



(12)发明专利申请

(10)申请公布号 CN 110708204 A

(43)申请公布日 2020.01.17

(21)申请号 201911129341.8

(22)申请日 2019.11.18

(71)申请人 上海维谛信息科技有限公司
地址 200050 上海市长宁区广顺路33号8幢
1层1593室

(72)发明人 李腾达 沈琼玉

(74)专利代理机构 北京酷爱智慧知识产权代理
有限公司 11514

代理人 刘娟

(51)Int.Cl.

H04L 12/24(2006.01)

权利要求书1页 说明书8页 附图1页

(54)发明名称

一种基于运维知识库的异常处理方法、系统、终端及介质

(57)摘要

本发明属于数据处理领域,具体涉及一种基于运维知识库的异常处理方法、系统、终端及介质,所述方法包括:包括以下步骤:监测并获取各个方面的运行数据;对运行数据进行分析,得到异常数据并生成告警信息;对告警信息进行故障收敛分类,得到分类后的故障事件;根据运维知识库对故障事件进行分析,得到故障处理方案。本发明对运行数据进行监测分析,在出现异常情况时,进行告警和故障分析,并给出故障处理方案。



1. 一种基于运维知识库的异常处理方法,其特征在于,包括以下步骤:
监测并获取各个方面的运行数据;
对运行数据进行分析,得到异常数据并生成告警信息;
对告警信息进行故障收敛分类,得到分类后的故障事件;
根据运维知识库对故障事件进行分析,得到故障处理方案。
2. 根据权利要求1所述的一种基于运维知识库的异常处理方法,其特征在于,所述运维知识库包括若干运维故障问题的故障处理方案。
3. 根据权利要求2所述的一种基于运维知识库的异常处理方法,其特征在于,所述运行数据包括但不限于日志数据、设备运行数据、交易数据和批量处理数据。
4. 根据权利要求3所述的一种基于运维知识库的异常处理方法,其特征在于,所述告警信息包括日志异常告警信息、指标异常告警信息和跑批异常告警信息。
5. 根据权利要求4所述的一种基于运维知识库的异常处理方法,其特征在于,所述对运行数据进行分析,得到异常数据并生成告警信息,具体为:
根据日志数据,解析日志模板,并提取日志关键信息,根据日志关键信息进行异常检测,得到日志异常告警信息;
根据日志数据、设备运行数据和交易数据,进行趋势预测和指标分析,得到指标异常告警信息;
根据批量处理数据,进行跑批异常检测,得到跑批异常告警信息。
6. 根据权利要求5所述的一种基于运维知识库的异常处理方法,其特征在于,所述对告警信息进行故障收敛分类,得到分类后的故障事件,具体为:
对告警信息进行去噪处理、聚合处理或预测处理,从而对告警信息进行故障分类,得到故障事件。
7. 根据权利要求6所述的一种基于运维知识库的异常处理方法,其特征在于,所述根据运维知识库对故障事件进行分析,得到故障处理方案,具体为:
将故障事件与运维知识库里的故障模板进行匹配,从而得到故障事件的类别;
根据故障事件的类别得到对应的故障处理方案。
8. 一种基于运维知识库的异常处理系统,适用于权利要求1-7任一项所述的基于运维知识库的异常处理方法,其特征在于,包括但不限于数据监测单元、分析告警单元、故障收敛单元和故障决策单元;
所述数据监测单元,用于监测并获取各个方面的运行数据;
所述分析告警单元,用于对运行数据进行分析,得到异常数据并生成告警信息;
所述故障收敛单元,用于对告警信息进行故障收敛分类,得到分类后的故障事件;
所述故障决策单元,根据运维知识库对故障事件进行分析,得到故障处理方案。
9. 一种终端,包括处理器和存储器,所述处理器和存储器相互连接,所述存储器用于存储计算机程序,所述计算机程序包括程序指令,其特征在于,所述处理器被配置用于调用所述程序指令,执行如权利要求1至7任一项所述的方法。
10. 一种计算机可读存储介质,其特征在于,所述计算机存储介质存储有计算机程序,所述计算机程序包括程序指令,所述程序指令当被处理器执行时使所述处理器执行如权利要求1至7任一项所述的方法。

一种基于运维知识库的异常处理方法、系统、终端及介质

技术领域

[0001] 本发明属于数据处理领域,具体涉及一种基于运维知识库的异常处理方法、系统、终端及介质。

背景技术

[0002] 随着企业信息化的速度越来越高,运维的压力也像山一样的成堆增长,企业的利润越来越依赖于IT系统的建设水平。因此,如何快速提高企业运维的管理也是各个企业目前最急迫的需求。现阶段的情况是大多数企业都已经完成了IT基础设施的布局,接下来需要通过提高运维水平及业务管理能力来提升整个运维的运行效率。

[0003] 现有技术中,通过各种技术工具对IT系统进行运维监控,如Tivoli、Zabbix、APM、网络抓包、应用埋点监控等会采集各种监控指标,这些指标通常被存储为时序数据(包含采集时间和指标值)。现有的运维监控通常是采集各种指标,并没有对监控的异常情况进行很好的分析处理。

发明内容

[0004] 针对现有技术中的缺陷,本发明提供了一种基于运维知识库的异常处理方法、系统、终端及介质,对运行数据进行监测分析,在出现异常情况时,进行告警和故障分析,并给出故障处理方案。

[0005] 第一方面,本发明提供了一种基于运维知识库的异常处理方法,包括以下步骤:

[0006] 监测并获取各个方面的运行数据;

[0007] 对运行数据进行分析,得到异常数据并生成告警信息;

[0008] 对告警信息进行故障收敛分类,得到分类后的故障事件;

[0009] 根据运维知识库对故障事件进行分析,得到故障处理方案。

[0010] 优选地,所述运维知识库包括若干运维故障问题的故障处理方案。

[0011] 优选地,所述运行数据包括但不限于日志数据、设备运行数据、交易数据和批量处理数据。

[0012] 优选地,所述告警信息包括日志异常告警信息、指标异常告警信息和跑批异常告警信息。

[0013] 优选地,所述对运行数据进行分析,得到异常数据并生成告警信息,具体为:

[0014] 根据日志数据,解析日志模板,并提取日志关键信息,根据日志关键信息进行异常检测,得到日志异常告警信息;

[0015] 根据日志数据、设备运行数据和交易数据,进行趋势预测和指标分析,得到指标异常告警信息;

[0016] 根据批量处理数据,进行跑批异常检测,得到跑批异常告警信息。

[0017] 优选地,所述对告警信息进行故障收敛分类,得到分类后的故障事件,具体为:

[0018] 对告警信息进行去噪处理、聚合处理或预测处理,从而对告警信息进行故障分类,

得到故障事件。

[0019] 优选地,所述根据运维知识库对故障事件进行分析,得到故障处理方案,具体为:

[0020] 将故障事件与运维知识库里的故障模板进行匹配,从而得到故障事件的类别;

[0021] 根据故障事件的类别得到对应的故障处理方案。

[0022] 第二方面,本发明提供了一种基于运维知识库的异常处理系统,适用于第一方面所述的基于运维知识库的异常处理方法,包括但不限于数据监测单元、分析告警单元、故障收敛单元和故障决策单元;

[0023] 所述数据监测单元,用于监测并获取各个方面的运行数据;

[0024] 所述分析告警单元,用于对运行数据进行分析,得到异常数据并生成告警信息;

[0025] 所述故障收敛单元,用于对告警信息进行故障收敛分类,得到分类后的故障事件;

[0026] 所述故障决策单元,根据运维知识库对故障事件进行分析,得到故障处理方案。

[0027] 第三方面,本发明提供了一种终端,包括处理器和存储器,所述处理器和存储器相互连接,所述存储器用于存储计算机程序,所述计算机程序包括程序指令,所述处理器被配置用于调用所述程序指令,执行第一方面所述的方法。

[0028] 第四方面,本发明提供了一种计算机可读存储介质,所述计算机存储介质存储有计算机程序,所述计算机程序包括程序指令,所述程序指令当被处理器执行时使所述处理器执行第一方面所述的方法。

[0029] 本发明的技术方案,对运行数据进行监测分析,在出现异常情况时,进行告警和故障分析,并给出故障处理方案。

附图说明

[0030] 为了更清楚地说明本发明具体实施方式或现有技术中的技术方案,下面将对具体实施方式或现有技术描述中所需要使用的附图作简单地介绍。在所有附图中,类似的元件或部分一般由类似的附图标记标识。附图中,各元件或部分并不一定按照实际的比例绘制。

[0031] 图1为本实施例中基于运维知识库的异常处理方法的流程图;

[0032] 图2为本实施例中基于运维知识库的异常处理系统的结构图。

具体实施方式

[0033] 下面将结合本发明实施例中的附图,对本发明实施例中的技术方案进行清楚、完整地描述,显然,所描述的实施例是本发明一部分实施例,而不是全部的实施例。基于本发明中的实施例,本领域普通技术人员在没有做出创造性劳动前提下所获得的所有其他实施例,都属于本发明保护的范围。

[0034] 应当理解,当在本说明书和所附权利要求书中使用时,术语“包括”和“包含”指示所描述特征、整体、步骤、操作、元素和/或组件的存在,但并不排除一个或多个其它特征、整体、步骤、操作、元素、组件和/或其集合的存在或添加。

[0035] 还应当理解,在此本发明说明书中所使用的术语仅仅是出于描述特定实施例的目的而并不意在限制本发明。如在本发明说明书和所附权利要求书中所使用的那样,除非上下文清楚地指明其它情况,否则单数形式的“一”、“一个”及“该”意在包括复数形式。

[0036] 还应当进一步理解,在本发明说明书和所附权利要求书中使用的术语“和/或”是

指相关联列出的项中的一个或多个的任何组合以及所有可能组合,并且包括这些组合。

[0037] 如在本说明书和所附权利要求书中所使用的那样,术语“如果”可以依据上下文被解释为“当...时”或“一旦”或“响应于确定”或“响应于检测到”。类似地,短语“如果确定”或“如果检测到[所描述条件或事件]”可以依据上下文被解释为意指“一旦确定”或“响应于确定”或“一旦检测到[所描述条件或事件]”或“响应于检测到[所描述条件或事件]”。

[0038] 具体实现中,本发明实施例中描述的终端包括但不限于诸如具有触摸敏感表面(例如,触摸屏显示器和/或触模板)的移动电话、膝上型计算机或平板计算机之类的其它便携式设备。还应当理解的是,在某些实施例中,所述设备并非便携式通信设备,而是具有触摸敏感表面(例如,触摸屏显示器和/或触模板)的台式计算机。

[0039] 实施例一:

[0040] 本实施例提供了一种基于运维知识库的异常处理方法,如图1所示,包括以下步骤:

[0041] S1,监测并获取各个方面的运行数据;

[0042] S2,对运行数据进行分析,得到异常数据并生成告警信息;

[0043] S3,对告警信息进行故障收敛分类,得到分类后的故障事件;

[0044] S4,根据运维知识库对故障事件进行分析,得到故障处理方案。

[0045] 本实施例的运行数据包括但不限于日志数据、设备运行数据、交易数据和批量处理数据。所述日志数据包括系统日志数据、应用日志数据等,所述日志,如典型日志、数据库日志、WAS(WebSphere Application Server)日志、web logic日志和交换机日志。WebSphere Application Server是一个基于Java的Web应用程序服务器,它构建在开放标准的基础之上,能部署与管理从简单的Web站点到强大的电子商务解决方案的诸多应用程序。所述设备运行数据包括CPU数据、内存数据、中间件数据、DB数据等。所述交易数据包括交易量、响应时间、成功率、响应率、交易延迟、交易类型、返回码、渠道等。所述批量处理数据包括任务开始时间、任务结束时间、交易量等。

[0046] 在采集到运行数据后,对运行数据进行分析,得到异常数据并生成告警信息,所述告警信息包括日志异常告警信息、指标异常告警信息和跑批异常告警信息。其中,进行分析告警的步骤S2具体包括S21、S22和S23这几个步骤,如下:

[0047] S21,根据日志数据,解析日志模板,并提取日志关键信息,根据日志关键信息进行异常检测,得到日志异常告警信息。

[0048] S22,根据日志数据、设备运行数据和交易数据,进行趋势预测和指标分析,得到指标异常告警信息。

[0049] S23,根据批量处理数据,进行跑批异常检测,得到跑批异常告警信息。

[0050] 本实施例中,日志的异常检测分为两个部分,第一部分是日志模式提取、第二部分是基于模式提取对日志进行异常检测。传统方法是通过正则表达式对日志进行解析,日志往往有大量的格式,解析工作非常复杂,需要花费大量时间进行调试,另外需要实时对来自多个系统的海量日志进行多个规则的解析,性能较差。本实施例中采用机器学习算法对历史日志进行离线训练,提取出模板,并且每天晚上,当日志量满足的情况下,会自动触发增量训练,自动更新日志模板。SPARK任务会调用这些模板对日志进行实时解析。通过解析出来的模板,会实时将日志解析成为结构化日志,日志异常检测则是通过变量、模板占比、突

增、分布等多种算法组合来定位日志异常。日志的异常检测分为两种：主动检测以及被动检测。主动检测可以配置多种规则，如根据匹配模板的日志条数，未匹配到模板的日志条数，模板中的某个变量的突变、分布；被动检测是KPI经过单指标实时检测，检测出问题的时候，触发对近期（本实施例中设置为故障前后2小时）日志的异常检测，具体的检测规则也包括模板数量、变量分布情况等与正常情况自动进行对比分析，判断日志是否有异常发现。具体实施时，采用的CPU (30C) +MEM (60G) 的资源可以实现对每天TB级别的日志进行实时解析以及分钟级别异常检测。当日志异常时，进行日志异常告警。

[0051] 本实施例中对日志数据进行预测分析得到日志指标KPI，对设备运行数据进行预测分析得到基础监控指标KPI，对交易数据进行预测分析得到业务指标KPI和交易指标KPI，对日志指标KPI、基础监控指标KPI、业务指标KPI和交易指标KPI进行综合分析，从而得到指标异常信息，并进行指标异常告警。

[0052] 所谓批处理是指用户将一批任务按照它们的性质分组（或分批），然后再成组（或成批）地提交给操作系统后就不再干预，由操作系统控制它们自动运行，并输出结果，从而提高对任务的处理效率。操作系统进行批处理的过程中，对其进行跑批检测，对异常情况进行匹配告警。

[0053] 本实施例中，通过异常检测得到告警信息后，对告警信息进行去噪处理、聚合处理或预测处理，从而对告警信息进行故障分类，得到故障事件。

[0054] 在传统告警中，比较容易出现告警风暴，典型的情况是当数据库服务器宕机之后，主机监控，数据库监控，网络监控会同时产生告警。传统的商业软件或者开源监控软件通常是基于规则，例如基于某个时间段告警的IP，告警的某些字段相同，从而来对告警进行聚类，存在的缺陷就是对于不同类型的告警需要另外设置，较为繁琐。在本实施例中采用的是通过算法组合来对告警进行聚合，有效的避免了告警风暴，目前生产上配置的资源为CPU (4C) +MEM (4G) 可以满足一天对10000条告警进行聚合处理。

[0055] 告警去噪主要分为两个类型：周期降噪和内容降噪。周期降噪，基于告警历史发生的模型判断周期性，具有周期性则自动降噪。

[0056] 计算告警周期性的方法分为以下三个步骤：

[0057] 1) 计算当前告警时间与历史告警时间的差分diff；

[0058] 2) 每个差分diff除以周期大小得到商和余数；

[0059] 3) 找出余数小于阈值w的商值，排序去重后计算相邻商的差值等于1的数量count，count>阈值t时表示满足周期。

[0060] 内容降噪，基于告警特征+idf+告警等级进行降噪，方法如下：

[0061] 1) 选择告警特征 $\overline{F} = \{ \text{监控指标名、机器名、模块名} \dots \}$ ；

[0062] 2) 排列组合所有告警特征，计算每种组合的idf；

[0063] 3) 计算告警信息熵 $\overline{entropy}$

[0064]
$$= \sqrt{c(1,n) * 1 * idf_1 + c(2,n) * 2 * idf_2 + \dots + c(n,n) * n * idf_n} / (2^{n-1} * n)$$

[0065] 其中，n为正整数。如果告警的信息熵 $\overline{entropy} < \text{阈值 } t$ ，则被降噪。

[0066] 本实施例中,在得到故障事件后,将故障事件与运维知识库里的故障模板进行匹配,从而得到故障事件的类别;根据故障事件的类别得到对应的故障处理方案。本实施例的运维知识库包括若干运维故障问题的故障处理方案。

[0067] 综上所述,本实施例对运行数据进行监测分析,在出现异常情况时,进行告警和故障分析,并给出故障处理方案。本实施例基于运维知识库,对系统进行异常检测和故障处理,除了可以较为精确的解决系统问题以外,通过对运维知识库的数据分析能迅速发现潜在的问题,进行趋势分析,对企业中的新需求提供策略和建议。

[0068] 实施例二:

[0069] 本实施例提供了一种基于运维知识库的异常处理系统,适用于实施例一所述的基于运维知识库的异常处理方法,如图2所示,包括但不限于数据监测单元、分析告警单元、故障收敛单元和故障决策单元;

[0070] 所述数据监测单元,用于监测并获取各个方面的运行数据;

[0071] 所述分析告警单元,用于对运行数据进行分析,得到异常数据并生成告警信息;

[0072] 所述故障收敛单元,用于对告警信息进行故障收敛分类,得到分类后的故障事件;

[0073] 所述故障决策单元,根据运维知识库对故障事件进行分析,得到故障处理方案。

[0074] 本实施例的运行数据包括但不限于日志数据、设备运行数据、交易数据和批量处理数据。所述日志数据包括系统日志数据、应用日志数据等,所述日志,如典型日志、数据库日志、WAS(WebSphere Application Server)日志、web logic日志和交换机日志。WebSphere Application Server是一个基于Java的Web应用程序服务器,它构建在开放标准的基础之上,能部署与管理从简单的Web站点到强大的电子商务解决方案的诸多应用程序。所述设备运行数据包括CPU数据、内存数据、中间件数据、DB数据等。所述交易数据包括交易量、响应时间、成功率、响应率、交易延迟、交易类型、返回码、渠道等。所述批量处理数据包括任务开始时间、任务结束时间、交易量等。

[0075] 在采集到运行数据后,对运行数据进行分析,得到异常数据并生成告警信息,所述告警信息包括日志异常告警信息、指标异常告警信息和跑批异常告警信息。其中,进行分析告警的具体方式如下:

[0076] 根据日志数据,解析日志模板,并提取日志关键信息,根据日志关键信息进行异常检测,得到日志异常告警信息。

[0077] 根据日志数据、设备运行数据和交易数据,进行趋势预测和指标分析,得到指标异常告警信息。

[0078] 根据批量处理数据,进行跑批异常检测,得到跑批异常告警信息。

[0079] 本实施例中,日志的异常检测分为两个部分,第一部分是日志模式提取、第二部分是基于模式提取对日志进行异常检测。传统方法是通过正则表达式对日志进行解析,日志往往有大量的格式,解析工作非常复杂,需要花费大量时间进行调试,另外需要实时对来自多个系统的海量日志进行多个规则的解析,性能较差。本实施例中采用机器学习算法对历史日志进行离线训练,提取出模板,并且每天晚上,当日志量满足的情况下,会自动触发增量训练,自动更新日志模板。SPARK任务会调用这些模板对日志进行实时解析。通过解析出来的模板,会实时将日志解析成为结构化日志,日志异常检测则是通过变量、模板占比、突增、分布等多种算法组合来定位日志异常。日志的异常检测分为两种:主动检测以及被动检

测。主动检测可以配置多种规则,如根据匹配模板的日志条数,未匹配到模板的日志条数,模板中的某个变量的突变、分布;被动检测是KPI经过单指标实时检测,检测出问题的时候,触发对近期(本实施例中设置为故障前后2小时)日志的异常检测,具体的检测规则也包括模板数量、变量分布情况等与正常情况自动进行对比分析,判断日志是否有异常发现。具体实施时,采用的CPU (30C) +MEM (60G) 的资源可以实现对每天TB级别的日志进行实时解析以及分钟级别异常检测。当日志异常时,进行日志异常告警。

[0080] 本实施例中对日志数据进行预测分析得到日志指标KPI,对设备运行数据进行预测分析得到基础监控指标KPI,对交易数据进行预测分析得到业务指标KPI和交易指标KPI,对日志指标KPI、基础监控指标KPI、业务指标KPI和交易指标KPI进行综合分析,从而得到指标异常信息,并进行指标异常告警。

[0081] 所谓批处理是指用户将一批任务按照它们的性质分组(或分批),然后再成组(或成批)地提交给操作系统后就不再干预,由操作系统控制它们自动运行,并输出结果,从而提高对任务的处理效率。操作系统进行批处理的过程中,对其进行跑批检测,对异常情况进行匹配告警。

[0082] 本实施例中,通过异常检测得到告警信息后,对告警信息进行去噪处理、聚合处理或预测处理,从而对告警信息进行故障分类,得到故障事件。

[0083] 在传统告警中,比较容易出现告警风暴,典型的情况是当数据库服务器宕机之后,主机监控,数据库监控,网络监控会同时产生告警。传统的商业软件或者开源监控软件通常是基于规则,例如基于某个时间段告警的IP,告警的某些字段相同,从而来对告警进行聚类,存在的缺陷就是对于不同类型的告警需要另外设置,较为繁琐。在本实施例中采用的是通过算法组合来对告警进行聚合,有效的避免了告警风暴,目前生产上配置的资源为CPU (4C) +MEM (4G) 可以满足一天对10000条告警进行聚合处理。

[0084] 告警去噪主要分为两个类型:周期降噪和内容降噪。周期降噪,基于告警历史发生的模型判断周期性,具有周期性则自动降噪。

[0085] 计算告警周期性的方法分为以下三个步骤:

[0086] 1) 计算当前告警时间与历史告警时间的差分diff;

[0087] 2) 每个差分diff除以周期大小得到商和余数;

[0088] 3) 找出余数小于阈值w的商值,排序去重后计算相邻商的差值等于1的数量count, count > 阈值t时表示满足周期。

[0089] 内容降噪,基于告警特征+idf+告警等级进行降噪,方法如下:

[0090] 1) 选择告警特征 $F = \{ \text{监控指标名、机器名、模块名} \dots \}$;

[0091] 2) 排列组合所有告警特征,计算每种组合的idf;

[0092] 3) 计算告警信息熵 $\overline{\text{entropy}}$

[0093]
$$= \overline{c(1,n) * 1 * idf_1 + c(2,n) * 2 * idf_2 + \dots + c(n,n) * n * idf_n} / (2^{n-1} * n)$$

[0094] 其中,n为正整数。如果告警的信息熵 $\overline{\text{entropy}} < \text{阈值 } t$,则被降噪。

[0095] 本实施例中,在得到故障事件后,将故障事件与运维知识库里的故障模板进行匹

配,从而得到故障事件的类别;根据故障事件的类别得到对应的故障处理方案。本实施例的运维知识库包括若干运维故障问题的故障处理方案。

[0096] 综上所述,本实施例对运行数据进行监测分析,在出现异常情况时,进行告警和故障分析,并给出故障处理方案。本实施例基于运维知识库,对系统进行异常检测和故障处理,除了可以较为精确的解决系统问题以外,通过对运维知识库的数据分析能迅速发现潜在的问题,进行趋分析,对企业中的新需求提供策略和建议。

[0097] 实施例三:

[0098] 本实施例提供了一种终端,包括处理器和存储器,所述处理器和存储器相互连接,所述存储器用于存储计算机程序,所述计算机程序包括程序指令,所述处理器被配置用于调用所述程序指令,执行实施例一所述的方法。

[0099] 应当理解,在本发明实施例中,所称处理器可以是中央处理单元(Central Processing Unit,CPU),该处理器还可以是其他通用处理器、数字信号处理器(Digital Signal Processor,DSP)、专用集成电路(Application Specific Integrated Circuit,ASIC)、现成可编程门阵列(Field-Programmable Gate Array,FPGA)或者其他可编程逻辑器件、分立门或者晶体管逻辑器件、分立硬件组件等。通用处理器可以是微处理器或者该处理器也可以是任何常规的处理器等。

[0100] 该存储器可以包括只读存储器和随机存取存储器,并向处理器提供指令和数据。存储器的一部分还可以包括非易失性随机存取存储器。例如,存储器还可以存储设备类型的信息。

[0101] 具体实现中,本实施例中所描述的终端可执行实施例一提供的方法实施例所描述的的实现方式,也在此不再赘述。

[0102] 实施例四:

[0103] 本实施例提供了一种计算机可读存储介质,所述计算机存储介质存储有计算机程序,所述计算机程序包括程序指令,所述程序指令当被处理器执行时使所述处理器执行实施例一所述的方法。

[0104] 计算机可读存储介质可以是前述实施例的终端的内部存储单元,例如终端的硬盘或内存。计算机可读存储介质也可以是终端的外部存储设备,例如终端上配备的插接式硬盘,智能存储卡(Smart Media Card,SMC),安全数字(Secure Digital,SD)卡,闪存卡(Flash Card)等。进一步地,计算机可读存储介质还可以既包括终端的内部存储单元也包括外部存储设备。计算机可读存储介质用于存储计算机程序以及终端所需的其他程序和数据。计算机可读存储介质还可以用于暂时地存储已经输出或者将要输出的数据。

[0105] 具体实现中,本实施例中所描述的终端可执行实施例一提供的方法实施例所描述的的实现方式,也在此不再赘述。

[0106] 本领域普通技术人员可以意识到,结合本文中所公开的实施例描述的各示例的单元,能够以电子硬件、计算机软件或者二者的结合来实现,为了清楚地说明硬件和软件的可互换性,在上述说明中已经按照功能一般性地描述了各示例的组成及步骤。这些功能究竟以硬件还是软件方式来执行,取决于技术方案的特定应用和设计约束条件。专业技术人员可以对每个特定的应用来使用不同方法来实现所描述的功能,但是这种实现不应认为超出本发明的范围。

[0107] 在本申请所提供的实施例中,应该理解到,所述步骤的划分,仅仅为一种逻辑功能划分,实际实现时可以有另外的划分方式,例如多个步骤可结合为一个步骤,一个步骤可拆分为多个步骤,或一些特征可以忽略等。

[0108] 最后应说明的是:以上各实施例仅用以说明本发明的技术方案,而非对其限制;尽管参照前述各实施例对本发明进行了详细的说明,本领域的普通技术人员应当理解:其依然可以对前述各实施例所记载的技术方案进行修改,或者对其中部分或者全部技术特征进行等同替换;而这些修改或者替换,并不使相应技术方案的本质脱离本发明各实施例技术方案的范围,其均应涵盖在本发明的权利要求和说明书的范围当中。

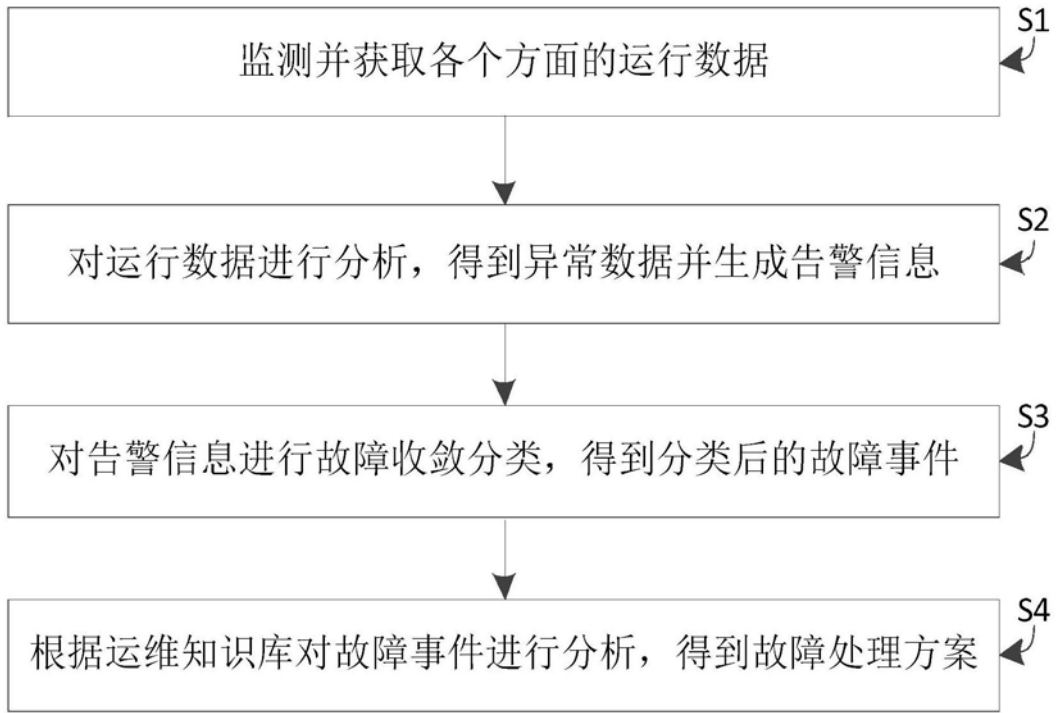


图1



图2