



(12) 发明专利

(10) 授权公告号 CN 109189810 B

(45) 授权公告日 2021.07.02

(21) 申请号 201810991706.7

G06F 16/22 (2019.01)

(22) 申请日 2018.08.28

(56) 对比文件

(65) 同一申请的已公布的文献号

CN 104123346 A, 2014.10.29

申请公布号 CN 109189810 A

CN 107145489 A, 2017.09.08

CN 105354151 A, 2016.02.24

(43) 申请公布日 2019.01.11

CN 106844703 A, 2017.06.13

(73) 专利权人 拉扎斯网络科技(上海)有限公司

CN 107273413 A, 2017.10.20

地址 200333 上海市普陀区真北路788号

US 2017351656 A1, 2017.12.07

507室

US 2007124158 A1, 2007.05.31

(72) 发明人 梁福坤 刘海宇 张超

Aviva_ye. 优化多维度查询效率. 《https://

(74) 专利代理机构 北京智信四方知识产权代理

cnblogs.com/avivaye/p/5250474.html》. 2016,

有限公司 11519

张冬冬等. 聚集索引树: 一种解决区块聚集

代理人 钟文芳 宋海龙

查询问题的方法. 《第十九届全国数据库学术会

议》. 2006,

(51) Int. Cl.

审查员 田志方

G06F 16/2453 (2019.01)

G06F 16/28 (2019.01)

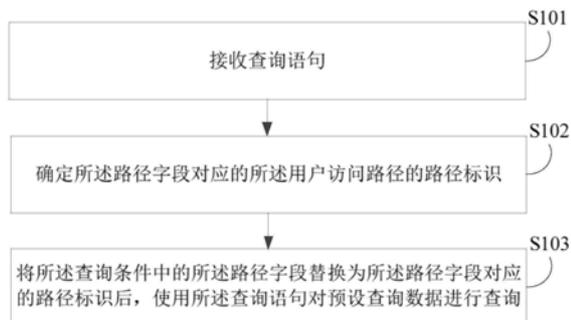
权利要求书2页 说明书13页 附图3页

(54) 发明名称

查询方法、装置、电子设备及计算机可读存储介质

(57) 摘要

本公开实施例公开了一种查询方法、装置、电子设备及计算机可读存储介质,所述方法包括:接收查询语句;其中,所述查询语句的查询条件至少包括表示用户访问路径的一个或多个路径字段;所述用户访问路径为用户在预设平台中发生预设行为时的访问路径;确定所述路径字段对应的所述用户访问路径的路径标识;将所述查询条件中的所述路径字段替换为所述路径字段对应的路径标识后,使用所述查询语句对预设查询数据进行查询;其中,所述预设查询数据至少包括以所述路径标识为查询维度的用户流量数据。本公开实施例能够直接使用路径标识对预设查询数据进行查询,节省查询时间,且无需对用户流量数据按照对象为查询维度建立查询数据,节省了存储空间。



1. 一种用户流量数据的查询方法,其特征在于,包括:

接收查询语句;其中,所述查询语句的查询条件至少包括表示用户访问路径的一个或多个路径字段;所述用户访问路径为用户从进入预设平台开始到发生预设行为时所经过的访问路径;

确定所述路径字段对应的所述用户访问路径的路径标识;

将所述查询条件中的所述路径字段替换为所述路径字段对应的路径标识后,使用所述查询语句对预设查询数据进行查询;其中,所述预设查询数据至少包括以所述路径标识为查询维度的用户流量数据;所述路径标识根据所述用户访问路径的多个路径字段进行维度聚合得到。

2. 根据权利要求1所述的查询方法,其特征在于,还包括:

获取所述预设平台中所述用户产生所述预设行为时的多个所述用户访问路径;

对所获取的每个所述用户访问路径赋予唯一的路径标识;

建立所述路径标识与所述用户访问路径之间的关联关系数据。

3. 根据权利要求2所述的查询方法,其特征在于,确定所述路径字段对应的所述用户访问路径的路径标识,包括:

根据所述路径字段从所述关联关系数据中匹配得到一个或多个所述路径标识。

4. 根据权利要求2或3所述的查询方法,其特征在于,还包括:

获取预设时间段内在所述预设平台产生的用户流量数据;

针对所述用户流量数据,以所述路径标识为查询维度生成预设查询数据。

5. 一种用户流量数据的查询装置,其特征在于,包括:

接收模块,被配置为接收查询语句;其中,所述查询语句的查询条件至少包括表示用户访问路径的一个或多个路径字段;所述用户访问路径为用户从进入预设平台开始到发生预设行为时所经过的访问路径;

确定模块,被配置为确定所述路径字段对应的所述用户访问路径的路径标识;

查询模块,被配置为将所述查询条件中的所述路径字段替换为所述路径字段对应的路径标识后,使用所述查询语句对预设查询数据进行查询;其中,所述预设查询数据至少包括以所述路径标识为查询维度的用户流量数据;所述路径标识根据所述用户访问路径的多个路径字段进行维度聚合得到。

6. 根据权利要求5所述的查询装置,其特征在于,还包括:

获取模块,被配置为获取所述预设平台中所述用户产生所述预设行为时的多个所述用户访问路径;

赋值模块,被配置为对所获取的每个所述用户访问路径赋予唯一的路径标识;

建立模块,被配置为建立所述路径标识与所述用户访问路径之间的关联关系数据。

7. 根据权利要求6所述的查询装置,其特征在于,所述确定模块包括:

匹配子模块,被配置为根据所述路径字段从所述关联关系数据中匹配得到一个或多个所述路径标识。

8. 根据权利要求6或7所述的查询装置,其特征在于,还包括:

获取子模块,被配置为获取预设时间段内在所述预设平台产生的用户流量数据;

生成子模块,被配置为针对所述用户流量数据,以所述路径标识为查询维度生成预设

查询数据。

9. 一种电子设备,其特征在于,包括存储器和处理器;其中,
所述存储器用于存储一条或多条计算机指令,其中,所述一条或多条计算机指令被所述处理器执行以实现权利要求1-4任一项所述的方法步骤。

10. 一种计算机可读存储介质,其上存储有计算机指令,其特征在于,该计算机指令被处理器执行时实现权利要求1-4任一项所述的方法步骤。

查询方法、装置、电子设备及计算机可读存储介质

技术领域

[0001] 本公开涉及计算机技术领域,具体涉及一种查询方法、装置、电子设备及计算机可读存储介质。

背景技术

[0002] 流量分析平台是通过对进入系统平台的用户流量从访问路径、终端、版本、渠道、用户等多个维度进行分析,帮助相关人员或者部门更好的了解其业务的用户流量情况,从而进一步优化业务。

[0003] 已有技术中,在每天固定时间段对前一天或者当天所产生的用户流量数据从不同维度进行数据建模,但是由于对用户流量数据进行分析所使用的维度众多,导致建模得到的数据量很大,在进行查询分析时消耗时间较长。

发明内容

[0004] 本公开实施例提供一种查询方法、装置、电子设备及计算机可读存储介质。

[0005] 第一方面,本公开实施例中提供了一种查询方法。

[0006] 具体的,所述查询方法,包括:

[0007] 接收查询语句;其中,所述查询语句的查询条件至少包括表示用户访问路径的一个或多个路径字段;所述用户访问路径为用户在预设平台中发生预设行为时的访问路径;

[0008] 确定所述路径字段对应的所述用户访问路径的路径标识;

[0009] 将所述查询条件中的所述路径字段替换为所述路径字段对应的路径标识后,使用所述查询语句对预设查询数据进行查询;其中,所述预设查询数据至少包括以所述路径标识为查询维度的用户流量数据。

[0010] 结合第一方面,本公开在第一方面的第一种实现方式中,还包括:

[0011] 获取所述预设平台中所述用户产生所述预设行为时的多个所述用户访问路径;

[0012] 对所获取的每个所述用户访问路径赋予唯一的路径标识;

[0013] 建立所述路径标识与所述用户访问路径之间的关联关系数据。

[0014] 结合第一方面的第一种实现方式,,本公开在第一方面的第二种实现方式中,确定所述路径字段对应的所述用户访问路径的路径标识,包括:

[0015] 根据所述路径字段从所述关联关系数据中匹配得到一个或多个所述路径标识。

[0016] 结合第一方面的第一种实现方式和第一方面的第二种实现方式,本公开在第一方面的第三种实现方式中,还包括:

[0017] 获取预设时间段内在所述预设平台产生的用户流量数据;

[0018] 针对所述用户流量数据,以所述路径标识为查询维度生成预设查询数据。

[0019] 第二方面,本公开实施例提供了一种查询装置,包括:

[0020] 接收模块,被配置为接收查询语句;其中,所述查询语句的查询条件至少包括表示用户访问路径的一个或多个路径字段;所述用户访问路径为用户在预设平台中发生预设行

为时的访问路径；

[0021] 确定模块,被配置为确定所述路径字段对应的所述用户访问路径的路径标识；

[0022] 查询模块,被配置为将所述查询条件中的所述路径字段替换为所述路径字段对应的路径标识后,使用所述查询语句对预设查询数据进行查询;其中,所述预设查询数据至少包括以所述路径标识为查询维度的用户流量数据。

[0023] 结合第二方面,本公开在第二方面的第一种实现方式中,还包括:

[0024] 获取模块,被配置为获取所述预设平台中所述用户产生所述预设行为时的多个所述用户访问路径;

[0025] 赋值模块,被配置为对所获取的每个所述用户访问路径赋予唯一的路径标识;

[0026] 建立模块,被配置为建立所述路径标识与所述用户访问路径之间的关联关系数据。

[0027] 结合第二方面的第一种实现方式,本公开在第二方面的第二种实现方式中所述确定模块包括:

[0028] 匹配子模块,被配置为根据所述路径字段从所述关联关系数据中匹配得到一个或多个所述路径标识。

[0029] 结合第二方面的第一种实现方式和第二方面的第二种实现方式,本公开在第二方面的第三种实现方式中,还包括:

[0030] 获取子模块,被配置为获取预设时间段内在所述预设平台产生的用户流量数据;

[0031] 生成子模块,被配置为针对所述用户流量数据,以所述路径标识为查询维度生成预设查询数据。

[0032] 所述功能可以通过硬件实现,也可以通过硬件执行相应的软件实现。所述硬件或软件包括一个或多个与上述功能相对应的模块。

[0033] 在一个可能的设计中,查询装置的结构中包括存储器和处理器,所述存储器用于存储一条或多条支持查询装置执行上述第一方面中查询方法的计算机指令,所述处理器被配置为用于执行所述存储器中存储的计算机指令。所述查询装置还可以包括通信接口,用于查询装置与其他设备或通信网络通信。

[0034] 第三方面,本公开实施例提供了一种电子设备,包括存储器和处理器;其中,所述存储器用于存储一条或多条计算机指令,其中,所述一条或多条计算机指令被所述处理器执行以实现第一方面所述的方法步骤。

[0035] 第四方面,本公开实施例提供了一种计算机可读存储介质,用于存储查询装置所用的计算机指令,其包含用于执行上述第一方面中查询方法所涉及的计算机指令。

[0036] 本公开实施例提供的技术方案可以包括以下有益效果:

[0037] 本公开实施例在对用户流量数据进行分析时,至少以用户访问路径中的一个或多个对象为查询条件进行查询,并且在接收到查询语句后,对查询语句进行拆分,首先从查询条件中获得对象,并根据对象获得该对象所对应的访问路径的路径标识,之后再根据路径标识带入查询语句后对包括用户流量数据的预设查询数据进行查询。通过这种方式,可以根据系统平台上产生的用户流量数据预先建立以用户访问路径对应的路径标识为查询维度的预设查询数据,并在进行用户流量分析时,对查询语句进行拆分,先根据查询条件中对象对应的路径标识,之后再根据路径标识查询预设查询数据,避免了一次查询时,以对象为维

度进行查询导致的反映速度慢的问题,能够节省查询时间,且无需对用户流量数据按照对象为查询维度建立查询数据,节省了存储空间。

[0038] 应当理解的是,以上的一般描述和后文的细节描述仅是示例性和解释性的,并不能限制本公开。

附图说明

[0039] 结合附图,通过以下非限制性实施方式的详细描述,本公开的其它特征、目的和优点将变得更加明显。在附图中:

[0040] 图1示出根据本公开一实施方式的查询方法的流程图;

[0041] 图2示出根据本公开另一实施方式的查询方法中建立关联关系数据部分的流程图;

[0042] 图3示出根据图1所示实施方式的步骤S102的流程图;

[0043] 图4示出根据本公开一实施方式的查询装置的结构框图;

[0044] 图5示出根据本公开另一实施方式的查询装置中建立关联关系数据部分的结构框图;

[0045] 图6示出根据图4所示实施方式的确定模块402的结构框图;

[0046] 图7示出根据本公开一实施方式的应用场景示意图;

[0047] 图8是适于用来实现根据本公开一实施方式的查询方法的电子设备的结构示意图。

具体实施方式

[0048] 下文中,将参考附图详细描述本公开的示例性实施方式,以使本领域技术人员可容易地实现它们。此外,为了清楚起见,在附图中省略了与描述示例性实施方式无关的部分。

[0049] 在本公开中,应理解,诸如“包括”或“具有”等的术语旨在指示本说明书中所公开的特征、数字、步骤、行为、部件、部分或其组合的存在,并且不欲排除一个或多个其他特征、数字、步骤、行为、部件、部分或其组合存在或被添加的可能性。

[0050] 另外还需要说明的是,在不冲突的情况下,本公开中的实施例及实施例中的特征可以相互组合。下面将参考附图并结合实施例来详细说明本公开。

[0051] 图1示出根据本公开一实施方式的查询方法的流程图。如图1所示,所述查询方法包括以下步骤S101-S103:

[0052] 在步骤S101中,接收查询语句;其中,所述查询语句的查询条件至少包括表示用户访问路径的一个或多个路径字段;所述用户访问路径为用户在预设平台中发生预设行为时的访问路径;

[0053] 在步骤S102中,确定所述路径字段对应的所述用户访问路径的路径标识;

[0054] 在步骤S103中,将所述查询条件中的所述路径字段替换为所述路径字段对应的路径标识后,使用所述查询语句对预设查询数据进行查询;其中,所述预设查询数据至少包括以所述路径标识为查询维度的用户流量数据。

[0055] 在对用户流量数据进行统计分析时,为了节省分析人员的时间及其他资源,方便

分析人员直接使用查询语句获取有用信息,通常会对系统平台上预定时间段内产生的用户流量数据预先进行建模。建模的过程通常包括:获取用户在系统平台中产生的用户流量数据,这些数据通常以日志的形式存在;在获得用户流量数据之后,根据实际情况对用户流量数据进行多维度建模,也即以多种不同的维度为主键建立查询数据;在建立了查询数据之后,就可以直接使用这些维度中的一个或多个进行查询。例如,对于用户在系统平台中产生的用户访问路径对应的用户流量数据进行建模时,可以以用户访问路径可能包括的各个页面、频道、位置、内容、按钮等维度建立多种查询数据。

[0056] 本实施例中,预设平台可以是用户能够进行访问的线上平台,该预设平台可以为用户提供信息浏览、商品展示、商品购买等等服务。用户在访问预设平台时,用户产生的用户流量数据可以以日志形式记录下来。用户访问路径可以是用户从进入预设平台开始到产生某种预设行为所走过的路径,也即用户进入预设平台到产生预设行为而历经的页面元素。预设行为是用户在系统平台产生的某种操作行为,可以根据实际情况预先设定,例如对于电商运营平台而言,在对用户成单的流量进行统计分析时,可以设定预设行为为用户的下单行为。

[0057] 本实施例种,用户访问路径由一个或多个访问对象构成,而每个访问对象可以根据预设平台的不同而不同,具体根据预设平台的实际情况而定。访问对象可以是预设平台中的页面、按钮、横幅广告、菜单等页面元素。例如,用户通过点击某外卖平台首页的餐饮进入了频道二级页面,在频道二级页面的横幅广告选择了一个商户进入这个商户的点菜页。点完菜后点击“选好了”按钮,进入订单页,并点击“去支付”按钮提交订单。那么这个用户产生预设行为即下单的用户访问路径就是:首页餐饮->频道二级页的横幅广告->点菜页选好了->订单页提交订单。

[0058] 需要说明的是,在后台实现过程中,上述用户访问路径中的访问对象利用一个或多个不同的路径字段来表示,也就是说,在数据表示层面来看,用户访问路径采用一个或多个路径字段来表示。例如,可以根据网页设计方案将预设平台分成多个层级,预设平台的主页为第一层级,而从主页进入到的下一页面为第二层级,依次类推;每一层级中的页面元素还可以进行区域、位置、内容和内容类型的划分,例如,上述例子中的用户访问路径中的访问对象可以通过下面的路径字段来表示:电商平台中的首页对应层级字段是P01,餐饮对应的区域字段为A-3,餐饮对应的内容字段为A0001;频道二级页面的横幅广告对应的层级字段是P02,区域为B-1;点菜页选好了对应层级字段是P05,位置字段为E-0-1,订单页提交订单对应层级字段是P08,位置字段为H-0-1。在查询的时候,查询条件以用户访问路径中的一个或多个字段进行查询。例如查询包括P01层级区域字段A-3的用户访问路径,则可以在查询语句的where之后限定“P01_position_group='A-3'”。

[0059] 本实施例中,用户访问路径被预先赋予了路径标识。用户从进入预设平台到产生预设行为,可能经过的路径为多种多样,在此期间,用户访问该预设平台而产生的用户流量数据都以日志的形式记录下来。对于上述每种可能的路径,预先赋予路径标识,并在获得对上述用户流量数据的查询语句后,可以针对查询条件中表示用户访问路径的一个或多个路径字段匹配得到对应的路径表示。需要说明的是,在查询语句的查询条件中,表示用户访问路径的字段不一定是构成一个完整路径的多个路径字段,而可以是用户访问路径的其中一部分路径字段。例如,对于上述用户在电商平台下单时所经过的“首页餐饮->频道二级

页的横幅广告->点菜页选好了->订单页提交订单”这一路径而言,在查询语句中,可以仅以“P01_position_group='A-3'”为查询条件进行查询。

[0060] 本实施例中,在确定了查询条件中表示用户访问路径的路径字段对应的路径标识后,可以在查询条件中将字段替换为对应的路径标识,之后再对预设查询数据进行查询。

[0061] 预设查询数据为预先经过计算的,具有查询维度的用户流量数据。用户在访问预设平台的过程中,所产生流量数据都以日志形式记录下来。而每隔预定的时间段,系统就会对这些流量数据进行预计算。例如,每天凌晨,针对上一日产生的用户流量数据,对其进行数据建模。数据建模的过程就是将用户流量数据按照预设维度进行预计算,使得预计算得到的预设查询数据以预设维度组织而成,也即以预设维度为主键。而本实施例中,预设查询数据至少包括以路径标识为查询维度的用户流量数据。当然可以理解的是,对用户流量数据的分析不限于从用户访问路径这一维度,还可以通过其他维度,例如用户所在大区、城市、商圈、终端、版本、渠道、用户数据等也进行分析。因此,实际应用中,针对用户流量数据,还会建立以其他维度为主键的其他预设查询数据。由于用户访问路径由多个路径字段构成,而以用户访问路径为维度对用户流量数据进行分析时,传统的处理方式是以用户访问路径中的每个路径字段为维度建立预设查询数据,也就是说,如果用户访问路径中最多包括N个字段,那么就用户访问路径这一分析维度而言,需要建立N个维度的预设查询数据。如果例如大区、城市、商圈、终端、版本、渠道、用户数据等其他多个维度包括M个,则在建立单维度的预设查询数据的数量是N+M个,而建立组合维度(N+M中的任意两个维度为主键)的预设查询数据时则数量为2的N+M次方个,数据膨胀率会非常大,建立起来的预设查询数据的占用空间也会巨大。考虑到用户访问路径由于由多个路径字段表示,而且按照传统方法的话需要以每个路径字段为维度建立对应的预设查询数据,数据量较大。

[0062] 因此,本公开通过将用户访问路径的多个路径字段维度进行维度聚合,形成一个路径标识,并以路径标识为维度建立预设查询数据,而不再以路径字段为维度建立预设查询数据。但是这种方式同时会带来一个弊端,就是如果查询条件中不涉及路径字段的话,查询速度会非常快;而如果存在路径字段的话,由于没有针对路径字段预先建立预设查询数据,查询起来会非常慢。因此本公开实施例,为解决这一问题,在获得查询语句后,将查询条件中表示用户访问路径的路径字段替换为对应的路径标识,并以路径标识为查询条件从以路径标识为维度的预设查询数据中进行查询,这样不但节省了以访问路径标识中每个路径字段为维度建立预设查询数据的时间资源和空间资源,并且在后续查询过程中通过将查询语句分两步走的方式(第一步确定路径标识,第二步以路径标识为查询条件进行查询),大大提高了查询速度,使得通过路径标识建立预设查询数据的方式对于查询速度的影响可以忽略不记。

[0063] 本公开上述实施例为解决查询速度慢的问题,通过将查询条件中包括路径字段的查询语句拆分成两部分来实现,下面通过示例说明:

[0064] 接收到的查询语句如下:

[0065] `SELECT count(distinct cuid) as user_num,sum(real_total_price)/1000 as real_total_price,count(0) as order_num FROM fact_flow_order_funnel_ex join dim_trace_details_ex on path_id=id WHERE index_day='20180306'and p05_position_id in('E','E','E-0','E-1','E-0','E-0-1','E-1-1','E-1-2','E-1-3')and`

p02_position_group='B-1'and me_position_id in('H','H','H-0','H-0','H-0-1')
and p01_content_id='A0001'and p01_position_group='A-3'and from_type in('na-
android','na-iphone')

[0066] 其中,查询条件where后面的p05_position_id、p02_position_id、me_position_id、p01_content_id、p01_position_group都是路径字段;

[0067] 本公开实施例通过首先下面的查询语句确定路径字段对应的路径标识:

```
[0068] select id from dim_trace_details_ex where p05_position_id in ('E','E','E-0','E-1','E-0','E-0-1','E-1-1','E-1-2','E-1-3') and p02_position_group='B-1' and me_position_id in ('H','H','H-0','H-0','H-0-1') and p01_content_id='A0001' and p01_position_group='A-3'
```

[0069] 之后,本公开实施例将所获得的路径标识替换上述路径字段,得到下面的查询语句:

```
[0070] SELECT count(distinct cuid) AS user_num,sum(real_total_price)/1000 AS real_total_price,count(0) AS order_num
FROM fact_flow_order_funnel_ex JOIN dim_trace_details_ex ON path_id=id
WHERE index_day='20180306'
AND path_id in(16965 ,13460 ,14214 ,14410 ,15027 ,15657 ,23466 ,21482 ,22183 ,22509 ,
19450 ,31455 ,31493 ,30368 ,29604 ,38874 ,38022 ,39081 ,19338 ,35923 ,
40272 ,35298 ,45985 ,46775 ,46377 ,42187 ,47784 ,40572 ,28472 ,57907 ,
56725 ,48569 ,67676 ,67665 ,65878 ,61280 ,67564 ,67842 ,69070 ,68478 ,
69410 ,70190 ,36883 ,44227 ,63077 ,66215 ,56945 ,60217 ,62075 ,74546 ,
74107 ,64913 ,65711 ,75178 ,74959 ,74669 ,72492 ,58131 ,76484 ,75989 ,
76295 ,76727 ,76829 ,77496 ,77228 ,49190 ,78050 ,80750 ,81336 ,81695 ,
81724 ,82219 ,82732 ,83591 ,83112 ,59574 ,43670 ,86599 ,86671 ,87230 ,
87828 ,88900 ,89442 ,89377 ,89241 ,89394 ,90064 ,90735 ,91740 ,92386 ,
92935 ,92819 ,93646 ,94593 ,94338 ,63336 ,94684 ,95439 ,95702 ,73068 ,
97880 ,97727 ,98414 ,99914 ,99677 ,100057 ,102123 ,102007 ,102119 ,102013 ,
104797 ,104870 ,104693 ,105203 ,105850 ,105844 ,107003 ,106925 ,106935 ,108085 ,
113385 ,113937 ,114275 ,2994 ,1581 ,2303 ,6903 ,9873 ,9534 ,7684 ,8327 ,3729 ,
114809 ,4077 ,115034 ,517 ,8164 ,118208 ,118238 ,117856 ,118291 ,4132 ,120208 ,
49313 ,56239 ,120113 ,120252 ,120949)
AND from_type IN ('na-android','na-iphone')
```

[0071] 其中,path_id为根据路径字段确定的路径标识。

[0072] 本公开实施例在对用户流量数据进行分析时,至少以用户访问路径中的一个或多个对象为查询条件进行查询,并且在接收到查询语句后,对查询语句进行拆分,首先从查询条件中获得对象,并根据对象获得该对象所对应的访问路径的路径标识,之后再路径标识带入查询语句后对包括用户流量数据的预设查询数据进行查询。通过这种方式,可以根据系统平台上产生的用户流量数据预先建立以用户访问路径对应的路径标识为查询维度的预设查询数据,并在进行用户流量分析时,对查询语句进行拆分,先根据查询条件中对象对应的路径标识,之后再根据路径标识查询预设查询数据,避免了一次查询时,以对象为维度进行查询导致的反映速度慢的问题,能够节省查询时间,且无需对用户流量数据按照对象为查询维度建立查询数据,节省了存储空间。

[0073] 在本实施例的一个可选实现方式中,如图2所示,所述方法进一步包括以下步骤S201-S203:

[0074] 在步骤S201中,获取所述预设平台中所述用户产生所述预设行为时的多个所述用户访问路径;

[0075] 在步骤S202中,对所获取的每个所述用户访问路径赋予唯一的路径标识;

[0076] 在步骤S203中,建立所述路径标识与所述用户访问路径之间的关联关系数据。

[0077] 该可选的实现方式中,在对用户流量数据进行分析之前,为了分析过程中能够快速查询到所需要的数据,可以对用户流量数据进行预计算。在用户访问预设平台过程中,用户产生的流量数据通常以日志形式记录下来,而这些日志数据如果不经预计算的话,会非常杂乱,无法进行有效的检索。而预先以一个或多个分析维度建立预设查询数据后,再对预设查询数据进行查询不但会提高查询速度,还能够提高查询有效性。本实施例主要针对用户访问路径这一分析维度而言。

[0078] 用户访问路径可以根据实际预设平台的架构进行划分。例如,由于预设平台的网页设计都是有层级关系的,因此可以人为对用户访问路径进行层级划分。例如,按照从预设系统的主页进入任一页面所要经过路径的最大层级进行划分,假如最大层级为10级,主页则可以设为P0级,而从主页点击按钮或其他页面元素进入的是下一级P1,依此类推,第10级P9所对应的页面或者其他访问对象则不包含任何其他能够通过点击或者其他操作而进入下一级的页面。如果按照已有技术对该用户访问路径所对应的用户流量数据进行数据建模,需要从P0-P9十个层级以及每个层级所包括的页面元素进行建模,假如每个层级所包括的页面元素可以从4个方面进行识别,包括页面元素的区域、位置、内容以及内容标识,那么在针对用户访问路径进行数据建模时,为了能够从路径字段进行查询,则需要建立 $4 \times 10 = 40$ 种维度的预设查询数据,也即以40个上述路径字段为主键建立40种不同的预设查询数据,这使得数据建模所花费的时间过长,并且建模得到的40种不同的预设查询数据所占用的存储空间也很大,尤其是在需要跟其他分析维度进行组合维度的数据建模时,预设查询数据以指数级膨胀。而本实施例中,可以预先对用户访问路径赋予唯一的路径标识,在数据建模时,以路径标识为维度建立预设查询数据,这样不但能够节省数据建模的时间,而且还能够节省查询数据的存储空间。

[0079] 本实施例中,获取用户在预设平台中产生预设行为时可能会经过的多个用户访问路径,所获取的多个用户访问路径可以是用户产生预设行为时所有可能的路径,也可以是经常出现的其中一部分路径,具体根据实际情况设定,在此不做限制。对于获得的每条用户访问路径赋予唯一的路径标识,并将路径标识和对应的用户访问路径关联存储,得到关联关系数据。

[0080] 在本实施例的一个可选实现方式中,所述步骤S102即确定所述路径字段对应的所述用户访问路径的路径标识的步骤,进一步包括以下步骤:

[0081] 根据所述路径字段从所述关联关系数据中匹配得到一个或多个所述路径标识。

[0082] 该可选的实现方式中,预先建立了路径标识与用户访问路径中路径字段之间的关联关系数据后,可以使用路径字段从关联关系数据中进行匹配,得到该路径字段可能对应的用户访问路径。关联关系数据可以以映射表的形式存储。一个路径字段可能对应多个路径标识,这是因为有些用户访问路径是有部分交叉重合的。

[0083] 在本实施例的一个可选实现方式中,如图3所示,所述步骤S102进一步包括以下步骤S301-S302:

[0084] 在步骤S301中,获取预设时间段内用户在所述预设平台产生的用户流量数据;

[0085] 在步骤S302中,针对所述用户流量数据,以所述路径标识为查询维度生成预设查询数据。

[0086] 在该可选的实现方式中,用户流量分析可以周期性进行,也可以有针对性的在某

个条件的触发下进行。例如,每天定时对上一日产生的用户流量数据进行分析。用户流量数据是用户在访问预设平台时产生的数据,包括用户数据、用户在预设平台上的进行的操作、所访问的页面元素等等。在用户访问预设平台的期间,用户流量数据以日志的形式记录下来。在对用户流量数据进行分析之前,首先从日志中获取用户流量数据。由于所获取的用户流量数据是一条一条的日志数据,因此可以对其进行一种或多种查询维度的预计算,也即将用户流量数据按照查询维度为主键进行组织,生成预设查询数据。该可选的实现方式中,针对用户产生预设行为时所经过的用户访问路径对应的路径标识对用户流量数据进行预计算,生成以路径标识为查询维度的预设查询数据,这样后续就可以直接使用路径标识对预设查询数据进行查询。通过这种方式,可以使得用户流量数据能够以路径标识进行检索,为后续用户流量数据的分析提供数据支持。

[0087] 下述为本公开装置实施例,可以用于执行本公开方法实施例。

[0088] 图4示出根据本公开一实施方式的查询装置的结构框图,该装置可以通过软件、硬件或者两者的结合实现成为电子设备的部分或者全部。如图4所示,所述查询装置包括接收模块401、确定模块402和查询模块403:

[0089] 接收模块401,被配置为接收查询语句;其中,所述查询语句的查询条件至少包括表示用户访问路径的一个或多个路径字段;所述用户访问路径为用户在预设平台中发生预设行为时的访问路径;

[0090] 确定模块402,被配置为确定所述路径字段对应的所述用户访问路径的路径标识;

[0091] 查询模块403,被配置为将所述查询条件中的所述路径字段替换为所述路径字段对应的路径标识后,使用所述查询语句对预设查询数据进行查询;其中,所述预设查询数据至少包括以所述路径标识为查询维度的用户流量数据。

[0092] 本实施例中,预设平台可以是用户能够进行访问的线上平台,该预设平台可以为用户提供信息浏览、商品展示、商品购买等等服务。用户在访问预设平台时,用户产生的用户流量数据可以以日志形式记录下来。用户访问路径可以是用户从进入预设平台开始到产生某种预设行为所走过的路径,也即用户进入预设平台到产生预设行为而历经的页面元素。预设行为是用户在系统平台产生的某种操作行为,可以根据实际情况预先设定,例如对于电商运营平台而言,在对用户成单的流量进行统计分析时,可以设定预设行为为用户的下单行为。

[0093] 本实施例中,用户访问路径由一个或多个访问对象构成,而每个访问对象可以根据预设平台的不同而不同,具体根据预设平台的实际情况而定。访问对象可以是预设平台中的页面、按钮、横幅广告、菜单等页面元素。例如,用户通过点击某外卖平台首页的餐饮进入了频道二级页面,在频道二级页面的横幅广告选择了一个商户进入这个商户的点菜页。点完菜后点击“选好了”按钮,进入订单页,并点击“去支付”按钮提交订单。那么这个用户产生预设行为即下单的用户访问路径就是:首页餐饮->频道二级页的横幅广告->点菜页选好了->订单页提交订单。

[0094] 需要说明的是,在后台实现过程中,上述用户访问路径中的访问对象利用一个或多个不同的路径字段来表示,也就是说,在数据表示层面来看,用户访问路径采用一个或多个路径字段来表示。例如,可以根据网页设计方案将预设平台分成多个层级,预设平台的主页为第一层级,而从主页进入到的下一页面为第二层级,依次类推;每一层级中的页面元素

还可以进行区域、位置、内容和内容类型的划分,例如,上述例子中的用户访问路径中的访问对象可以通过下面的路径字段来表示:电商平台中的首页对应层级字段是P01,餐饮对应的区域字段为A-3,餐饮对应的内容字段为A0001;频道二级页面的横幅广告对应的层级字段是P02,区域为B-1;点菜页选好了对应层级字段是P05,位置字段为E-0-1,订单页提交订单对应层级字段是P08,位置字段为H-0-1。在查询的时候,查询条件以用户访问路径中的一个或多个字段进行查询。例如查询包括P01层级区域字段A-3的用户访问路径,则可以在查询语句的where之后限定“P01_position_group='A-3'”。

[0095] 本实施例中,用户访问路径被预先赋予了路径标识。用户从进入预设平台到产生预设行为,可能经过的路径为多种多样,在此期间,用户访问该预设平台而产生的用户流量数据都以日志的形式记录下来。对于上述每种可能的路径,预先赋予路径标识,并在获得对上述用户流量数据的查询语句后,可以针对查询条件中表示用户访问路径的一个或多个路径字段匹配得到对应的路径表示。需要说明的是,在查询语句的查询条件中,表示用户访问路径的字段不一定是构成一个完整路径的多个路径字段,而可以是用户访问路径的其中一部分路径字段。例如,对于上述用户在电商平台下单时所经过的“首页餐饮->频道二级页的横幅广告->点菜页选好了->订单页提交订单”这一路径而言,在查询语句中,可以仅以“P01_position_group='A-3'”为查询条件进行查询。

[0096] 本实施例中,在确定了查询条件中表示用户访问路径的路径字段对应的路径标识后,可以在查询条件中将字段替换为对应的路径标识,之后再对预设查询数据进行查询。

[0097] 预设查询数据为预先经过计算的,具有查询维度的用户流量数据。用户在访问预设平台的过程中,所产生流量数据都以日志形式记录下来。而每隔预定的时间段,系统就会对这些流量数据进行预计算。例如,每天凌晨,针对上一日产生的用户流量数据,对其进行数据建模。数据建模的过程就是将用户流量数据按照预设维度进行预计算,使得预计算得到的预设查询数据以预设维度组织而成,也即以预设维度为主键。而本实施例中,预设查询数据至少包括以路径标识为查询维度的用户流量数据。当然可以理解的是,对用户流量数据的分析不限于从用户访问路径这一维度,还可以通过其他维度,例如用户所在大区、城市、商圈、终端、版本、渠道、用户数据等也进行分析。因此,实际应用中,针对用户流量数据,还会建立以其他维度为主键的其他预设查询数据。由于用户访问路径由多个路径字段构成,而以用户访问路径为维度对用户流量数据进行分析时,传统的处理方式是以用户访问路径中的每个路径字段为维度建立预设查询数据,也就是说,如果用户访问路径中最多包括N个字段,那么就用户访问路径这一分析维度而言,需要建立N个维度的预设查询数据。如果例如大区、城市、商圈、终端、版本、渠道、用户数据等其他多个维度包括M个,则在建立单维度的预设查询数据的数量是N+M个,而建立组合维度(N+M中的任意两个维度为主键)的预设查询数据时则数量为2的N+M次方个,数据膨胀率会非常大,建立起来的预设查询数据的占用空间也会巨大。考虑到用户访问路径由于由多个路径字段表示,而且按照传统方法的话需要以每个路径字段为维度建立对应的预设查询数据,数据量较大。

[0098] 因此,本公开通过将用户访问路径的多个路径字段维度进行维度聚合,形成一个路径标识,并以路径标识为维度建立预设查询数据,而不再以路径字段为维度建立预设查询数据。但是这种方式同时会带来一个弊端,就是如果查询条件中不涉及路径字段的话,查询速度会非常快;而如果存在路径字段的话,由于没有针对路径字段预先建立预设查询数

据,查询起来会非常慢。因此本公开实施例,为解决这一问题,在获得查询语句后,将查询条件中表示用户访问路径的路径字段替换为对应的路径标识,并以路径标识为查询条件从以路径标识为维度的预设查询数据中进行查询,这样不但节省了以访问路径标识中每个路径字段为维度建立预设查询数据的时间资源和空间资源,并且在后续查询过程中通过将查询语句分两步走的方式(第一步确定路径标识,第二步以路径标识为查询条件进行查询),大大提高了查询速度,使得通过路径标识建立预设查询数据的方式对于查询速度的影响可以忽略不记。

[0099] 本公开上述实施例为解决查询速度慢的问题,通过将查询条件中包括路径字段的查询语句拆分成两部分来实现,下面通过示例说明:

[0100] 接收到的查询语句如下:

```
[0101] SELECT count(distinct cuid) as user_num,sum(real_total_price)/1000 as
real_total_price,count(0) as order_num FROM fact_flow_order_funnel_ex join
dim_trace_details_ex on path_id=id WHERE index_day='20180306'and p05_
position_id in('E','E','E-0','E-1','E-0','E-0-1','E-1-1','E-1-2','E-1-3')and
p02_position_group='B-1'and me_position_id in('H','H','H-0','H-0','H-0-1')
and p01_content_id=A0001'and p01_position_group='A-3'and from_type in('na-
android','na-iphone')
```

[0102] 其中,查询条件where后面的p05_position_id、p02_position_id、me_position_id、p01_content_id、p01_position_group都是路径字段;

[0103] 本公开实施例通过首先下面的查询语句确定路径字段对应的路径标识:

```
[0104] select id from dim_trace_details_ex where p05_position_id in ('E','E','E-0','E-1','E-0','E-0-
1','E-1-1','E-1-2','E-1-3') and p02_position_group='B-1' and me_position_id in
('H','H','H-0','H-0','H-0-1') and p01_content_id='A0001' and p01_position_group='A-3'
```

[0105] 之后,本公开实施例将所获得的路径标识替换上述路径字段,得到下面的查询语句:

```
[0106] SELECT count(distinct cuid) AS user_num,sum(real_total_price)/1000 AS real_total_price,count(0) AS order_num
FROM fact_flow_order_funnel_ex JOIN dim_trace_details_ex ON path_id=id
WHERE index_day='20180306'
AND path_id in(16965,13460,14214,14410,15027,15657,23466,21482,22183,22509,
19450,31455,31493,30368,29604,38874,38022,39081,19338,35923,
40272,35298,45985,46775,46377,42187,47784,40572,28472,57907,
56725,48569,67676,67665,65878,61280,67564,67842,69070,68478,
69410,70190,36883,44227,63077,66215,56945,60217,62075,74546,
74107,64913,65711,75178,74959,74669,72492,58131,76484,75989,
76295,76727,76829,77496,77228,49190,78050,80750,81336,81695,
81724,82219,82732,83591,83112,59574,43670,86599,86671,87230,
87828,88900,89442,89377,89241,89394,90064,90735,91740,92386,
92935,92819,93646,94593,94338,63336,94684,95439,95702,73068,
97880,97727,98414,99914,99677,100057,102123,102007,102119,102013,
104797,104870,104693,105203,105850,105844,107003,106925,106935,108085,
113385,113937,114275,2994,1581,2303,6903,9873,9534,7684,8327,3729,
114809,4077,115034,517,8164,118208,118238,117856,118291,4132,120208,
49313,56239,120113,120252,120949)
AND from_type IN ('na-android','na-iphone')
```

[0107] 其中,path_id为根据路径字段确定的路径标识。

[0108] 本公开实施例在对用户流量数据进行分析时,至少以用户访问路径中的一个或多

个对象为查询条件进行查询,并且在接收到查询语句后,对查询语句进行拆分,首先从查询条件中获得对象,并根据对象获得该对象所对应的访问路径的路径标识,之后再根据路径标识带入查询语句后对包括用户流量数据的预设查询数据进行查询。通过这种方式,可以根据系统平台上产生的用户流量数据预先建立以用户访问路径对应的路径标识为查询维度的预设查询数据,并在进行用户流量分析时,对查询语句进行拆分,先根据查询条件中对象对应的路径标识,之后再根据路径标识查询预设查询数据,避免了一次查询时,以对象为维度进行查询导致的反映速度慢的问题,能够节省查询时间,且无需对用户流量数据按照对象为查询维度建立查询数据,节省了存储空间。

[0109] 在本实施例的一个可选实现方式中,如图5所示,所述装置进一步包括获取模块501、赋值模块502和建立模块503:

[0110] 获取模块501,被配置为获取所述预设平台中所述用户产生所述预设行为时的多个所述用户访问路径;

[0111] 赋值模块502,被配置为对所获取的每个所述用户访问路径赋予唯一的路径标识;

[0112] 建立模块503,被配置为建立所述路径标识与所述用户访问路径之间的关联关系数据。

[0113] 该可选的实现方式中,在对用户流量数据进行分析之前,为了分析过程中能够快速查询到所需要的数据,可以对用户流量数据进行预计算。在用户访问预设平台过程中,用户产生的流量数据通常以日志形式记录下来,而这些日志数据如果不经预计算的话,会非常杂乱,无法进行有效的检索。而预先以一个或多个分析维度建立预设查询数据后,再对预设查询数据进行查询不但会提高查询速度,还能够提高查询有效性。本实施例主要针对用户访问路径这一分析维度而言。

[0114] 用户访问路径可以根据实际预设平台的架构进行划分。例如,由于预设平台的网页设计都是有层级关系的,因此可以人为对用户访问路径进行层级划分。例如,按照从预设系统的主页进入任一页面所要经过路径的最大层级进行划分,假如最大层级为10级,主页则可以设为P0级,而从主页点击按钮或其他页面元素进入的是下一级P1,依此类推,第10级P9所对应的页面或者其他访问对象则不包含任何其他能够通过点击或者其他操作而进入下一级的页面。如果按照已有技术对该用户访问路径所对应的用户流量数据进行数据建模,需要从P0-P9十个层级以及每个层级所包括的页面元素进行建模,假如每个层级所包括的页面元素可以从4个方面进行识别,包括页面元素的区域、位置、内容以及内容标识,那么在针对用户访问路径进行数据建模时,为了能够从路径字段进行查询,则需要建立 $4 \times 10 = 40$ 种维度的预设查询数据,也即以40个上述路径字段为主键建立40种不同的预设查询数据,这使得数据建模所花费的时间过长,并且建模得到的40种不同的预设查询数据所占用的存储空间也很大,尤其是在需要跟其他分析维度进行组合维度的数据建模时,预设查询数据以指数级膨胀。而本实施例中,可以预先对用户访问路径赋予唯一的路径标识,在数据建模时,以路径标识为维度建立预设查询数据,这样不但能够节省数据建模的时间,而且还能够节省查询数据的存储空间。

[0115] 本实施例中,获取用户在预设平台中产生预设行为时可能会经过的多个用户访问路径,所获取的多个用户访问路径可以是用户产生预设行为时所有可能的路径,也可以是经常出现的其中一部分路径,具体根据实际情况设定,在此不做限制。对于获得的每条用户

访问路径赋予唯一的路径标识,并将路径标识和对应的用户访问路径关联存储,得到关联关系数据。

[0116] 在本实施例的一个可选实现方式中,所述确定模块402进一步包括匹配子模块:

[0117] 匹配子模块,被配置为根据所述路径字段从所述关联关系数据中匹配得到一个或多个所述路径标识。

[0118] 该可选的实现方式中,预先建立了路径标识与用户访问路径中路径字段之间的关联关系数据后,可以使用路径字段从关联关系数据中进行匹配,得到该路径字段可能对应的用户访问路径。关联关系数据可以以映射表的形式存储。一个路径字段可能对应多个路径标识,这是因为有些用户访问路径是有部分交叉重合的。

[0119] 在本实施例的一个可选实现方式中,如图6所示,所述装置进一步还包括获取子模块601和生成子模块602:

[0120] 获取子模块601,被配置为获取预设时间段内在所述预设平台产生的用户流量数据;

[0121] 生成子模块602,被配置为针对所述用户流量数据,以所述路径标识为查询维度生成预设查询数据。

[0122] 在该可选的实现方式中,用户流量分析可以周期性进行,也可以有针对性的在某个条件的触发下进行。例如,每天定时对上一日产生的用户流量数据进行分析。用户流量数据是用户在访问预设平台时产生的数据,包括用户数据、用户在预设平台上的进行的操作、所访问的页面元素等等。在用户访问预设平台的期间,用户流量数据以日志的形式记录下来。在对用户流量数据进行分析之前,首先从日志中获取用户流量数据。由于所获取的用户流量数据是一条一条的日志数据,因此可以对其进行一种或多种查询维度的预计算,也即将用户流量数据按照查询维度为主键进行组织,生成预设查询数据。该可选的实现方式中,针对用户产生预设行为时所经过的用户访问路径对应的路径标识对用户流量数据进行预计算,生成以路径标识为查询维度的预设查询数据,这样后续就可以直接使用路径标识对预设查询数据进行查询。通过这种方式,可以使得用户流量数据能够以路径标识进行检索,为后续用户流量数据的分析提供数据支持。

[0123] 图7示出了本公开实施例中提出的数据处理方法或数据处理装置的应用场景图。Apache Kylin是一个开源的分布式分析引擎,提供Hadoop/Spark之上的SQL查询接口及多维分析(OLAP)能力以支持超大规模数据。本公开实施例提出的数据处理方法可以应用在Apache Kylin处,用于多维度数据的处理,例如用户行为分析时,可以向Apache Kylin发送查询语句,Apache Kylin可以根据查询语句按照本公开实施例提出的方法对查询语句进行处理后,从hive中查询出多维度数据,能够提高查询效率。

[0124] 图8是适于用来实现根据本公开实施方式的查询方法的电子设备的结构示意图。

[0125] 如图8所示,电子设备800包括中央处理单元(CPU)801,其可以根据存储在只读存储器(ROM)802中的程序或者从存储部分808加载到随机访问存储器(RAM)803中的程序而执行上述图1所示的实施方式中的各种处理。在RAM803中,还存储有电子设备800操作所需的各种程序和数据。CPU801、ROM802以及RAM803通过总线804彼此相连。输入/输出(I/O)接口805也连接至总线804。

[0126] 以下部件连接至I/O接口805:包括键盘、鼠标等的输入部分806;包括诸如阴极射

线管 (CRT)、液晶显示器 (LCD) 等以及扬声器等的输出部分807;包括硬盘等的存储部分808;以及包括诸如LAN卡、调制解调器等的网络接口卡的通信部分809。通信部分809经由诸如因特网的网络执行通信处理。驱动器810也根据需要连接至I/O接口805。可拆卸介质811,诸如磁盘、光盘、磁光盘、半导体存储器等等,根据需要安装在驱动器810上,以便于从其上读出的计算机程序根据需要被安装入存储部分808。

[0127] 特别地,根据本公开的实施方式,上文参考图1描述的方法可以被实现为计算机软件程序。例如,本公开的实施方式包括一种计算机程序产品,其包括有形地包含在及其可读介质上的计算机程序,所述计算机程序包含用于执行图1的方法的程序代码。在这样的实施方式中,该计算机程序可以通过通信部分809从网络上被下载和安装,和/或从可拆卸介质811被安装。

[0128] 附图中的流程图和框图,图示了按照本公开各种实施方式的系统、方法和计算机程序产品的可能实现的体系架构、功能和操作。在这点上,流程图或框图中的每个方框可以代表一个模块、程序段或代码的一部分,所述模块、程序段或代码的一部分包含一个或多个用于实现规定的逻辑功能的可执行指令。也应当注意,在有些作为替换的实现中,方框中所标注的功能也可以以不同于附图中所标注的顺序发生。例如,两个接连地表示的方框实际上可以基本并行地执行,它们有时也可以按相反的顺序执行,这依所涉及的功能而定。也要注意,框图和/或流程图中的每个方框、以及框图和/或流程图中的方框的组合,可以用执行规定的功能或操作的专用的基于硬件的系统来实现,或者可以用专用硬件与计算机指令的组合来实现。

[0129] 描述于本公开实施方式中所涉及到的单元或模块可以通过软件的方式实现,也可以通过硬件的方式来实现。所描述的单元或模块也可以设置在处理器中,这些单元或模块的名称在某种情况下并不构成对该单元或模块本身的限定。

[0130] 作为另一方面,本公开还提供了一种计算机可读存储介质,该计算机可读存储介质可以是上述实施方式中所述装置中所包含的计算机可读存储介质;也可以是单独存在,未装配入设备中的计算机可读存储介质。计算机可读存储介质存储有一个或者一个以上程序,所述程序被一个或者一个以上的处理器用来执行描述于本公开的方法。

[0131] 以上描述仅为本公开的较佳实施例以及对所运用技术原理的说明。本领域技术人员应当理解,本公开中所涉及的发明范围,并不限于上述技术特征的特定组合而成的技术方案,同时也应涵盖在不脱离所述发明构思的情况下,由上述技术特征或其等同特征进行任意组合而形成的其它技术方案。例如上述特征与本公开中公开的(但不限于)具有类似功能的技术特征进行互相替换而形成的技术方案。

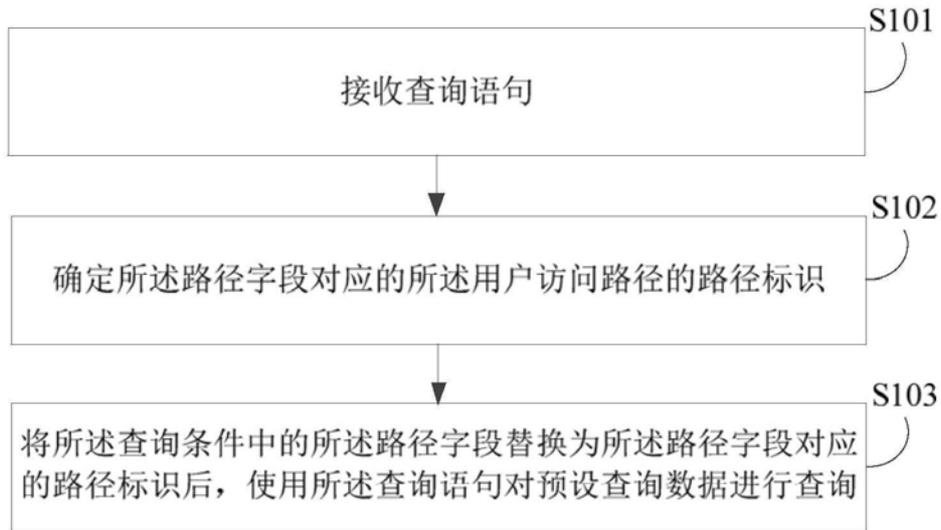


图1

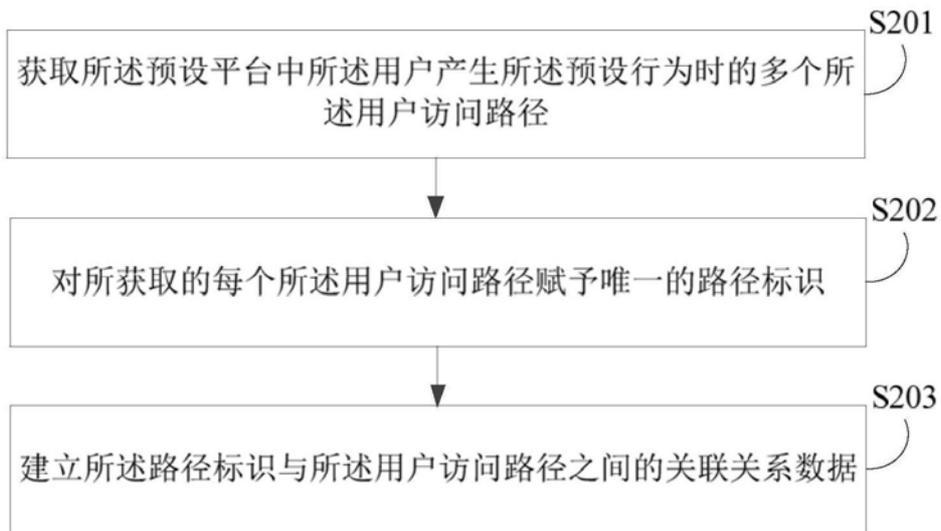


图2

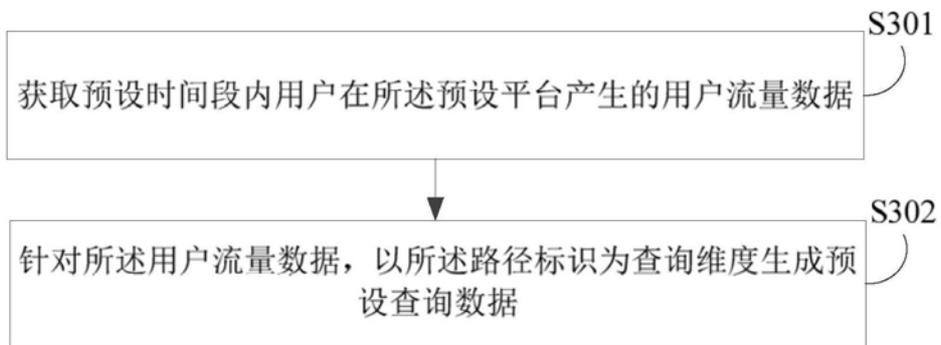


图3

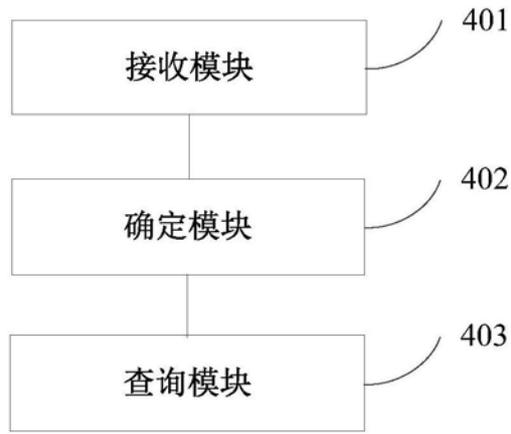


图4

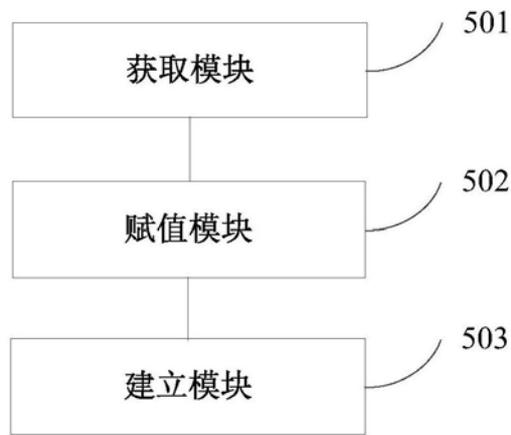


图5



图6



图7

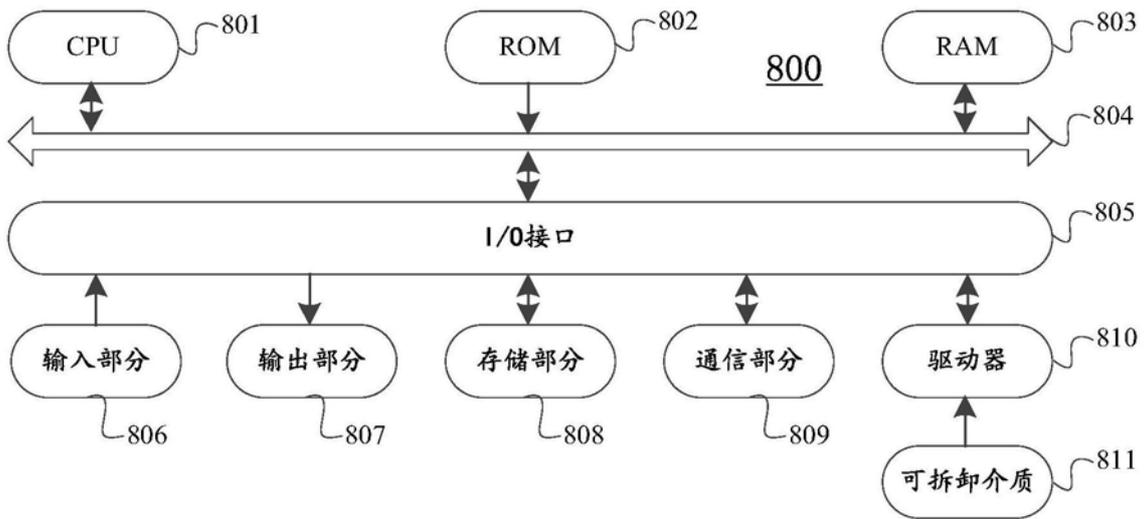


图8