



(12)发明专利申请

(10)申请公布号 CN 109635192 A  
(43)申请公布日 2019.04.16

(21)申请号 201811481741.0

(22)申请日 2018.12.05

(71)申请人 宁波深擎信息科技有限公司  
地址 315200 浙江省宁波市镇海区骆驼街  
道锦业街18号(镇海大厦)办公大楼6  
层

申请人 上海深擎信息科技有限公司

(72)发明人 柴志伟 许涵洋

(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 邹成娇

(51)Int.Cl.

G06F 16/9535(2019.01)

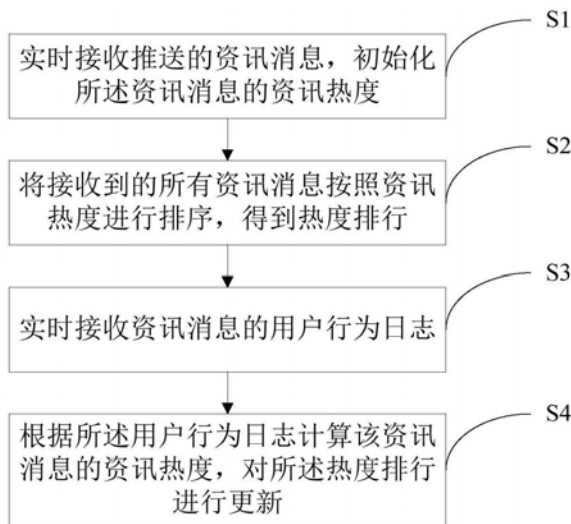
权利要求书2页 说明书6页 附图2页

(54)发明名称

面向微服务的海量资讯热度排行更新方法及平台

(57)摘要

本发明提供了面向微服务的海量资讯热度排行更新方法及平台,所述海量资讯热度排行更新方法搭建于微服务架构的分布式计算平台上,包括以下步骤:实时接收推送的资讯消息,初始化所述资讯消息的资讯热度;将接收到的所有资讯消息按照资讯热度进行排序,得到热度排行;实时接收资讯消息的用户行为日志;根据所述用户行为日志计算该资讯消息的资讯热度,对所述热度排行进行更新。该方法能够对应用服务器进行水平扩展,适用于资讯数量呈指数级增长情况下的资讯热度排行的及时更新。



1. 一种面向微服务的海量资讯热度排行更新方法,其特征在于,所述海量资讯热度排行更新方法搭建于微服务架构的分布式计算平台上,包括以下步骤:

实时接收推送的资讯消息,初始化所述资讯消息的资讯热度;

将接收到的所有资讯消息按照资讯热度进行排序,得到热度排行;

实时接收资讯消息的用户行为日志;

根据所述用户行为日志计算该资讯消息的资讯热度,对所述热度排行进行更新。

2. 根据权利要求1所述面向微服务的海量资讯热度排行更新方法,其特征在于,所述用户行为日志由利用所述分布式计算平台的窗口函数统计用户的行为数据所得。

3. 根据权利要求2所述面向微服务的海量资讯热度排行更新方法,其特征在于,所述根据所述用户行为日志计算该资讯消息的资讯热度,对所述热度排行进行更新具体包括:

针对每个资讯消息设置多个标准行为数据及其对应的权重值;

获取该资讯消息的用户行为日志中包含的行为数据,读取与所述行为数据对应的标准行为数据的权重值;

根据该资讯消息中用户行为日志的行为数据以及对应的权重值计算反馈贡献值;

利用预设的时间衰减因子对所述反馈贡献值进行计算,得到该资讯消息的资讯热度;

根据所述资讯热度对所述热度排行进行更新。

4. 根据权利要求3所述面向微服务的海量资讯热度排行更新方法,其特征在于,

所述标准行为数据包括点击次数、收藏次数和分享次数;

所述反馈贡献值的计算公式如下:

$$S(\text{Users}) = \text{Weight}(\text{click}) * \text{click} + \text{Weight}(\text{favor}) * \text{favor} + \text{Weight}(\text{share}) * \text{share};$$

其中, $S(\text{Users})$ 为反馈贡献值, $\text{click}$ 为点击次数, $\text{Weight}(\text{click})$ 为点击次数对应的权重值, $\text{favor}$ 为收藏次数, $\text{Weight}(\text{favor})$ 为收藏次数对应的权重值, $\text{share}$ 为分享次数, $\text{Weight}(\text{share})$ 为分享次数对应的权重值。

5. 根据权利要求4所述面向微服务的海量资讯热度排行更新方法,其特征在于,所述资讯热度的计算公式如下:

$$\text{Score} = (S(\text{Type}) + S(\text{Users})) / T(\text{Time});$$

其中, $\text{Score}$ 为资讯热度, $S(\text{Type})$ 为预设的资讯标准权重;

$T(\text{Time})$ 为所述时间衰减因子:

$$T(\text{Time}) = e^{(k * (T1 - T0))};$$

其中, $T0$ 为资讯消息的发布时间, $T1$ 是当前时间, $k$ 为常量参数。

6. 一种基于微服务架构的分布式计算平台,其特征在于,所述分布式计算平台执行以下步骤:

实时接收推送的资讯消息,初始化所述资讯消息的资讯热度;

将接收到的所有资讯消息按照资讯热度进行排序,得到热度排行;

实时接收资讯消息的用户行为日志;

根据所述用户行为日志计算该资讯消息的资讯热度,对所述热度排行进行更新。

7. 根据权利要求6所述基于微服务架构的分布式计算平台,其特征在于,

所述用户行为日志由利用所述分布式计算平台的窗口函数统计用户的行为数据。

8. 根据权利要求7所述基于微服务架构的分布式计算平台,其特征在于,所述根据所述

用户行为日志计算该资讯消息的资讯热度,对所述热度排行进行更新具体包括:

针对每个资讯消息设置多个标准行为数据及其对应的权重值;

获取该资讯消息的用户行为日志中包含的行为数据,读取与所述行为数据对应的标准行为数据的权重值;

根据该资讯消息中用户行为日志的行为数据以及对应的权重值计算反馈贡献值;

利用预设的时间衰减因子对所述反馈贡献值进行计算,得到该资讯消息的资讯热度;

根据所述资讯热度对所述热度排行进行更新。

9. 根据权利要求8所述基于微服务架构的分布式计算平台,其特征在于,

所述标准行为数据包括点击次数、收藏次数和分享次数;

所述反馈贡献值的计算公式如下:

$$S(\text{Users}) = \text{Weight}(\text{click}) * \text{click} + \text{Weight}(\text{favor}) * \text{favor} + \text{Weight}(\text{share}) * \text{share};$$

其中, $S(\text{Users})$ 为反馈贡献值, $\text{click}$ 为点击次数, $\text{Weight}(\text{click})$ 为点击次数对应的权重值, $\text{favor}$ 为收藏次数, $\text{Weight}(\text{favor})$ 为收藏次数对应的权重值, $\text{share}$ 为分享次数, $\text{Weight}(\text{share})$ 为分享次数对应的权重值。

10. 根据权利要求9所述基于微服务架构的分布式计算平台,其特征在于,所述资讯热度的计算公式如下:

$$\text{Score} = (S(\text{Type}) + S(\text{Users})) / T(\text{Time});$$

其中, $\text{Score}$ 为资讯热度, $S(\text{Type})$ 为预设的资讯标准权重;

$T(\text{Time})$ 为所述时间衰减因子;

$$T(\text{Time}) = e^{(k * (T1 - T0))};$$

其中, $T0$ 为资讯消息的发布时间, $T1$ 是当前时间, $k$ 为常量参数。

## 面向微服务的海量资讯热度排行更新方法及平台

### 技术领域

[0001] 本发明属于计算机技术领域,具体涉及面向微服务的海量资讯热度排行更新方法及平台。

### 背景技术

[0002] 资讯热度排行,即对已有的资讯集合,通过结合用户的兴趣度以及资讯的新颖度等计算得到的实时排行榜。资讯热度排行经常被使用在个性化推荐系统、新闻首页滚动资讯榜单等场景。

[0003] 目前已有的资讯热度排行算法主要是在普通单机上完成热度排行的计算

[0004] 但是近年来,随着互联网的爆炸性发展,全球信息量的增长速度远远快于我们处理信息的能力,资讯数量的指数级增长使得传统的计算模式在实时性上无法满足当前用户的需求。同时,资讯数量的增长也使得在单机无法在有限时间内计算得出结果。例如每隔5分钟计算得到最热门的资讯消息榜单,而传统的计算模式每一次计算全量资讯消息排行耗时超过5分钟。

[0005] 另外,由于用户数量以及资讯数量的不确定性,传统的后台系统无法有效进行水平扩展,以应对短时间内大量的用户请求。

### 发明内容

[0006] 针对现有技术中的缺陷,本发明提供一种面向微服务的海量资讯热度排行更新方法及平台,能够对应用服务器进行水平扩展,适用于资讯数量呈指数级增长情况下的资讯热度排行的及时更新。

[0007] 第一方面,一种面向微服务的海量资讯热度排行更新方法,所述海量资讯热度排行更新方法搭建于微服务架构的分布式计算平台上,包括以下步骤:

[0008] 实时接收推送的资讯消息,初始化所述资讯消息的资讯热度;

[0009] 将接收到的所有资讯消息按照资讯热度进行排序,得到热度排行;

[0010] 实时接收资讯消息的用户行为日志;

[0011] 根据所述用户行为日志计算该资讯消息的资讯热度,对所述热度排行进行更新。

[0012] 优选地,所述用户行为日志由利用所述分布式计算平台的窗口函数统计用户的行为数据所得。

[0013] 优选地,所述根据所述用户行为日志计算该资讯消息的资讯热度,对所述热度排行进行更新具体包括:

[0014] 针对每个资讯消息设置多个标准行为数据及其对应的权重值;

[0015] 获取该资讯消息的用户行为日志中包含的行为数据,读取与所述行为数据对应的标准行为数据的权重值;

[0016] 根据该资讯消息中用户行为日志的行为数据以及对应的权重值计算反馈贡献值;

[0017] 利用预设的时间衰减因子对所述反馈贡献值进行计算,得到该资讯消息的资讯热

度;

[0018] 根据所述资讯热度对所述热度排行进行更新。

[0019] 优选地,所述标准行为数据包括点击次数、收藏次数和分享次数。

[0020] 所述反馈贡献值的计算公式如下:

[0021]  $S(\text{Users}) = \text{Weight}(\text{click}) * \text{click} + \text{Weight}(\text{favor}) * \text{favor} + \text{Weight}(\text{share}) * \text{share};$

[0022] 其中, $S(\text{Users})$ 为反馈贡献值, $\text{click}$ 为点击次数, $\text{Weight}(\text{click})$ 为点击次数对应的权重值, $\text{favor}$ 为收藏次数, $\text{Weight}(\text{favor})$ 为收藏次数对应的权重值, $\text{share}$ 为分享次数, $\text{Weight}(\text{share})$ 为分享次数对应的权重值。

[0023] 优选地,所述资讯热度的计算公式如下:

[0024]  $\text{Score} = (S(\text{Type}) + S(\text{Users})) / T(\text{Time});$

[0025] 其中, $\text{Score}$ 为资讯热度, $S(\text{Type})$ 为预设的资讯标准权重;

[0026]  $T(\text{Time})$ 为所述时间衰减因子:

[0027]  $T(\text{Time}) = e^{(k * (T1 - T0))};$

[0028] 其中, $T0$ 为资讯消息的发布时间, $T1$ 是当前时间, $k$ 为常量参数。

[0029] 第二方面,一种基于微服务架构的分布式计算平台,所述分布式计算平台执行以下步骤:

[0030] 实时接收推送的资讯消息,初始化所述资讯消息的资讯热度;

[0031] 将接收到的所有资讯消息按照资讯热度进行排序,得到热度排行;

[0032] 实时接收资讯消息的用户行为日志;

[0033] 根据所述用户行为日志计算该资讯消息的资讯热度,对所述热度排行进行更新。

[0034] 优选地,所述用户行为日志由利用所述分布式计算平台的窗口函数统计用户的行为数据。

[0035] 优选地,所述根据所述用户行为日志计算该资讯消息的资讯热度,对所述热度排行进行更新具体包括:

[0036] 针对每个资讯消息设置多个标准行为数据及其对应的权重值;

[0037] 获取该资讯消息的用户行为日志中包含的行为数据,读取与所述行为数据对应的标准行为数据的权重值;

[0038] 根据该资讯消息中用户行为日志的行为数据以及对应的权重值计算反馈贡献值;

[0039] 利用预设的时间衰减因子对所述反馈贡献值进行计算,得到该资讯消息的资讯热度;

[0040] 根据所述资讯热度对所述热度排行进行更新。

[0041] 优选地,所述标准行为数据包括点击次数、收藏次数和分享次数。

[0042] 所述反馈贡献值的计算公式如下:

[0043]  $S(\text{Users}) = \text{Weight}(\text{click}) * \text{click} + \text{Weight}(\text{favor}) * \text{favor} + \text{Weight}(\text{share}) * \text{share};$

[0044] 其中, $S(\text{Users})$ 为反馈贡献值, $\text{click}$ 为点击次数, $\text{Weight}(\text{click})$ 为点击次数对应的权重值, $\text{favor}$ 为收藏次数, $\text{Weight}(\text{favor})$ 为收藏次数对应的权重值, $\text{share}$ 为分享次数, $\text{Weight}(\text{share})$ 为分享次数对应的权重值。

[0045] 优选地,所述资讯热度的计算公式如下:

[0046]  $Score = (S(Type) + S(Users)) / T(Time)$ ;

[0047] 其中,Score为资讯热度,S(Type)为预设的资讯标准权重;

[0048] T(Time)为所述时间衰减因子;

[0049]  $T(Time) = e^{(k * (T1 - T0))}$ ;

[0050] 其中,T0为资讯消息的发布时间,T1是当前时间,k为常量参数。

[0051] 由上述技术方案可知,本发明提供的面向微服务的海量资讯热度排行更新方法及平台,能够对应用服务器进行水平扩展,适用于资讯数量呈指数级增长情况下的资讯热度排行的及时更新。

## 附图说明

[0052] 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。

[0053] 图1为本申请实施例一提供的海量资讯热度排行更新方法的流程图。

[0054] 图2为本申请实施例二提供的热度排行更新方法的流程图。

## 具体实施方式

[0055] 下面将结合附图对本发明技术方案的实施例进行详细的描述。以下实施例仅用于更加清楚地说明本发明的技术方案,因此只作为示例,而不能以此来限制本发明的保护范围。需要注意的是,除非另有说明,本申请使用的技术术语或者科学术语应当为本发明所属领域技术人员所理解的通常意义。

[0056] 实施例一:

[0057] 一种面向微服务的海量资讯热度排行更新方法,参见图1,所述海量资讯热度排行更新方法搭建于微服务架构的分布式计算平台上,参见图1,包括以下步骤:

[0058] S1:实时接收推送的资讯消息,初始化所述资讯消息的资讯热度;

[0059] 具体地,所述微服务架构的分布式计算平台为大数据实时计算平台,例如Apache Flink,Apache Blink,这些分布式计算平台数据流转的实时性可以达到毫秒级,能够满足资讯数量呈指数级增长情况下的资讯消息的热度排行计算。步骤S1在接收到新的资讯消息时,对资讯消息的资讯热度进行初始化,给资讯消息的资讯热度定义一个初始值。

[0060] S2:将接收到的所有资讯消息按照资讯热度进行排序,得到热度排行;

[0061] 具体地,热度排行中资讯消息的资讯热度从前到后依次降低。

[0062] S3:实时接收资讯消息的用户行为日志;

[0063] 具体地,用户行为日志是指用户每次访问网络时所有的行为数据,包括访问、浏览、搜索、点击、收藏、分享等多种行为数据。

[0064] S4:根据所述用户行为日志计算该资讯消息的资讯热度,对所述热度排行进行更新。

[0065] 具体地,该方法根据用户行为日志中的行为数据进行实时(或定时,如每隔5分钟)计算每一个资讯消息的资讯热度,并进行热度排行更新。所述用户行为日志由日志采集模

块提供,常见的日志采集模块可以由Kafka/MQ等技术实现,只需要提供相应的接口可以获取用户的行为数据。

[0066] 该方法利用分布式计算平台进行资讯消息的热度排行更新,利用集群完成现有技术中普通单机无法完成的计算量。另外,微服务架构可以对不同的业务逻辑进行扩展,根据业务量的增长进行节点扩展,而无需让UI节点也跟着扩展,这样当用户量或者资讯消息数量增长时,可以对应用服务器进行水平扩展,可以有效利用微服务架构进行性能提升,保证系统的健壮性。克服现有技术中,单机应用无法根据应用各个模块的流量和压力进行拆分,热度排序算法受限于算法原理,无法在多台机器上分布式进行计算。

[0067] 实施例二:

[0068] 实施例二在实施例一的基础上,增加了以下内容:

[0069] 所述用户行为日志由利用所述分布式计算平台的窗口函数统计用户的行为数据所得。

[0070] 具体地,本实施例的分布式计算平台选用Spark实时分布式计算平台,这样就可以利用Spark的窗口函数,对固定时间间隔的用户行为数据进行统计,获得用户行为日志。

[0071] 参见图2,优选地,所述根据所述用户行为日志计算该资讯消息的资讯热度,对所述热度排行进行更新具体包括:

[0072] S11:针对每个资讯消息设置多个标准行为数据及其对应的权重值;

[0073] 具体地,所述标准行为数据包括点击次数、收藏次数和分享次数。用户点击、收藏、分享或者阅读某资讯消息时,将用户的行为反馈到后台系统。由于用户不同的行为对资讯消息的流行度的意义不同,所以不同的行为贡献的权重不同。比如,分享该资讯消息,可能使得该资讯消息更容易被其他用户看到,所以权重比仅仅用户自己收藏要高。比如点击的权重是1,收藏的权重是5,分享的权重是10。但不是全部的行为都是正反馈。例如用户在点击一篇内容较长的资讯消息后,阅读时长仅仅只有5秒钟,这样就可以假设为用户对该资讯消息的标题感兴趣,而对资讯消息内容不感兴趣,该行为是负反馈。

[0074] S12:获取该资讯消息的用户行为日志中包含的行为数据,读取与所述行为数据对应的标准行为数据的权重值;

[0075] S13:根据该资讯消息中用户行为日志的行为数据以及对应的权重值计算反馈贡献值;

[0076] 所述反馈贡献值的计算公式如下:

[0077]  $S(\text{Users}) = \text{Weight}(\text{click}) * \text{click} + \text{Weight}(\text{favor}) * \text{favor} + \text{Weight}(\text{share}) * \text{share};$

[0078] 其中,S(Users)为反馈贡献值,click为点击次数,Weight(click)为点击次数对应的权重值,favor为收藏次数,Weight(favor)为收藏次数对应的权重值,share为分享次数,Weight(share)为分享次数对应的权重值。

[0079] 具体地,权重值可以由管理人员根据经验进行设定,也可以通过其他方法对数据进行分析后根据不同行为的数量分布进行设定。

[0080] S14:利用预设的时间衰减因子对所述反馈贡献值进行计算,得到该资讯消息的资讯热度;

[0081] 优选地,所述资讯热度的计算公式如下:

[0082]  $Score = (S(\text{Type}) + S(\text{Users})) / T(\text{Time})$ ;

[0083] 其中,Score为资讯热度,S(Type)为预设的资讯标准权重;

[0084] T(Time)为所述时间衰减因子;

[0085]  $T(\text{Time}) = e^{(k * (T1 - T0))}$ ;

[0086] 其中,T0为资讯消息的发布时间,T1是当前时间,k为常量参数。

[0087] 具体地,S(Type)是资讯消息本身的权重,与用户行为无关。S(Type)反映了资讯的重要程度,可由资讯的发布者在发布者指定。比如重磅消息的S(Type)值要比娱乐资讯的S(Type)值大。

[0088] 由于资讯热度随时间的衰减不是线性的,已经发布的资讯消息的资讯热度必须随着时间流逝而衰减,并且趋势应该是衰减越来越快,直至趋近于零热度,所以采用上述公式计算随着时间推移的资讯消息的资讯热度,使得得到的资讯热度充分考虑了时间因素。

[0089] S15:根据所述资讯热度对所述热度排行进行更新。

[0090] 该方法得到的热度排行充分考虑了时间因素和用户的行为反馈。

[0091] 本发明实施例所提供的方法,为简要描述,实施例部分未提及之处,可参考前述方法实施例中相应内容。

[0092] 实施例三:

[0093] 一种基于微服务架构的分布式计算平台,所述分布式计算平台执行以下步骤:

[0094] 实时接收推送的资讯消息,初始化所述资讯消息的资讯热度;

[0095] 将接收到的所有资讯消息按照资讯热度进行排序,得到热度排行;

[0096] 实时接收资讯消息的用户行为日志;

[0097] 根据所述用户行为日志计算该资讯消息的资讯热度,对所述热度排行进行更新。

[0098] 优选地,所述用户行为日志由利用所述分布式计算平台的窗口函数统计用户的行为数据。

[0099] 优选地,所述根据所述用户行为日志计算该资讯消息的资讯热度,对所述热度排行进行更新具体包括:

[0100] 针对每个资讯消息设置多个标准行为数据及其对应的权重值;

[0101] 获取该资讯消息的用户行为日志中包含的行为数据,读取与所述行为数据对应的标准行为数据的权重值;

[0102] 根据该资讯消息中用户行为日志的行为数据以及对应的权重值计算反馈贡献值;

[0103] 利用预设的时间衰减因子对所述反馈贡献值进行计算,得到该资讯消息的资讯热度;

[0104] 根据所述资讯热度对所述热度排行进行更新。

[0105] 优选地,所述标准行为数据包括点击次数、收藏次数和分享次数。

[0106] 所述反馈贡献值的计算公式如下:

[0107]  $S(\text{Users}) = \text{Weight}(\text{click}) * \text{click} + \text{Weight}(\text{favor}) * \text{favor} + \text{Weight}(\text{share}) * \text{share}$ ;

[0108] 其中,S(Users)为反馈贡献值,click为点击次数,Weight(click)为点击次数对应的权重值,favor为收藏次数,Weight(favor)为收藏次数对应的权重值,share为分享次数,Weight(share)为分享次数对应的权重值。



[0109] 优选地,所述资讯热度的计算公式如下:

[0110]  $Score = (S(Type) + S(Users)) / T(Time)$ ;

[0111] 其中,Score为资讯热度,S(Type)为预设的资讯标准权重;

[0112] T(Time)为所述时间衰减因子:

[0113]  $T(Time) = e^{(k * (T1 - T0))}$ ;

[0114] 其中,T0为资讯消息的发布时间,T1是当前时间,k为常量参数。

[0115] 该分布式计算平台能够对应用服务器进行水平扩展,适用于资讯数量呈指数级增长情况下的资讯热度排行的及时更新。

[0116] 本发明实施例所提供的分布式计算平台,为简要描述,实施例部分未提及之处,可参考前述方法实施例中相应内容。

[0117] 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。

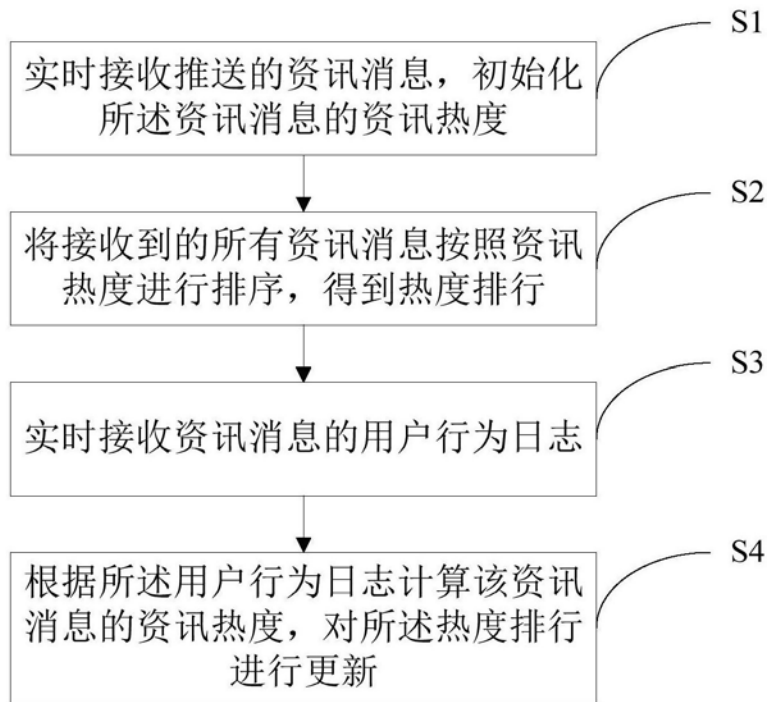


图1

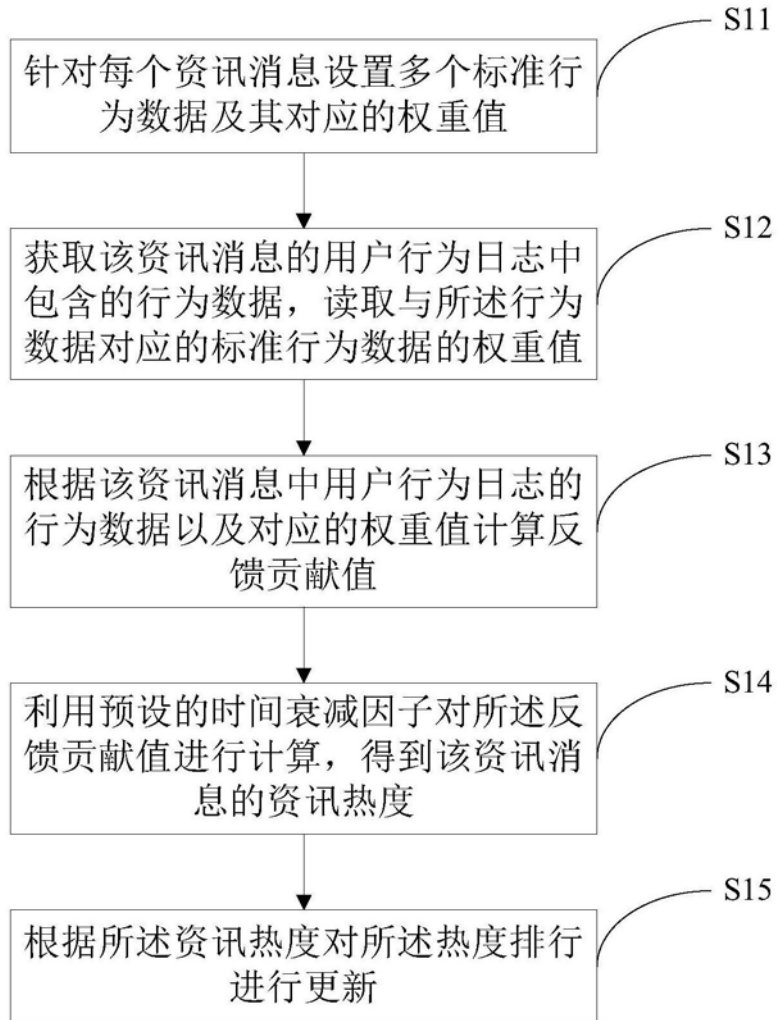


图2