和 S. Subramanian（编）， *Proceedings of the 16th international conference on Human-computer interaction with mobile devices & services, MobileHCI 2014, Toronto ON Canada September 23-26 2014* 91 100 2014 ，第，页 – 。ACM，。23. C. Ghezzi、M. Pezz`e、M. Sama 和 G. Tamburrelli。从用户密集型网络应用中挖掘行为模型。载于 P. Jalote、L. C. Briand 和 A. van der Hoek（编）， *Proc. of ICSE'14*，第 277–287 页。ACM，2014年。24. M. Girolami 和 A. Kab´an。马尔可夫链的单纯形混合：动态用户画像的分布式建模。载于 S. Thrun、L. K. Saul 和 B. Sch¨olkopf（编）， *Advances in Neural Information Processing Systems 16 (NIPS'03)*，第 9–16 页。麻省理工学院出版社，2004年。25. M. Hall、M. Bell、A. Morrison、S. Reeves、S. Sherwood 和 M. Chalmers。调整泛在计算软件及其评估。载于 T. C. N. Graham、G. Calvary 和 P. D. Gray（编）， *Proc. of EICS'09*，第 143–148 页。ACM，2009年。26. A. Howes、X. Chen、A. Acharya 和 R. L. Lewis。交互作为部分可观测马尔可夫决策过程的涌现属性。载于 A. Oulasvirta、P. O. Kristensson、X. Bi 和 A. Howes（编），*Interaction* 10, 287 310 2018 *Computational* ，章节 页 – 。牛津学术出版社，。27. S. S. Intille、J. Rondoni、C. Kukla、I. Ancona 和 L. Bao。一种情境感知体验抽样工具。载于 G. Cockton 和 P. Korhonen（编），*(CHI'03)* 972 973 2003 *Extended Abstracts of the 2003 Conference on Human Factors in Computing Systems* ，页 – 。ACM，。28. G. F. Jenks。统计制图中的数据模型概念。 *International Yearbook of Cartography 7*，第 186–190 页，1967年。29. S. L. Jones、D. Ferreira、S. Hosio、J. Gonçalves 和 V. Kostakos。智能手机应用使用的重访分析。载于 K. Mase、M. Langheinrich、D. Gatica-Perez、H. Gellersen、T. Choudhury 和 K. Yatani（编），*Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing (UbiComp'15)* 2015 。ACM，。30. V. Kostakos、D. Ferreira、J. Gonçalves 和 S. Hosio。智能手机使用建模：马尔可夫状态转移模型。载于 P. Lukowicz、A. Kr¨uger、A. Bulling、Y. Lim 和 S. N. Patel（编），，页 – 。 *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing (UbiComp'16)* 486 497, 2016 31. M. Z. Kwiatkowska、G. Norman 和 D. Parker。PRISM 4.0：概率实时系统的验证。载于 G. Gopalakrishnan 和 S. Qadeer（编）， *Proc. of CAV'11*， *LNCS*第 6806 卷，第 585–591 页。Springer，2011年。32. I. Li、A. Dey 和 J. Forlizzi。个人信息学系统的阶段式模型。载于 E. D. Mynatt、D. Schoner、G. Fitzpatrick、S. E. Hudson、W. K. Edwards 和 T. Rodden（编），*Proceedings of the 28th International Conference on Human Factors in Computing Systems (CHI'10)* 557 566 2010 ，页 – 。ACM，。33. L. Lu、M. H. Dunham 和 Y. Meng。从点击流数据中挖掘显著使用模式。载于 O. Nasraoui、O. R. Zaïane、M. Spiliopoulou、B. Mobasher、B. M. Masand 和 P. S. Yu（编），，页 – 。 *Proceedings of the 7th International Workshop on Knowledge Discovery on the Web (WebKDD 2005)* 1 17 2005 Springer，。34. K. S. Luckow 和 C. S. Pasareanu。Log model：从日志数据中推断行为模型。载于，页 – 。 *Proceedings of the 18th IEEE International High-Level Design Validation and Test Workshop (HLDVT'16)* 25 29 2016 IEEE，。

35. D. McMillan、A. Morrison、O. Brown、M. Hall 和 M. Chalmers。《Further into the Wild：移动系统的全球试验》。载于 P. Flor´een、A. Kr¨uger 和 M. Spasojevic 编， *Proceedings of the 8th international conference on Pervasive Computing (Pervasive 2010)*，第 6030 卷， *Lecture Notes in Computer Science*，第 210–227 页。Springer，2010年。36. A. L. Montgomery、S. Li、K. Srinivasan 和 J. C. Liechty。利用点击流数据对在线浏览和路径进行建模分析。 *Marketing science*，23(4)：579–595，2004年。37. A. Morrison、X. Xiong、M. Higgs、M. Bell 和 M. Chalmers。iPhone 应用启动行为的大规模研究。载于 *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*，CHI '18，第 344:1–344:13 页，纽约，NY，美国，2018年。ACM。38. K. P. Murphy。 *Machine Learning: A Probabilistic Perspective*。麻省理工学院出版社，2012年。39. A. Oulasvirta、T. Rattenbury、L. Ma 和 E. Raita。习惯使智能手机使用更加普遍。*Personal and Ubiquitous Computing* 16 1 105 114, 2012 ，( )：–。40. J. Rooksby、P. Asadzadeh、A. Morrison、C. McCallum、C. Gray 和 M. Chalmers。落实移动应用部署的伦理规范。载于 H. B. L. Duh、C. Lueg、M. Billinghurst 和 W. Huang 编, *Proceedings of the 28th Australian Conference on Computer-Human Interaction (OzCHI'16)* 16, 406 415 2016 ，OzCHI '，第 – 页。ACM，。41. P. Singer、D. Helic、A. Hotho 和 M. Strohmaier。HypTrails：一种用于比较网络人类轨迹相关假设的贝叶斯方法。载于 A. Gangemi、S. Leonardi 和 A. Panconesi 编, *Proceedings of the 24th International Conference on World Wide Web WWW 2015* 1003 1013 2015 ， ，第 – 页。ACM，。42. P. Singer、D. Helic、B. Taraghi 和 M. Strohmaier。使用不同阶数的马尔可夫链模型检测人类导航模式中的记忆与结构。 *PLOS ONE*，9(7)：1–21，2014年。43. J. Stragier、G. Vandewiele、P. Coppens、F. Ongenae、W. Van den Broeck、F. De Turck 和 L. De Marez。移动健康应用开发中的数据挖掘：通过马尔可夫链分析评估应用内导航。*J. Med. Internet Res.*，21(6)，2019年6月。44. P. Tchounikine。面向挪用的设计：理论阐述。 *Human–Computer Interaction*，32 195，2017 4 155( )：–。45. H. W. Thimbleby、P. A. Cairns 和 M. Jones。使用马尔可夫模型进行可用性分析。 *ACM Trans. Comput.-Hum.* ，( )：–。46. D. *Interact.* 8 2 99 132, 2001
R. Thomas。分析定性评估数据的一般归纳法。 *American Journal of Evaluation* 27 2 237 246, 2006 ，( )：–。47. V. Tran、D. Maxwell、N. Fuhr 和 L. Azzopardi。使用马尔可夫链进行个性化搜索时间预测。载于 J. Kamps、E. Kanoulas、M. de Rijke、H. Fang 和 E. Yilmaz 编, *Proceedings of the ACM SIGIR International Conference on Theory of Information Retrieval (ICTIR'17)* 237 240 2017 ，第 – 页。ACM，。48. J. Wang、J. Sun、Q. Yuan 和 J. Pang。我们应该学习用于模型检验的概率模型吗？一种新方法与实证研究。载于 M. Huisman 和 J. Rubin 编， *Proceedings of the 20th International Conference on Fundamental Approaches to Software Engineering (FASE'17)*，第 10202 卷， *Lecture Notes in Computer Science*，第 3–21 页。Springer，2017年。49. L. Welch。隐马尔可夫模型与鲍姆–韦尔奇算法。 *IEEE Information Theory Society Newsletter*，2003年12月。50. J. A. Whittaker 和 J. H. Poore。软件规范的马尔可夫分析。 *ACM Trans. Softw. Eng. Methodol.*，2 1 93 106，1993年 ( )：–。51. G. Wolf。Know Thyself：全天候追踪生活的各个方面，从睡眠到情绪再到疼痛。https://www.wired.com/2009/06/lbnp-knowthyself/，2009年。访问日期：2021-06-01。52. S. Zhao、J. Ramos、J. Tao、Z. Jiang、S. Li、Z. Wu、G. Pan 和 A. K. Dey。通过应用使用行为发现不同类型的智能手机用户。载于 *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing (UbiComp'16)* 498 509 2016 ，第 – 页。ACM，。

A AppTracker 状态

AppTracker 1 中的状态如下：

– 摘要状态：
- 总体使用情况：所有记录的数据摘要，
- 统计：应用使用情况统计，
- 总体图表：从总体使用情况中选择时，显示详细的应用使用情况，
- 统计图表：从“统计”中选择时显示应用使用详情，
- 最近7天：最近七天使用次数最多的五个应用，– 具体状态：
- 选择时段：所选时间段的统计数据，
- AppsInPeriod：所选时间段内使用的应用，
- 时段内应用图表：从 AppsInPeriod 中选择时显示应用使用详情，
- 最近7天：最近七天使用次数最多的五个应用，– 会话相关状态：
- UseStart：会话开始（启动 AppTracker1 或将其切换至前台），
- UseStop：会话结束（关闭 AppTracker1 或将其置于后台），– 其他状态：

- T&C：条款与条件页面，
- 主界面：主菜单屏幕，
- 设置：设置选项，
- 反馈：提供反馈的屏幕，
- 信息：有关应用的信息，
- 任务：从反馈中选择的反馈问题。

AppTracker 2 中新增的状态如下：

– OverallAll：自安装 AppTracker2 以来的设备使用情况汇总统计数据，– ExploreData：用于深入探索所有已记录数据的三种选项，– 今日应用：当天应用使用情况摘要，– 按时段查看应用：按所选时间段查看各种应用的使用统计，– 时段内应用图表：从“按时段查看应用”中选中后查看详细的应用使用情况，– 按应用查看总览：所选应用的使用情况摘要，– 按应用查看总览图表：从“按应用查看总览”中选中后查看详细的应用使用情况。

B 概率时序逻辑

概率计算树逻辑（PCTL）及其扩展 PCTL* 是一种逻辑，可用于表达离散时间马尔可夫模型的某个状态满足某项时序属性的概率测度。其语法如下：

$$\text{状态公式 } \Phi ::= \mathit{true} \mid a \mid \neg\Phi \mid \Phi \wedge \Phi \mid \mathsf{P}_{\bowtie p}[\Psi] \mid \mathsf{S}_{\bowtie p}[\Phi] \quad \mathit{PCTL} \text{ 路径公式 } \Psi ::= \mathsf{X}\Phi \mid \Phi \mathsf{U}^{\leq N} \Phi$$

$$\mathit{PCTL*} \text{ 路径公式 } \Psi ::= \Phi \mid \Psi \wedge \Psi \mid \neg\Psi \mid \mathsf{X}\,\Psi \mid \Psi \mathsf{U}^{\leq N} \Psi$$

其中 a 表示一个原子命题、 $\bowtie \in \{\leq, <, \geq, >\}$、 $p \in [0,1]$和 $N \in \mathbb{N} \cup \{\infty\}$。

PCTL 和 PCTL* 公式（或性质）在离散时间马尔可夫链（DTMC）的状态上进行解释，其中状态公式 Φ在状态上求值，路径公式 Ψ 在路径上求值。若模型的初始状态满足 Φ，则称离散时间马尔可夫链满足状态公式 Φ 。我们用 $s \models \Phi$ 表示状态 s 满足 Φ （或者说， Φ 在状态 s中求值为真）。其中， $s \models$ true 恒为真； $s \models a$ 当且仅当 a 是标记状态 s的原子命题； $s \models \neg\Phi$ 当且仅当 $s \models \Phi$ 为假；$s \models \Phi_1 \wedge \Phi_2$当且仅当 $s \models \Phi_1$和 $s \models \Phi_2$； $s \models \mathsf{P}_{\bowtie p}[\Psi]$ 当且仅当从状态 s 出发的路径满足 Ψ的概率达到界限 $\bowtie p$； $s \models \mathsf{S}_{\bowtie p}[\Phi]$ 当且仅当处于满足 Ψ 的状态中的稳态（长期）概率达到界限 $\bowtie p$。算子 X 和 U 分别称为下一个算子和直到算子。通俗地说，对于一条从 s 出发的路径，路径公式 $\mathsf{X}\Phi$ 为真，当且仅当路径中紧接着 s 之后的下一个状态满足 Φ ；而 $\Phi_1 \mathsf{U}^{\leq N} \Phi_2$ 在路径 ω 上为真，当且仅当 Φ_2 在 N 个时间步内得到满足，并且 Φ_1 在此之前一直为真。

上述语法仅包含一组最小算子；命题算子false、析取 $\vee$和蕴含 $\Longrightarrow$均可由此导出。两个常见的派生路径算子是：$\mathit{eventually}$ 算子 F，其中 $\mathsf{F}^{\leq n} \Phi \equiv \mathit{true}\,\mathsf{U}^{\leq n} \Phi$，以及$\mathit{always}$算子 G，其中 $\mathsf{G}\ \Psi \equiv \neg(\mathsf{F}\ \neg\Psi)$。如果 $N = \infty$，即直到算子 U 不受界限约束，则省略上标。

C AnaAppTracker1 和 AppTracker 的 GPAM(3)模型的分析结果 2

与 GPAM(2) 的归纳式编码过程类似，我们使用表 1 中的所有状态，为 GPAM(3)模型实例化概率时序属性 VisitProbInit、VisitCountInit 和 StepCountInit。对使用状态 OverallUsage、最近7天、选择时段、统计和 AppsInPeriod，以及会话次数和长度实例化的这些属性进行模型检验，其结果见表 6。

表7展示了归纳式编码过程中的具体操作，表8则展示了活动模式标签。同样，会话长度与频率之间存在很强的相关性，即少/长和多/短。然而，我们还发现了另一组状态，并将其称为 TopLevelMenu，该组由从主菜单出发，通过一至两次按钮点击即可到达的状态组成。总体而言，与 GPAM(2) 相比，GPAM(3) 的会话类别划分更加细致。我们再次注意到早期使用与熟练用户使用之间的差异，不过这些差异不像 GPAM(2) 中那么明显。在第二个月和第三个月，我们观察到更多的浏览和聚焦行为，而没有出现浏览行为。在 GPAM(3) 的第二个月使用期间，我们注意到有两种模式具有相同的标签 GlancingSpecific：它们都抽象出相似的浏览行为，但各自围绕略有不同的主要状态列表展开，即 $\{4, 6, 5\}$ 和 $\{4, 5\}$。

AppTracker2 的 GPAM(3)模型分析结果列于表 9：会话数量和会话长度的类别、基于会话特征和主要状态对活动模式的初步分类，以及使用强度；这些结果与 AppTracker1 的类似。与之前一样，“–”

表 6. AppTracker1. GPAM(3). 属性分析 `VisitProbInit`，`VisitCountInit`，`StepCountInit` 用于总体使用情况，最近7天，选择时段，统计和`AppsInPeriod`，随后是会话的特征（属性 `VisitProbInit` 用于 `UseStop`，`SessionCount` 和 `SessionLength`）用于 $N = 50$。 蓝色表示最佳结果，紫色表示中间结果，橙色表示最差结果

属性。	Time	OverallUsage			Last7Days			SelectPeriod			Stats			AppsInPeriod		
	int.	AP1	AP2	AP3	AP1	AP2	AP3	AP1	AP2	AP3	AP1	AP2	AP3	AP1	AP2	AP3
VisitProbInit	[0,1]	0.99	0.78	0.99	0.52	0.91	0.98	0.01	0.90	0.29	0.99	0.85	0.66	0.03	0.00	0.01
	[0,7]	0.99	0.62	0.98	0.27	0.69	0.98	0.08	0.79	0.94	0.77	0.45	0.82	0.02	0.62	0.53
	[0,30]	0.99	0.59	0.99	0.23	0.60	0.99	0.07	0.65	0.98	0.16	0.36	0.80	0.04	0.53	0.71
	[30,60]	0.99	0.44	0.65	0.78	0.99	0.98	0.19	0.71	0.97	0.94	0.00	0.25	0.11	0.57	0.00
	[60,90]	0.99	0.90	0.76	0.40	0.99	0.74	0.11	0.37	0.99	0.95	0.55	0.40	0.09	0.33	0.88
VisitCountInit	[0,1]	18.89	5.44	1.80	0.73	3.76	2.31	0.01	0.34	2.18	7.39	1.79	1.84	0.33	0.01	0.00
	[0,7]	20.47	1.20	6.05	0.33	1.13	5.24	0.12	2.11	2.72	1.96	0.60	2.15	0.04	1.99	5.04
	[0,30]	22.66	1.12	5.37	0.27	0.89	5.94	0.10	1.80	3.48	0.26	0.45	1.92	0.06	2.88	4.07
	[30,60]	16.75	0.76	1.12	1.57	9.38	3.76	0.23	1.35	3.49	4.11	0.00	0.35	1.31	6.85	0.00
	[60,90]	12.84	2.79	4.00	0.50	8.16	1.58	0.51	0.44	5.59	4.10	0.89	1.25	0.43	3.85	7.57
StepCountInit	[0,1]	<escape>4.72 6.04 32.91 68.57 13.63 20.50 3350.93 148.3121.22</escape>									9.70	46.07	26.09	1415.69	—	—
	[0,7]	—	51.43	7.45	—	42.73	10.20	—	32.43	18.05	—	<escape>82.57 27.66</escape>		—	51.56	64.83
	[0,30]	5.01	55.98	8.59	247.66	55.63	8.68	1153.58	47.65	13.44	295.84	111.09	29.52	2670.48	65.95	40.51
	[30,60]	4.05	88.28	47.26	44.94	4.09	12.66	223.72	39.81	14.48	21.93	—	172.74	403.76	59.48	—
	[60,90]	2.27	21.54	34.66	98.96	5.05	36.49	416.06	106.05	8.39	<escape>16.21 63.78 96.67</escape>			489.26	123.08	22.93

时间	VisitProbInit 会话次数						SessionLength	
间隔	AP1	AP2	AP3	AP1	AP2	AP3	AP1 AP2	AP3
[0, 1]	0.16	0.99	0.02	0.18	12.36	0.02	376.93 2.95	2469.47
[0, 7]	0.27	0.99	0.89	0.32	11.91	1.82	— 3.12	24.40
[0, 30]	0.30	0.99	0.97	0.34	11.74	2.54	146.32 3.25	17.16
[30, 60]	0.70	0.99	0.99	1.22	8.83	8.58	57.66 4.71	4.68
[60, 90]	0.99	0.99	0.96	8.56	8.75	3.57	4.65 4.65	13.47

表示到达 UseStop 状态的概率小于 1，因此累积奖励（步数）为无穷大。GPAM(3)活动模式标签见表 10。由于浏览行为比 AppTracker1 中更常见，为了突出以一个或两个状态为中心的模式之间的差异，Focussing（总体全部）这一记法表示一种仅以 OverallAll 状态为中心的聚焦摘要。总之，在 AppTracker2 的 GPAM(3)模型中，我们发现了以下六种不同的活动模式（而 AppTracker1 中有四种）：聚焦摘要、FocussingSpecificByPeriod、GlancingSummary、聚焦（总体全部）、浏览（主要）和浏览摘要。

对于 GPAM(3)模型（见图 9(a)），对于早期用户而言，特定浏览一瞥几乎总是会转为聚焦摘要（处于统计状态时除外，此时概率较低；图 9(a) 未标示这一点，因为该图给出的是平均值）。在第二个月，用户很可能会在 GlancingSpecific$_3$（以最近7天和选择时段为中心）与聚焦摘要之间转移。在使用的第三个月，与特定浏览一瞥相比，从 GlancingSummary 或 FocussingSpecific 转移到另一种活动模式的概率非常低。

图10展示了 AppTracker1 和 AppTracker2 的不同 GPAM(3)模型的 LongRunPattern 属性结果。在早期使用模型中，与 BrowsingSummary 和 FocussingSpecific 相比，FocussingSummary 和 GlancingSpecific 出现的概率相同且占主导地位。第二个月，以 Last7Days 和 SelectPeriod 为中心的 GlancingSpecific 状态概率，高于以 Last7Days、AppsInPeriod 和 SelectPeriod 为中心的 GlancingSpecific 状态（分别为 0.38 和 0.18）；不过，在 Last7Days 和 SelectPeriod 子菜单中的总体浏览行为，比以摘要状态 OverallUsage 和 Stats（通常用于探索）为中心的聚焦行为更为常见。第三个月，FocussingSpecific 占据主导地位，与 GPAM(2) 一致。不过，我们首次看到 OverallUsage 和 Stats 之间出现浏览行为，这可能表明，尽管会话时间可能很短，但用户的操作并非探索性的，而是体现了更明确的意图。我们由此得出结论：对于熟练用户而言，查看特定状态会逐渐成为一种更具聚焦性的行为。

表 7. AppTracker1. GPAM(3)。用于标记活动模式的归纳式编码：对平均会话次数和长度进行分类，随后将其与主要状态一起用于活动模式的初步分类，并描述使用强度。会话长度“–”表示会话结束的可达概率小于 1（因此长度为无限）。状态标识符如下：3 为总体使用情况，4 为最近7天，5 为选择时段，6 为 AppsInPeriod，9 为统计，10 为总体图表，12 为统计图表，14 为时段内应用图表。状态标识符以摘要状态和特定状态突出显示。

会话次数类别			
极少	**few**	**mid**	**many**
{0.02, 0.18, 0.32, 0.34, 1.22}	{1.82, 2.54, 3.57}	{8.56, 8.58, 8.75, 8.83}	{11.74, 12.36}

会话时长类别					
极短	**短**	**mid**	**long**	**v.长**	（不适用）
{2.95, 3.12, 3.25}	{4.65, 4.68, 4.71}	{13.47}	{17.16, 24.40, 57.66}	{146.32, 376.93, 2469.47}	–

Time int.	AP1		AP2		AP3	
	会话	状态 ID	会话	状态 ID	会话	状态 ID
[0, 1]	v.少/v.长	3、 1	多/非常短	4 , 9	v.少/v.长	3 , 9
[0, 7]	v.少/–	3 ,	多/v.short 5 , (	3 4	少/长	3, 6
[0, 30]	v.少/v.长	3 ,	多/非常短	6、	少/长	4, 6
[30, 60]	少/长	3 , 1	中/短	4 , 5	中/短	4,
[60, 90]	中/短	3, 1	中/短	4,	少/中	6,

使用强度 使用强度特征

浏览以至少三个顶层菜单状态为中心的极少/极长或少/长会话（即从主菜单只需点击一到两次按钮即可到达），有时涉及不同状态组；浏览以一两个状态为中心的多/极短或中/短会话；聚焦以同一组中的状态为中心的中/中、少/中、少/长、极少/–或极少/极长会话

表8。 AppTracker1。GPAM(3)模型中的活动模式标签。

时间间隔 AP1	AP2	AP3
[0, 1]	聚焦摘要 特定浏览一瞥	浏览摘要
[0, 7]	聚焦摘要 特定浏览一瞥	浏览顶层菜单
[0, 30]	聚焦摘要 特定浏览一瞥	浏览顶层菜单
[30, 60]	聚焦摘要 特定浏览一瞥	特定浏览一瞥
[60, 90]	GlancingSummary	特定浏览一瞥 FocussingSpecific

表 9. AppTracker2。GPAM(3)。平均会话次数和时长的分类，随后用于对活动模式进行初步分类，并结合主要状态描述使用强度标签。状态标识符（id）如下：1 为主界面，5 为 OverallAll，6 为 ExploreData，8 为今日应用，9 为选择时段，10 为统计，11 为按时段查看应用，12 为时段内应用图表，13 为按应用查看总览，14 为统计图表，15 为按应用查看总览图表。状态标识符以摘要状态和特定状态突出显示。

会话次数类别			
极少	**few**	**mid**	**many**
$\{0.06, 0.09\}$	$\{1.62, 2.24, 2.77, 2.95, 3.56\}$	$\{5.56, 6.69, 6.90, 7.47, 8.21, 8.78, 9.20\}$	$\{14.80\}$

会话时长类别				
极短	短大发彩票	**mid**	**long**	（不适用）–
$\{2.32\}$	$\{4.25, 4.97\}$	$\{5.62, 6.49, 6.53, 7.02, 7.86, 12.29\}$	$\{18.32, 23.14, 31.37\}$	

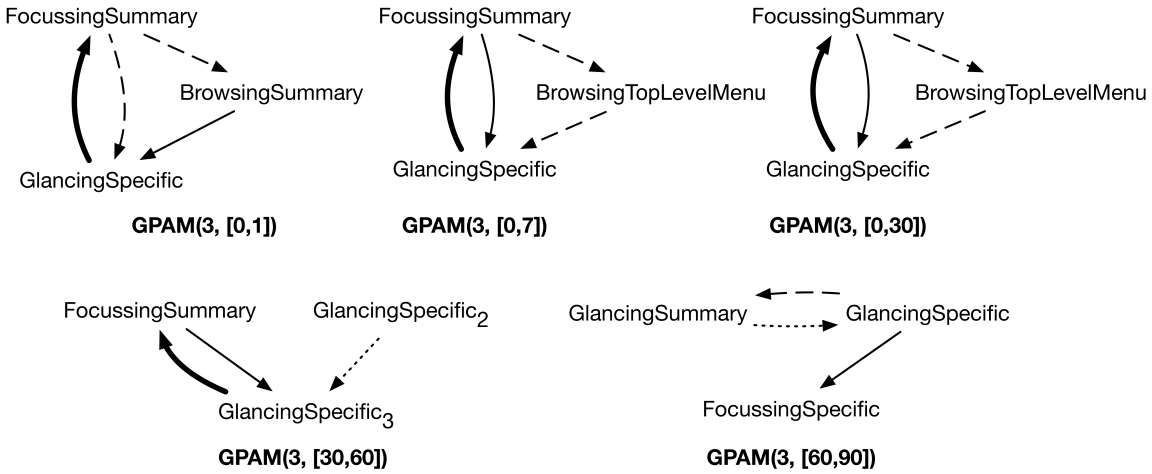
Time int.	**AP1**		**AP2**		**AP3**	
	会话	**状态 ID**	**会话状态 ID**		**会话**	**状态 ID**
$[0, 1]$	中/中	6、	v.少/–	5	少/长	5，
$[0, 7]$	少/长	5，	极少/–	5	中/中	6，
$[0, 30]$	中/短	5，	少/长	5，	中/中	9，
$[30, 60]$	中/中	5，	中/短	5，	少/中	11，
$[60, 90]$	少/–	5，	中/中	5，	多/极短	1

使用强度 使用强度特征

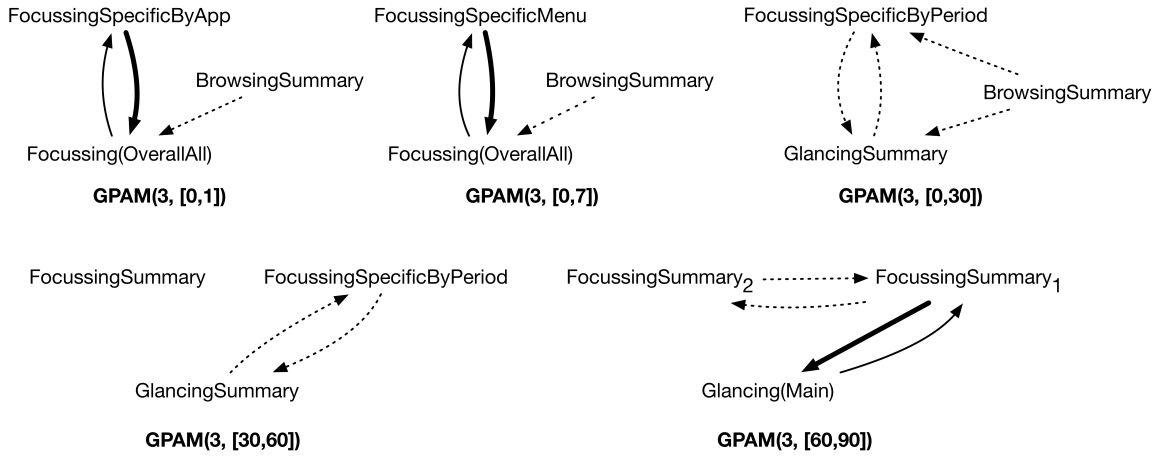
浏览以至少三个状态为中心的少/长会话，这些状态可从主菜单通过一到两次按钮点击到达（例如摘要和特定菜单状态），有时涉及不同状态组；浏览以一两个状态为中心的中/短或多/极短会话；聚焦以同一组中的状态为中心的中/中、少/中、少/–或极少/–会话。

表10. AppTracker2。GPAM(3)模型中的活动模式标签。

时间间隔	**AP1**	**AP2**	**AP3**
$[0, 1]$	按应用聚焦特定内容 聚焦	（总体全部） 浏览摘要	
$[0, 7]$	浏览摘要	聚焦（总体全部） 聚焦特定菜单	
$[0, 30]$	GlancingSummary	浏览摘要	FocussingSpecificByPeriod
$[30, 60]$	聚焦摘要	GlancingSummary	FocussingSpecificByPeriod
$[60, 90]$	聚焦摘要	聚焦摘要	浏览（主要）

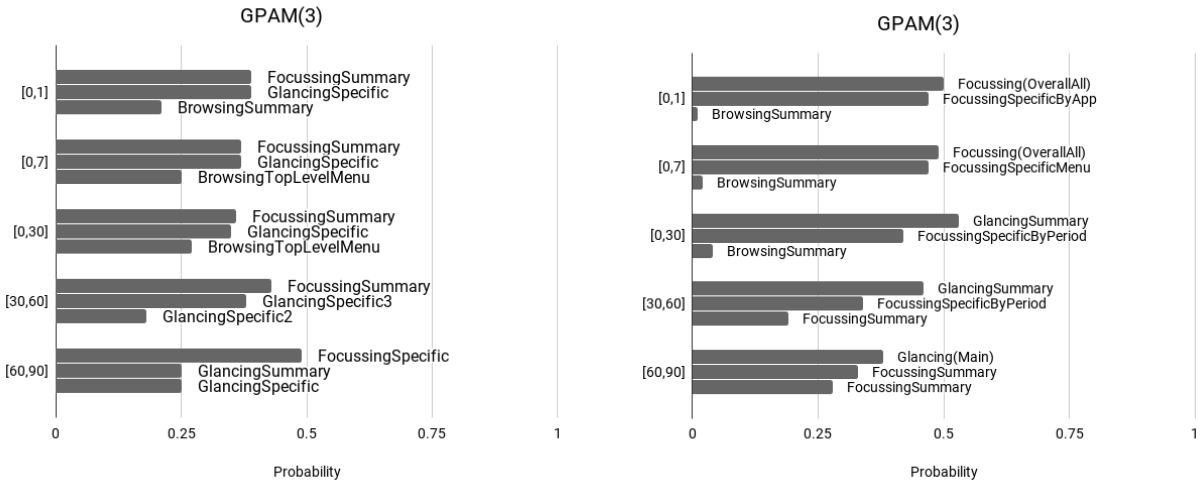


(a) AppTracker1



(b) AppTracker2

图 9. AppTracker1 和 AppTracker2。StateToPattern 的可视化：在不同时间区间的 GPAM(3)模型中，会话内从观测到的状态切换活动模式的平均似然。除 Stats 和统计图表状态的该属性返回的概率低于平均值外，所有状态返回的概率都非常相似。点状箭头表示 $[0.02, 0.2)$ 中的概率，虚线箭头表示 (0.2,0.5) 中的概率，细箭头表示 $[0.5, 0.8)$ 中的概率，粗箭头表示 ≥ 0.8。低于 0.02 的似然未予显示。对于 AppTracker1 中的 GPAM(3,[30,60])，我们用下标 2 或 3 区分两个标签相同的模式，以表示模式标识符（AP2 或 AP3）。



(a) AppTracker1 (b) AppTracker2

图 10. AppTracker1 和 AppTracker2。不同时间间隔下 GPAM(3)模型中长期处于各活动模式的概率可视化；每个时间间隔内的条目按降序排列。