(19)国家知识产权局

(12)发明专利

(10)授权公告号 CN 115061809 B
(45)授权公告日 2022.11.11

(21)申请号 202210944420.X

(22)申请日 2022.08.08

(65)同一申请的已公布的文献号
申请公布号 CN 115061809 A

(43)申请公布日 2022.09.16

(73)专利权人 杭州实在智能科技有限公司
地址 310000 浙江省杭州市余杭区余杭街道文一西路1818-2号6幢6层

(72)发明人 赵明 张军燕 孙林春

(74)专利代理机构 浙江永鼎律师事务所 33233
专利代理师 周希良

(51)Int.Cl.
G06F 9/48(2006.01)
G06F 9/455(2006.01)

(56)对比文件
US 2022129931 A1,2022.04.28
CN 113570347 A,2021.10.29
CN 113836264 A,2021.12.24
US 2020348964 A1,2020.11.05
CN 111429083 A,2020.07.17
CN 114415691 A,2022.04.29
CN 113570093 A,2021.10.29
CN 112001594 A,2020.11.27
CN 110427252 A,2019.11.08
US 10853097 B1,2020.12.01
CN 113157409 A,2021.07.23
JP 2021033998 A,2021.03.01
WO 2021238045 A1,2021.12.02
CN 114755984 A,2022.07.15
CN 112817748 A,2021.05.18
KR 102190459 B1,2020.12.11
CN 111209301 A,2020.05.29
CN 113467383 A,2021.10.01
CN 114675948 A,2022.06.28
US 2021110301 A1,2021.04.15
刘海涛 等.基于RPA+AI 的数字员工在电力.《电力信息与通信技术》.2022,第20卷(第4期),第88-92页. (续)

审查员 吴海旋

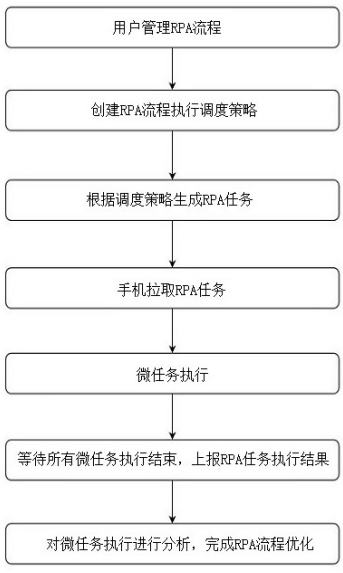
权利要求书2页 说明书8页 附图5页

(54)发明名称

基于安卓的RPA多任务调度方法及系统

(57)摘要

本发明属于计算机技术领域,具体涉及基于安卓的RPA多任务调度方法及系统。方法包括S1,用安卓RPA机器人通过网络直连RPA任务调度中心,使用户通过RPA任务调度中心对RPA流程进行管理;S2,用户创建RPA流程并执行调度策略;S3,RPA任务调度中心根据调度策略生成RPA任务;S4,手动拉取RPA任务,并在拉取到RPA任务后进行任务的依赖计算和拆解,拆解为微任务依赖树;S5,根据微任务依赖树对微任务进行最大限度的并行执行;S6,等待所有微任务执行完成后对微任务执行的结果进行上报;S7,对微任务执行结果进行分析和记录,完成RPA任务的执行流程优化处理。本发明具有执行效率高、节约成本、容易优化的特点。



CN 115061809 B

[转续页]

[接上页]

(56)对比文件

翟文正等.异构多核DAG任务模型的微粒群优化调度算法.《计算机工程与设计》.2016,(第07期),

王福军等.实时控制系统中的任务模型及调度算法设计.《石家庄经济学院学报》.2004,(第06期),

卞琛等.内存计算框架局部数据优先拉取策略.《计算机研究与发展》.2017,(第04期),

Xiuli Wu 等.Multiobjective Differential Evolution Algorithm for Solving Robotic Cell Scheduling Problem With Batch-Processing Machines.《IEEE Transactions on Automation Science and Engineering 》.2021,第18卷(第2期),第757-774页.

1.基于安卓的RPA多任务调度方法,其特征在于,包括如下步骤;

S1,用安卓RPA机器人通过网络直连RPA任务调度中心,使用户通过RPA任务调度中心对RPA流程进行管理;

S2,用户创建RPA流程并执行调度策略;

S3,RPA任务调度中心根据调度策略生成RPA任务;

S4,手动拉取RPA任务,并在拉取到RPA任务后进行任务的依赖计算和拆解,拆解为微任务依赖树;

S5,根据微任务依赖树对微任务进行最大限度的并行执行;

S6,等待所有微任务执行完成后对微任务执行的结果进行上报;

S7,对微任务执行结果进行分析和记录,完成RPA任务的执行流程优化处理;

步骤S3中,所述RPA任务包含若干个微任务;每个微任务均视为一个最小可执行单元;

每个微任务均包含任务ID、任务上级依赖和微任务元信息;

步骤S6中,每个微任务执行的过程中均记录开始时间、结束时间以及执行是否异常信息,并在RPA任务执行结束后统一上报给RPA任务调度中心;

步骤S5中所述并行执行过程具体如下:

S51,设定拉取RPA任务A和RPA任务B,根据微任务的ID、依赖和元信息,计算出两个RPA任务的微任务执行DAG图;

其中,设定任务A中的微任务3和任务B中的任务4元信息均为{ui:true},表示是一个UI独占任务;

S52,开启4个并行执行的微任务执行单元,通过步骤S51完成的微任务执行DAG图,开始执行微任务;

S53,从两个DAG图中解析出可执行的微任务A-1和B-1,分别记录A-1和B-1两个微任务的执行开始时间;

其中,A-1和B-1分别表示任务A中的微任务1和任务B中的微任务1;

S54,步骤S53中的2个微任务执行完成时,记录微任务的执行结果到全局变量中,微任务执行单元开始解析出下个可执行微任务,分别为A-2、A-3、B-2、B-3;

其中,A-2、A-3、B-2、B-3分别表示任务A中的微任务2、微任务3和任务B中的微任务2、微任务3;

S55,步骤S54中的4个微任务执行完成时,开始解析出可执行微任务A-4、B-4、B-5,并直到所有任务执行结束。

2.根据权利要求1所述的基于安卓的RPA多任务调度方法,其特征在于,步骤S6包括如下步骤:

S61,当一个RPA任务所有的微任务均执行完成时,即表示所述RPA任务已经完成执行,对所有微任务执行的结果统一上报至RPA任务调度中心。

3.基于安卓的RPA多任务调度系统,用于实现权利要求1-2任一项所述的基于安卓的RPA多任务调度方法,其特征在于,所述基于安卓的RPA多任务调度系统包括;

RPA任务调度中心模块,用于对RPA流程进行管理以及创建RPA流程并执行调度策略;

RPA任务生成模块,用于RPA任务调度中心根据调度策略生成RPA任务;

计算拆解模块,用于在拉取到RPA任务后进行任务的依赖计算和拆解,将RPA任务拆解

为微任务依赖树；

并行执行模块，用于根据微任务依赖树对微任务进行最大限度的并行执行；

结果上报模块，用于等待所有微任务执行完成后对微任务执行的结果进行上报；

分析优化模块，用于对微任务执行结果进行分析和记录，完成RPA任务的执行流程优化处理。

4.根据权利要求3所述的基于安卓的RPA多任务调度系统，其特征在于，所述并行执行模块具体如下：

设定拉取RPA任务A和RPA任务B，根据微任务的ID、依赖和元信息，计算出两个RPA任务的微任务执行DAG图；

其中，设定任务A中的微任务3和任务B中的任务4元信息均为{ui:true}，表示是一个UI独占任务；

开启4个并行执行的微任务执行单元，通过完成的微任务执行DAG图，开始执行微任务；

从两个DAG图中解析出可执行的微任务A-1和B-1，分别记录A-1和B-1两个微任务的执行开始时间；

其中，A-1和B-1分别表示任务A中的微任务1和任务B中的微任务1；

当2个微任务执行完成时，记录微任务的执行结果到全局变量中，微任务执行单元开始解析出下个可执行微任务，分别为A-2、A-3、B-2、B-3；

其中，A-2、A-3、B-2、B-3分别表示任务A中的微任务2、微任务3和任务B中的微任务2、微任务3；

当4个微任务执行完成时，开始解析出可执行微任务A-4、B-4、B-5，并直到所有任务执行结束。

5.根据权利要求3所述的基于安卓的RPA多任务调度系统，其特征在于，所述RPA任务调度中心模块包括：

RPA流程管理模块，用于管理用户RPA流程和流程调度策略；

任务队列模块，用于放置根据RPA流程和调度策略生成的RPA任务；

任务分析模块，用于分析汇总RPA任务的执行结果；

微任务分析模块，用于分析RPA任务中的微任务执行情况，包括微任务执行时间分析、执行结果分析和用户优化和处理任务结果。

基于安卓的RPA多任务调度方法及系统

技术领域

[0001] 本发明属于计算机技术领域,具体涉及基于安卓的RPA多任务调度方法及系统。

背景技术

[0002] RPA(Robotic Process Automation,简称RPA)是机器人流程自动化的简称,它是一种利用软件代替或者协助人类在计算机、手机等数字化设备中完成重复性工作与任务,替代人工完成自动化。它是一种新兴的数字化工具,可以帮助企业或员工完成重复、单调的流程性工作,减少人工错误,提高运营效率,降低运营成本。

[0003] 目前的RPA工具主要针对桌面端,并且以Windows系统为主。伴随移动互联网的发展,移动端的RPA软件需求也越来越大。如今5G时代已来临,数据流量不再成为制约手机使用场景和复杂业务处理的因素,移动端业务的突破与飞跃带来越来越多的移动端自动化场景,手机端的RPA流程也被广泛使用。手机端的RPA流程,是指借助全新的移动端设备(例如物理手机、云手机),通过RPA对手机应用软件进行自动操作(点击、滑动、图片截取、采集数据等),以满足基于手机本身的自动化场景操作和任务的执行。安卓作为一个开源的手机操作系统被各大手机厂商青睐,基于二次开发适配手机厂商的硬件设备,已经成功的占领了很大的市场。作为开源的手机操作系统,也被广泛的用于RPA流程运行的手机操作系统。

[0004] 随着RPA技术在手机端的发展,基于安卓系统的第三方RPA应用也越来越多。使用开源软件Appium、uiautomator2二次开发实现的安卓端的RPA软件被广泛使用。这些软件拥有着易用、稳定性高、开发简单的特点,能实现简单培训即可快速上手。目前已知的安卓RPA流程将不在局限于基于用户界面(User Interface,简称UI)的操作,还包含数据清洗、数据分析、报表生成、文档处理、预警告警等功能。

[0005] 近几年安卓端RPA流程的日益复杂化,在现实场景中运行的RPA任务,通常一个安卓端需要执行多个流程。流程在执行过程中会生成多个子任务,子任务执行过程中会有相互的依赖关系,有的子任务需要独占操作系统的,有的子任务只需要进行后台的数据处理。伴随着碎片化、轻量级的手机RPA任务越来越被广泛的运行,任务调度在安卓RPA流程中成为了一个不和或缺的部分。

[0006] 目前的安卓端RPA流程调度广泛采用的是以RPA任务调度中心统一调度,电脑端和手机端的配合的执行过程。流程执行时会以独占的方式占用电脑端和手机端的计算资源。在多任务运行时,所有任务排队进行。每个任务调度完成后,任务的执行状态(包括是否成功、任务执行时间、任务执行日志等信息)会上报至RPA任务调度中心。RPA任务调度中心会记录每个机器人的执行情况,以报表的方式呈现出任务执行成功率、执行效率等信息。目前这种RPA流程执行调度方式,过度依赖RPA任务调度中心。RPA安卓手机机器人的启动或停止都需要任务调度中心,且调度的效率往往也不高。

[0007] 现有RPA安卓端多任务执行调度技术,在任务下发和执行、执行方式和执行过程,会存在以下的一些缺点。

[0008] 1.传统RPA任务执行重度依赖电脑端RPA机器人。对于轻量级的安卓端RPA任务和

数据处理分析任务，原本只需要安卓手机端就可以很好的完成。现阶段的安卓端RPA机器人方案无任务调度相关功能模块，因此不能独立完成RPA任务的调度执行。另外从系统的可靠性指标分析，引入的模块越多，造成系统不稳定的因素越多。另外，电脑端RPA机器人和安卓端RPA机器人的通讯通常是通过USB接口或无线调试模式来实现，在实际执行RPA流程的过程中无疑给业务增加了不稳定的风险点。

[0009] 2.安卓端RPA执行效率不高。现有的电脑RPA机器人和安卓RPA机器人在调度执行的过程中是串行同步运行的，在任一时间只会有某个计算资源在执行RPA操作。从执行时间指标分析，串行的执行方式会有空闲的等待时间，无疑会增加整体RPA执行的时间。现代化的CPU设计都是经过了多线程并行计算优化，在并行计算方面都有很大的优势。作为一种优秀的调度技术，计算资源的利用率也是评价一种调度技术是否先进的度量方式。

[0010] 3.现有的调度执行方式会增加RPA执行成本。对于一个轻量手机RPA流程，只需要一台手机+RPA任务调度中心即可完成，而以现有的调度方案至少需要一台计算机和一部安卓手机共同来完成，因此无论是从资源消耗上分析是不够合理的。从经济效益上分析，手机RPA机器人和电脑RPA机器人的组合会带来更高的RPA执行成本。

[0011] 4.RPA流程执行过程分析优化不易。现有的安卓端RPA任务执行通常整体作为一个数据分析和统计最小单位，在执行过程中记录流程执行的时间、日志、执行结果，执行结束后将执行结果回传到RPA任务调度中心来记录任务的执行情况。RPA任务调度中心会根据任务执行结果进行时间维度、结果维度、效率维度做执行过程的分析，找到执行性能瓶颈所在从而进一步优化RPA执行流程。对于流程较为复杂的RPA任务，优化分析是一个不容忽视的工作，然而现有方案将手机端RPA任务作为一个整体进行任务分析，从可执行角度可操作性难度较大。

[0012] 因此，设计一种执行效率高、节约成本、容易优化的基于安卓的RPA多任务调度方法及系统，就显得十分重要。

[0013] 例如，申请号为CN202110075707.9的中国专利文献描述的一种基于安卓虚拟机处理任务的方法、计算机设备，该方法由计算机设备执行，计算机设备上配置有多个虚拟机，虚拟机是基于安卓系统创建的，每个虚拟机上配置有第一应用与第一程序，第一程序被虚拟机执行时第一应用会检测到与第一程序相关联的模拟用户的操作。方法包括：在接收到多个待处理任务时，确定第一虚拟机，将第一任务分配至第一虚拟机，调用第一虚拟机处理第一任务。虽然提供的方法，由于多个虚拟机可以同时运行，可以实现在同一时间并行处理多个待处理任务，从而提高RPA任务的处理效率，但是其缺点在于，但仍然存在增加RPA执行成本以及RPA流程执行过程分析优化不易的问题。

发明内容

[0014] 本发明是为了克服现有技术中，现有RPA安卓端多任务执行调度技术，存在依赖性强、执行效率低、增加RPA执行成本以及RPA流程执行过程分析优化不易的问题，提供了一种执行效率高、节约成本、容易优化的基于安卓的RPA多任务调度方法及系统。

[0015] 为了达到上述发明目的，本发明采用以下技术方案：

[0016] 基于安卓的RPA多任务调度方法，包括如下步骤；

[0017] S1，用安卓RPA机器人通过网络直连RPA任务调度中心，使用户通过RPA任务调度中

心对RPA流程进行管理；

[0018] S2,用户创建RPA流程并执行调度策略；

[0019] S3,RPA任务调度中心根据调度策略生成RPA任务；

[0020] S4,手动拉取RPA任务,并在拉取到RPA任务后进行任务的依赖计算和拆解,拆解为微任务依赖树；

[0021] S5,根据微任务依赖树对微任务进行最大限度的并行执行；

[0022] S6,等待所有微任务执行完成后对微任务执行的结果进行上报；

[0023] S7,对微任务执行结果进行分析和记录,完成RPA任务的执行流程优化处理。

[0024] 作为优选,步骤S3中,所述RPA任务包含若干个微任务;每个微任务均视为一个最小可执行单元。

[0025] 作为优选,每个微任务均包含任务ID、任务上级依赖和微任务元信息。

[0026] 作为优选,步骤S6中,每个微任务执行的过程中均记录开始时间、结束时间以及执行是否异常信息,并在RPA任务执行结束后统一上报给RPA任务调度中心。

[0027] 作为优选,步骤S5中所述并行执行过程具体如下：

[0028] S51,设定拉取RPA任务A和RPA任务B,根据微任务的ID、依赖和元信息,计算出两个RPA任务的微任务执行DAG图；

[0029] 其中,设定任务A中的微任务3和任务B中的任务4元信息均为{ui:true},表示是一个UI独占任务；

[0030] S52,开启4个并行执行的微任务执行单元,通过步骤S51完成的微任务执行DAG图,开始执行微任务；

[0031] S53,从两个DAG图中解析出可执行的微任务A-1和B-1,分别记录A-1和B-1两个微任务的执行开始时间；

[0032] 其中,A-1和B-1分别表示任务A中的微任务1和任务B中的微任务1；

[0033] S54,步骤S53中的2个微任务执行完成时,记录微任务的执行结果到全局变量中,微任务执行单元开始解析出下个可执行微任务,分别为A-2、A-3、B-2、B-3；

[0034] 其中,A-2、A-3、B-2、B-3分别表示任务A中的微任务2、微任务3和任务B中的微任务2、微任务3；

[0035] S55,步骤S54中的4个微任务执行完成时,开始解析出可执行微任务A-4、B-4、B-5,并直到所有任务执行结束。

[0036] 作为优选,步骤S6包括如下步骤：

[0037] S61,当一个RPA任务所有的微任务均执行完成时,即表示所述RPA任务已经完成执行,对所有微任务执行的结果统一上报至RPA任务调度中心。

[0038] 本发明还提供了基于安卓的RPA多任务调度系统,包括；

[0039] RPA任务调度中心模块,用于对RPA流程进行管理以及创建RPA流程并执行调度策略；

[0040] RPA任务生成模块,用于RPA任务调度中心根据调度策略生成RPA任务；

[0041] 计算拆解模块,用于在拉取到RPA任务后进行任务的依赖计算和拆解,将RPA任务拆解为微任务依赖树；

[0042] 并行执行模块,用于根据微任务依赖树对微任务进行最大限度的并行执行；

[0043] 结果上报模块，用于等待所有微任务执行完成后对微任务执行的结果进行上报；

[0044] 分析优化模块，用于对微任务执行结果进行分析和记录，完成RPA任务的执行流程优化处理。

[0045] 作为优选，所述并行执行模块具体如下：

[0046] 设定拉取RPA任务A和RPA任务B，根据微任务的ID、依赖和元信息，计算出两个RPA任务的微任务执行DAG图；

[0047] 其中，设定任务A中的微任务3和任务B中的任务4元信息均为{ui:true}，表示是一个UI独占任务；

[0048] 开启4个并行执行的微任务执行单元，通过完成的微任务执行DAG图，开始执行微任务；

[0049] 从两个DAG图中解析出可执行的微任务A-1和B-1，分别记录A-1和B-1两个微任务的执行开始时间；

[0050] 其中，A-1和B-1分别表示任务A中的微任务1和任务B中的微任务1；

[0051] 当2个微任务执行完成时，记录微任务的执行结果到全局变量中，微任务执行单元开始解析出下个可执行微任务，分别为A-2、A-3、B-2、B-3；

[0052] 其中，A-2、A-3、B-2、B-3分别表示任务A中的微任务2、微任务3和任务B中的微任务2、微任务3；

[0053] 当4个微任务执行完成时，开始解析出可执行微任务A-4、B-4、B-5，并直到所有任务执行结束。

[0054] 作为优选，所述RPA任务调度中心模块包括：

[0055] RPA流程管理模块，用于管理用户RPA流程和流程调度策略；

[0056] 任务队列模块，用于放置根据RPA流程和调度策略生成的RPA任务；

[0057] 任务分析模块，用于分析汇总RPA任务的执行结果；

[0058] 微任务分析模块，用于分析RPA任务中的微任务执行情况，包括微任务执行时间分析、执行结果分析和用户优化和处理任务结果。

[0059] 本发明与现有技术相比，有益效果是：(1)本发明独创性的使用了安卓手机直连RPA任务调度中心实现了一种基于安卓的RPA多任务调度系统，通过规范RPA任务定义，提出了RPA最小可执行单元微任务，综合考虑使用上的经济成本、执行效率，能够解决现有的的安卓RPA多任务调度方案的不足；(2)本发明打破了现有的电脑RPA机器人控制安卓RPA机器人的常规方案，引入了更少的子系统来解决RPA多任务调度的问题，在系统的可靠性上有一定的提升；(3)本发明在执行效率上，摒弃了传统按照指令串行发送至安卓RPA机器人，执行完成后通过USB或无线调试返回结果的冗长过程，将手机RPA任务完全独立的运行的安卓手机上，提升了执行效率和性能；(3)本发明使用纯安卓的RPA机器人，可以为用户在部署移动RPA流程时节省更多的经济成本，只需一台安卓手机和云端RPA任务调度中心即可完成手机RPA流程的快速部署和调度，特别在大批量的手机RPA机器人方面，效果显著；(4)本发明具有调度效率高、执行速度快、经济成本低，设计更合理的特点。

附图说明

[0060] 图1为本发明中基于安卓的RPA多任务调度方法的一种流程图；

[0061] 图2为本发明中基于安卓的RPA多任务调度系统的一种系统功能构架图；

[0062] 图3为本发明中RPA任务的一种构架示意图；

[0063] 图4为本发明中示例的两个RPA任务的一种微任务执行DAG图；

[0064] 图5为本发明实施例所提供的用户评论舆情监控系统的一种流程图。

具体实施方式

[0065] 为了更清楚地说明本发明实施例，下面将对照附图说明本发明的具体实施方式。显而易见地，下面描述中的附图仅仅是本发明的一些实施例，对于本领域普通技术人员来讲，在不付出创造性劳动的前提下，还可以根据这些附图获得其他的附图，并获得其他的实施方式。

[0066] 实施例：

[0067] 如图1所示的基于安卓的RPA多任务调度方法，包括如下步骤；

[0068] S1，用安卓RPA机器人通过网络直连RPA任务调度中心，使用户通过RPA任务调度中心对RPA流程进行管理；

[0069] S2，用户创建RPA流程并执行调度策略；

[0070] S3，RPA任务调度中心根据调度策略生成RPA任务；

[0071] S4，手动拉取RPA任务，并在拉取到RPA任务后进行任务的依赖计算和拆解，拆解为微任务依赖树；

[0072] S5，根据微任务依赖树对微任务进行最大限度的并行执行；

[0073] S6，等待所有微任务执行完成后对微任务执行的结果进行上报；

[0074] S7，对微任务执行结果进行分析和记录，完成RPA任务的执行流程优化处理。

[0075] 在任务调度层面，本发明优化了传统的电脑RPA机器人作为主体调度对象，安卓RPA机器人作为任务执行单元的执行方式。取而代之使用安卓RPA机器人通过网络直连RPA任务调度中心，通过任务拉取、任务依赖计算、任务拆解、并行运行，从而达到最高效的执行方式。

[0076] 在任务规范和执行层面，对传统的RPA任务进行了更细粒度的分解。用户通过调度中心对RPA流程进行管理，创建RPA流程执行调度策略。RPA调度中心根据调度策略生成RPA任务，机器人拉取到RPA任务后进行任务的依赖计算和拆解，拆分为微任务依赖树，随后安卓微任务执行单元根据微任务依赖树对微任务进行最大限度的并行执行，等待所有微任务执行完成后对任务执行的结果进行上报。对于任务执行过程中微任务执行结果的分析和记录，用于后期对完成RPA任务的执行优化处理。

[0077] 在RPA组件执行层面，抛弃了常规的电脑RPA机器人串行发送指令到安卓RPA机器人完成指令的执行。通常情况下，RPA的流程会有大量的多次循环判断手机状态、多次循环获取UI元素、多次循环操作UI元素、多次手机文件读写等操作，若采用常规的串行指令发送执行方式，执行过程中很大的过程会消耗在数据发送、数据接受、数据等待的过程中。通过本发明提出的调度和执行整体在安卓手机端执行的方案，对于设计过多UI循环判断复杂流程方面，执行相同的流程，效率会大的提升。

[0078] 本发明还提供了基于安卓的RPA多任务调度系统，包括；

[0079] RPA任务调度中心模块，用于对RPA流程进行管理以及创建RPA流程并执行调度策

略；

[0080] RPA任务生成模块，用于RPA任务调度中心根据调度策略生成RPA任务；

[0081] 计算拆解模块，用于在拉取到RPA任务后进行任务的依赖计算和拆解，将RPA任务拆解为微任务依赖树；

[0082] 并行执行模块，用于根据微任务依赖树对微任务进行最大限度的并行执行；

[0083] 结果上报模块，用于等待所有微任务执行完成后对微任务执行的结果进行上报；

[0084] 分析优化模块，用于对微任务执行结果进行分析和记录，完成RPA任务的执行流程优化处理。

[0085] 具体如图2所示的基于安卓的RPA多任务调度系统，在RPA任务调度中心侧，采用RPA流程管理、任务分析、数据报表、任务队列子系统：

[0086] 流程管理用于管理用户RPA流程和流程调度策略，包含了后端接口和用户操作WEB界面，根据RPA流程和调度策略生成的RPA任务位于RPA任务队列中，任务队列中的任务按照调度设备设置、调度资源设置分配给安卓RPA机器人执行。

[0087] 任务分析用于分析汇总RPA任务执行结果，微任务分析子系统用于分析安卓RPA任务中的微任务执行情况，包括微任务执行时间分析、执行结果分析，用户优化和处理任务结果。

[0088] 在RPA任务方面，本发明对RPA任务进行了重新定义，如图3所示：

[0089] 一个RPA任务包含多个微任务，每个微任务是一个最小可执行单元，如采集数据、分析数据、数据预警均可作为最小可执行的微任务。在技术实现方式上，一个微任务即一个可调用的函数，包含一个task 参数，每个微任务执行结束后会将执行结果写入到RPA任务的全局变量中，同时持久化到文件中。微任务在执行过程中抛异常即表示微任务执行出现失败，会被标记为微任务执行失败。

[0090] 每个微任务包含任务ID、任务上级依赖、微任务元信息。微任务的属性信息在技术实现上以函数注解的方式实现，如以下为一个示例微任务：

[0091] /*

[0092] @microTask

[0093] @id test

[0094] @depends foo,bar

[0095] @meta {ui:true}

[0096] */

[0097] function TestMiroTask(task){

[0098] // 微任务执行

[0099] }

[0100] 其中 @microTask 表示声明一个微任务，@id表示微任务的ID，@depends 表示微任务所依赖的上游微任务，@meta 表示微任务的元信息，比如ui表示了该任务是一个UI独占任务不能与其他UI独占任务并行执行。

[0101] 每个微任务执行的过程中都会记录开始时间、结束时间、执行是否异常等信息，在RPA任务执行结束后统一上报给RPA任务调度中心。

[0102] 在任务执行方面，本发明最大限度的保证了多任务的并行执行：

[0103] 1.手机RPA机器人从任务调度中心拉取多个可执行的RPA任务。拉取完成后，卓任务调度中心对RPA任务进行微任务分解和依赖计算。如图4所示示例，拉取的任务A和任务B，根据微任务的ID、依赖、元信息，计算出了两个任务的微任务执行DAG图，其中任务A中的微任务3和任务B中的任务4元信息均为{ui:true}，表示是一个UI独占任务。

[0104] 2.安卓RPA机器人多任务调度系统默认开启了4个并行执行的微任务执行单元，在技术上是通过Java的线程池来实现，通过步骤1完成的微任务执行DAG图，开始执行微任务。

[0105] 3.从两个DAG图中解析出可执行的微任务A-1和B-1(A-1和B-1分别表示任务A中的微任务1和任务B中的微任务1，下同)，分别记录A-1和B-1两个微任务的执行开始时间。

[0106] 4.当上一步骤中的2个微任务执行完成时(本例方便演示说明假设两个微任务同时执行完成)，记录微任务的执行结果到全局变量中，微任务执行单元开始解析出下个可执行微任务，分别为A-2、A-3、B-2、B-3，由于只有一个UI独占任务且不会和其他任务之间冲突，4个微任务执行单元开始同时执行4个微任务。

[0107] 5.当上一步骤中的4个微任务执行完成时，多任务调度系统开始A-4、B-4、B-5，直到所有任务执行结束。

[0108] 6.当一个RPA任务所有的微任务均执行完成时，即表示该RPA任务已经完成执行，统一上报至RPA任务调度中心。

[0109] 在任务分析层面，本发明也给与了极大的数据支撑，方便RPA流程的后期优化：

[0110] 无论是RPA任务还是多任务调度系统执行的微任务，均有任务执行时间、执行结果上报至RPA任务调度中心。

[0111] 在RPA流程优化分析方面，针对上个步骤给出的RPA任务、微任务的执行时间、执行结果，可以分析汇总RPA任务在微任务层面，更细粒度的执行成功率、执行耗时，特别是对于后期成功率低、执行速度慢的RPA流程分析，有非常大的数据支撑。

[0112] 综上，本发明通过对任务调度层面、任务规范和执行层面、RPA组件执行层面的多方面优化，实现了一种基于安卓的RPA多任务调度系统。

[0113] 基于本发明的技术方案，以一个用户评论舆情监控系统的安卓RPA流程为例，在具体实施和操作过程中的一个业务场景如图5所示：

[0114] 1.上传用户评论舆情监控手机RPA流程到RPA任务调度中心，按照调度策略设定为每小时执行一次用户评论舆情监控。

[0115] 2.安卓RPA机器人拉取到用户评论舆情监控RPA任务，在本地解析出用户评论舆情监控微任务执行依赖树。

[0116] 3.根据微任务执行情况，手机多任务调度系统开始执行APP1评论采集微任务，记录微任务执行时间和微任务执行结果，微任务执行结束后存储APP1的执行结果到全局变量中。由于微任务①和微任务②都具有{ui:true}的元信息，表示都为UI独占微任务，因此不可并行执行。

[0117] 4.APP1评论采集微任务执行完成后，开始同步执行APP2评论采集微任务和APP1评论分析微任务。APP1评论分析微任务会发送评论信息到NLP进行评论情绪识别，识别出差评等关注的信息。

[0118] 5.等待APP2评论采集微任务执行完成后执行APP2评论分析微任务，待APP1和APP2评论分析任务均执行完成后开始执行分析汇总微任务。分析汇总微任务对APP1和APP2采集

的评论信息进行数据汇总处理，方便下一步的数据指标分析。

[0119] 6.分析汇总微任务完成后，开始同步执行数据报表微任务和指标分析舆情告警微任务，数据报表微任务根据上一步汇总的数据结果生成评论数据报表统一上传至RPA调度中心的数据中心，指标分析舆情告警微任务用于分析评论信息是否触发舆情告警，如果触发及时发相关通知邮件至客服人员。

[0120] 7.待所有微任务执行完成，安卓RPA机器人上报RPA任务执行时间、执行结果数据。

[0121] 本发明提出了一种用于RPA连接AI算法平台的方法及系统，根据租户ID将数据存储到RPA后端数据库，算法平台通过发送请求从数据系统获取数据，经过自动化解析训练流程后建模、保存数据信息，并注册到sop，RPA获取服务信息列表后，最后解析封装成RPA组件在RPA设计器平台呈现，这时租户即可通过拖拽RPA组件的方式运用其在算法平台定制化生成的AI模型。这种快速部署自定义AI模型到自己的RPA业务当中的方式，可以降本增效，满足租户更广泛的需求。

[0122] 本发明的创新点如下：

[0123] 本发明在手机端的RPA任务多任务并行执行、提高执行效率、降低安卓端RPA部署成本等多方面进行了突破，以最高效的方式完成RPA任务的调度。

[0124] 在任务调度方面，升级了现有的需要电脑RPA机器人配合才能完成的安卓RPA流程执行，通过创新可以安卓手机RPA机器人直连RPA任务调度中心的方式，引入更少的系统执行依赖，提高了系统的稳定性。同时不需要电脑端RPA机器人的配合执行，大大减少了安卓端RPA流程的部署成本。这种直连的调度方式彻底解决了需要电脑端配合的问题。

[0125] 在执行效率方面，在执行过程中通过拆解RPA任务为多个可并行的微任务，通过多个并行的RPA微任务执行单元同时执行，提高了执行速度。同时避免了现有电脑RPA机器人发送单条串行的RPA指令，提高了RPA流程整体的执行效率。

[0126] 在任务监控优化方面，通过同时监控多个微任务的执行过程，记录微任务执行状态和执行时间，为后期RPA执行优化提供重要的数据支撑和分析依据。

[0127] 以上所述仅是对本发明的优选实施例及原理进行了详细说明，对本领域的普通技术人员而言，依据本发明提供的思想，在具体实施方式上会有改变之处，而这些改变也应视为本发明的保护范围。

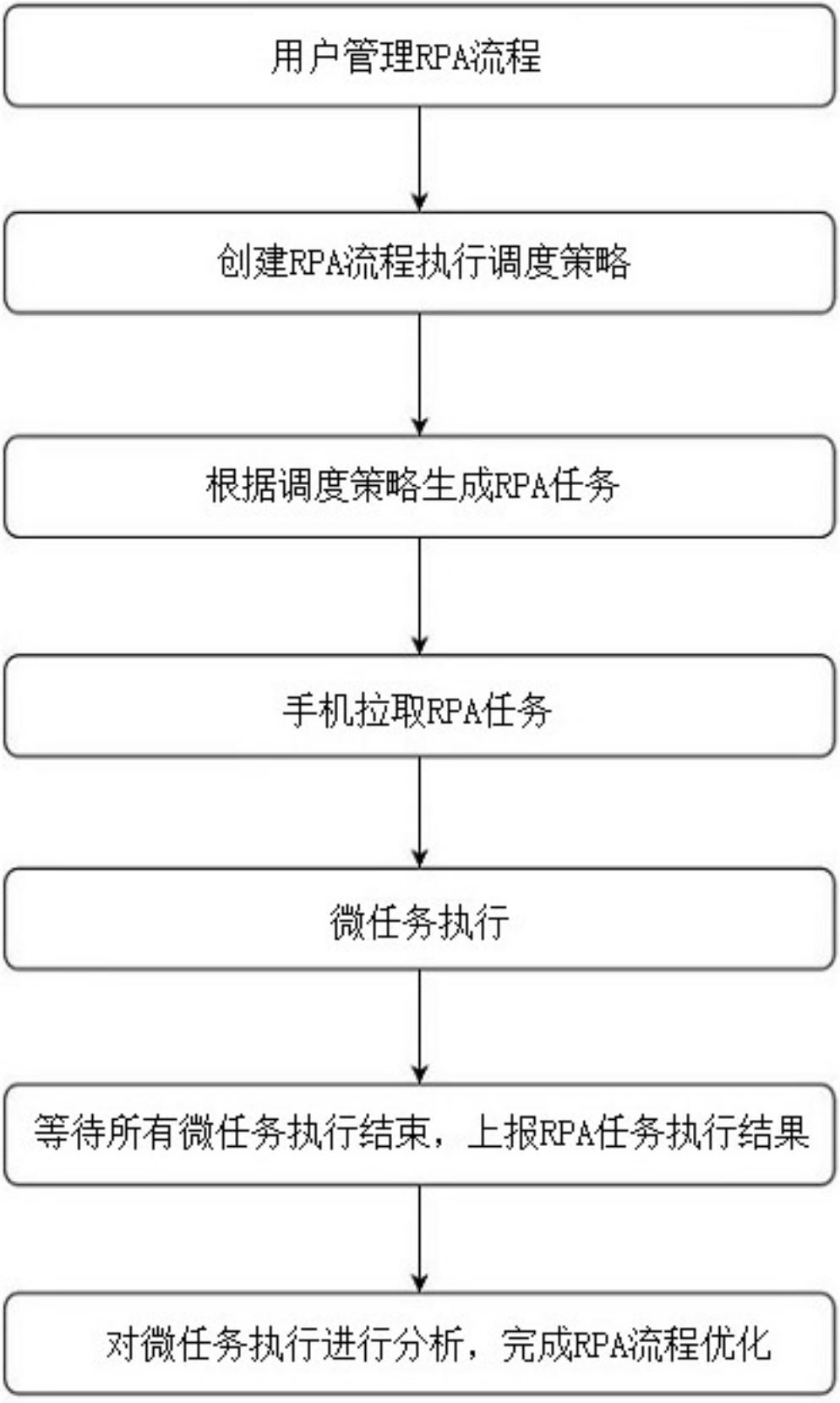

用户管理RPA流程
创建RPA流程执行调度策略
根据调度策略生成RPA任务
手机拉取RPA任务
微任务执行
等待所有微任务执行结束，上报RPA任务执行结果
对微任务执行进行分析，完成RPA流程优化

图1

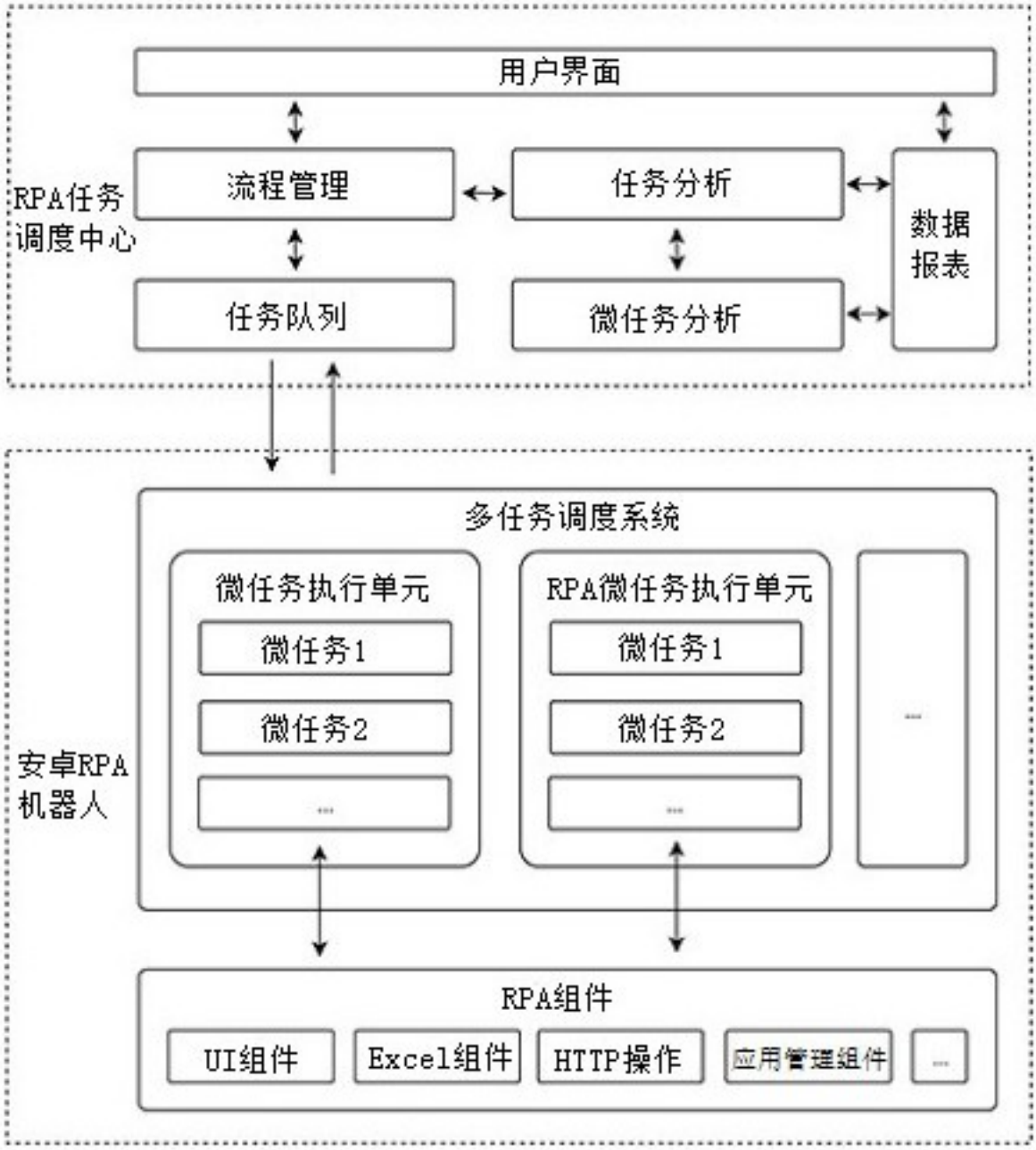

用户界面
RPA任务
调度中心
流程管理
任务分析
数据
报表
任务队列
微任务分析
多任务调度系统
微任务执行单元
微任务1
微任务2
...
RPA微任务执行单元
微任务1
微任务2
...
...
安卓RPA
机器人
RPA组件
UI组件
Excel组件
HTTP操作
应用管理组件
...

图2

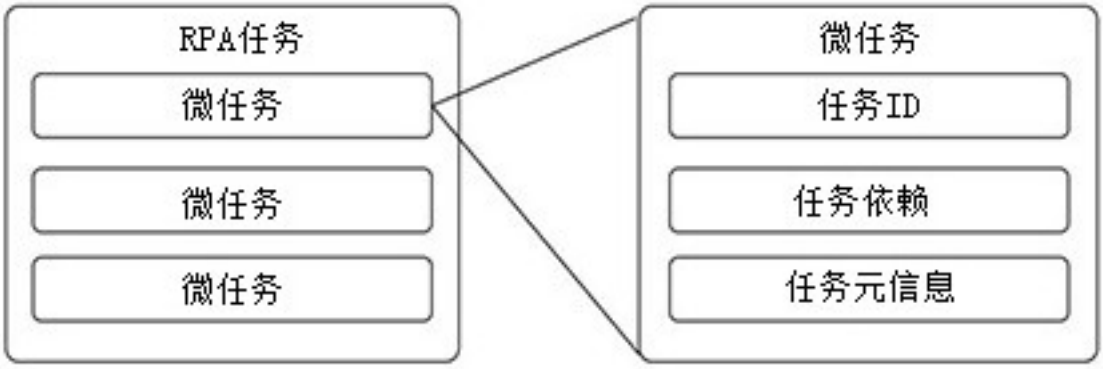


图3

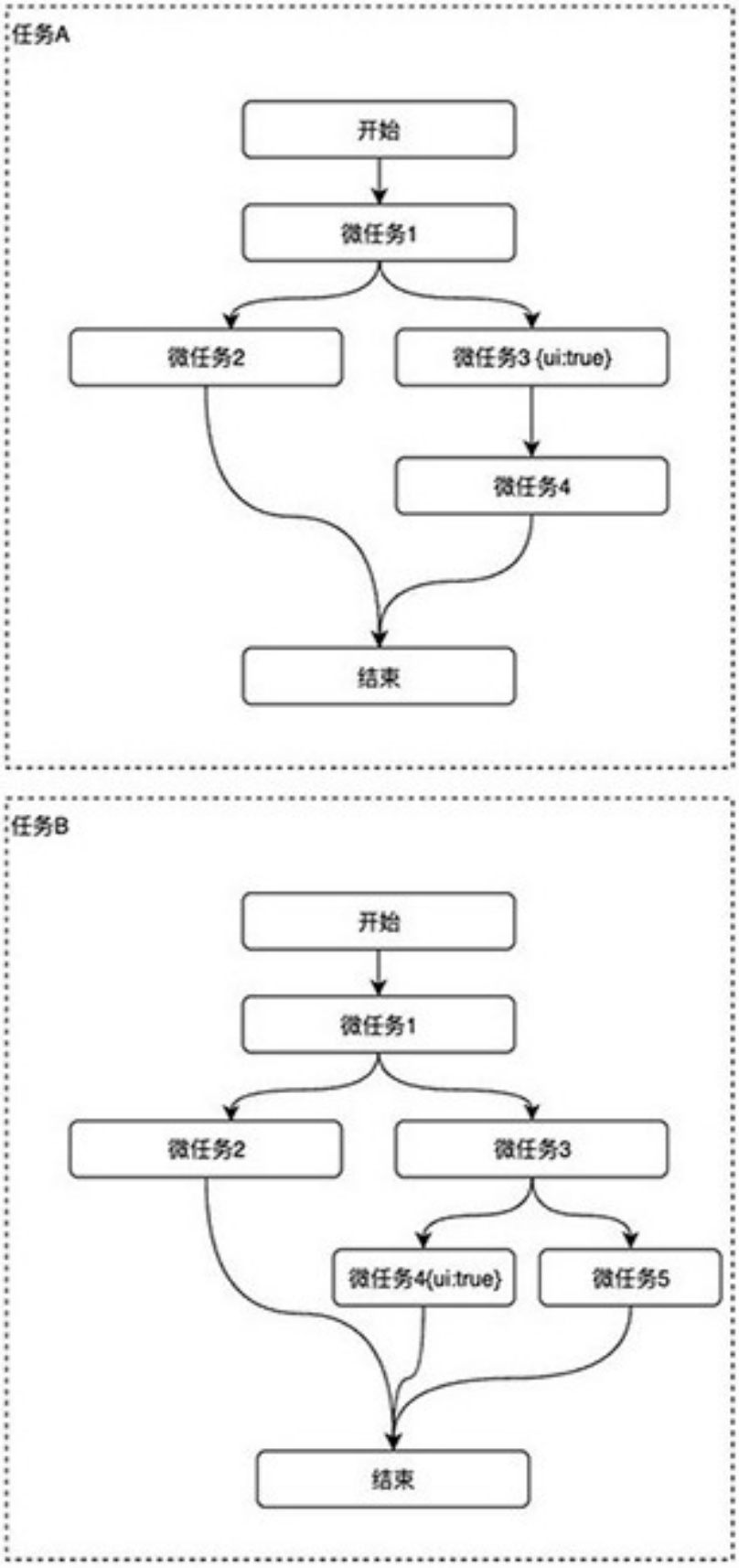

任务A
开始
微任务1
微任务2
微任务3 {ui:true}
微任务4
结束
任务B
开始
微任务1
微任务2
微任务3
微任务4{ui:true}
微任务5
结束

图4

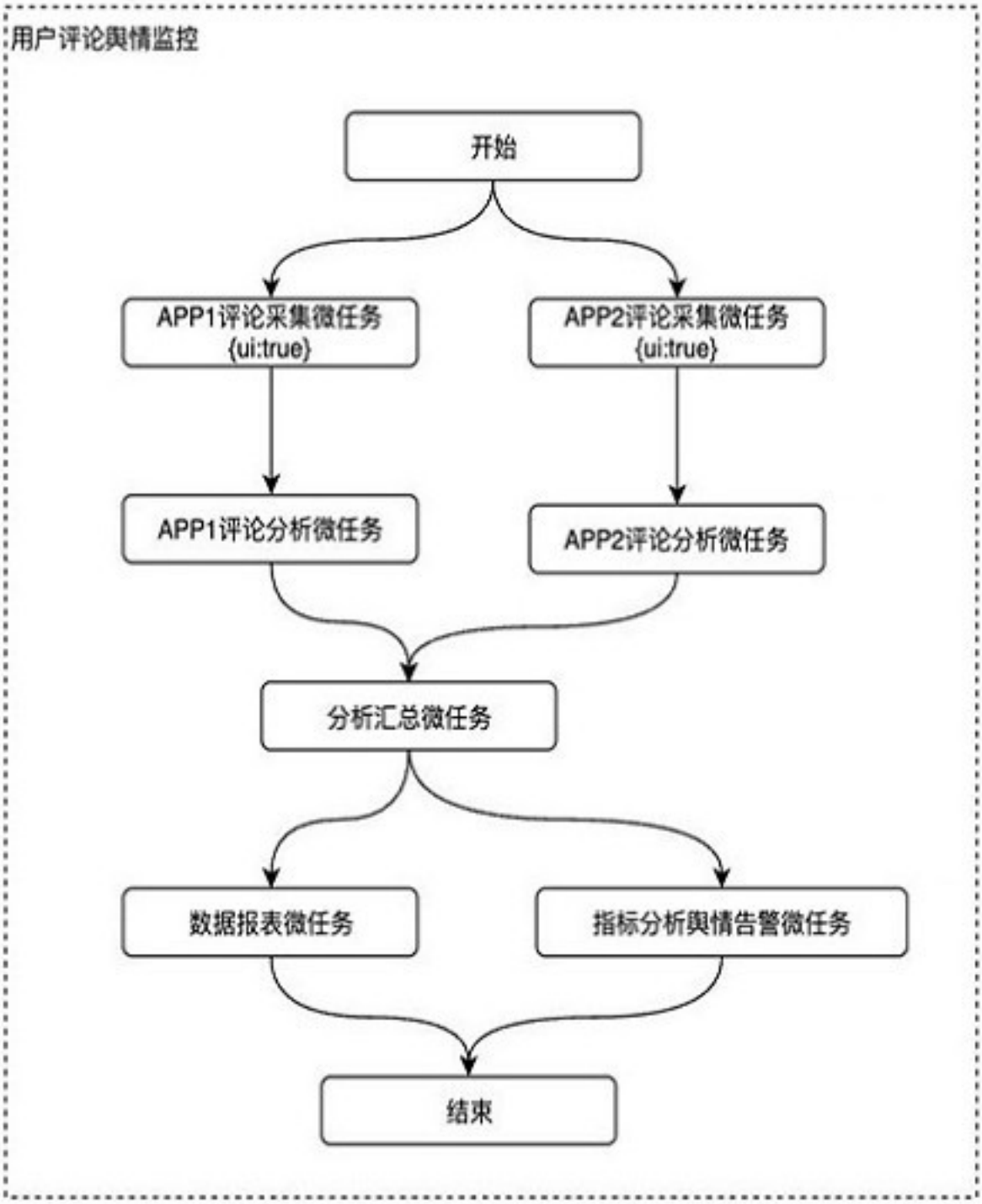

用户评论舆情监控
开始
APP1评论采集微任务
{ui:true}
APP2评论采集微任务
{ui:true}
APP1评论分析微任务
APP2评论分析微任务
分析汇总微任务
数据报表微任务
指标分析舆情告警微任务
结束

图5