



## (12) 发明专利申请

(10) 申请公布号 CN 103530304 A

(43) 申请公布日 2014. 01. 22

(21) 申请号 201310171026. 8

(22) 申请日 2013. 05. 10

(71) 申请人 TCL 集团股份有限公司

地址 516001 广东省惠州市鹅岭南路 6 号

TCL 工业大厦 8 楼技术中心

(72) 发明人 李朝 汪灏泓

(74) 专利代理机构 深圳市君胜知识产权代理事

务所 44268

代理人 王永文 杨宏

(51) Int. Cl.

G06F 17/30 (2006. 01)

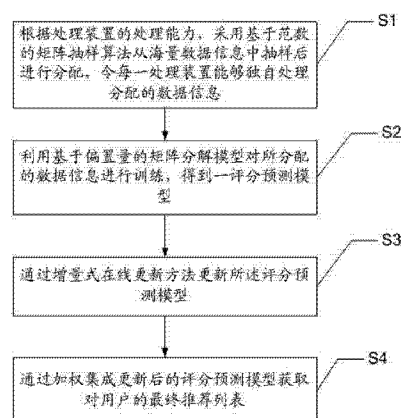
权利要求书2页 说明书9页 附图2页

### (54) 发明名称

基于自适应分布式计算的在线推荐方法、系统和移动终端

### (57) 摘要

本发明公开了一种基于自适应分布式计算的在线推荐方法、系统和移动终端,其中,自适应性分布式计算主要是通过每一处理装置自适应地将大数据抽样后训练矩阵分解模型来实现的,然后自发地进行增量式的模型在线更新,最后对各个模型进行加权集成推荐。这种推荐系统不需要通过任何分布式计算模型去建立集群。而且能够在单台或者多台普通计算机上有效处理的处理海量数据,不仅稳定性好,可扩展性高,而且还可以大大节约成本和开发效率。



1. 一种基于自适应分布式计算的在线推荐方法,用于通过处理装置对海量数据信息处理后向用户进行推荐,其特征在于,所述在线推荐方法包括以下步骤:

A、根据处理装置的处理能力,采用基于范数的矩阵抽样算法从海量数据信息中抽样后进行分配,令每一处理装置能够独自处理分配的数据信息;

B、利用基于偏置量的矩阵分解模型对所分配的数据信息进行训练,得到一评分预测模型;

C、通过增量式在线更新方法更新所述评分预测模型;

D、通过加权集成更新后的评分预测模型获取对用户的最终推荐列表。

2. 根据权利要求1所述的基于自适应分布式计算的在线推荐方法,其特征在于,所述步骤A中,采用基于范数的矩阵抽样算法具体包括以下步骤:

A1、获取海量数据信息对应的数据矩阵;

A2、对所述数据矩阵的行和列同时进行采样,得到一子矩阵;并根据向量的第一或第二范数来确保采样后的子矩阵包含的数据和海量数据之间的近似度小于预定的误差阈值。

3. 根据权利要求2所述的基于自适应分布式计算的在线推荐方法,其特征在于,所述步骤A2中具体包括以下步骤:

A21、对数据矩阵的行和列同时进行采样,行和列的采样数目分别为p和q,输出一包含p行q列的子矩阵;

A22、根据第二范数计算行和列在整个矩阵范数中的比例,生成每一行和列的范数比例;

A23、对上述范数比例进行归一化处理后,得到样本取样的概率,并生成相应的概率区间;

A24、随机生成一大于0小于1的数,判断其是否在上述概率区间内,如是则抽取与其对应的样本。

4. 根据权利要求1所述的基于自适应分布式计算的在线推荐方法,其特征在于,所述步骤C中增量式在线更新方法包括对已知用户进行预测和对新用户/物品进行预测。

5. 根据权利要求1所述的基于自适应分布式计算的在线推荐方法,其特征在于,所述步骤D中所述加权集成中的权重是根据每一处理装置的处理能力来进行分配。

6. 根据权利要求5所述的基于自适应分布式计算的在线推荐方法,其特征在于,所述步骤D中加权集成中的权重是根据每一处理装置的处理能力来进行分配具体包括:

D1、每一处理装置的抽样的行列个数分别为 $c_j$ 和 $r_j$ ;

D2、则设置权重 $w_j = \frac{r_j}{\sum r_j} \times \frac{c_j}{\sum c_j}$ 。

7. 根据权利要求1或4所述的基于自适应分布式计算的在线推荐方法,其特征在于,所述处理装置为计算机,所述处理装置的处理能力包括计算机的内存和运算能力。

8. 一种基于自适应分布式计算的在线推荐系统,用于通过处理装置对海量数据信息处理后向用户进行推荐,其特征在于,所述在线推荐系统包括:

自适应负载均衡单元,用于根据处理装置的处理能力,采用基于范数的矩阵抽样算法从海量数据信息中抽样后进行分配,令每一处理装置能够独自处理分配的数据信息;

分布式矩阵分解单元,用于利用基于偏置量的矩阵分解模型对所分配的数据信息进行训练,得到一评分预测模型;

增量式在线更新单元,用于通过增量式在线更新方法更新所述评分预测模型;

在线集成推荐单元,用于通过加权集成更新后的评分预测模型获取对用户的最终推荐列表。

9. 根据权利要求8所述的基于自适应分布式计算的在线推荐系统,其特征在于,所述增量式在线更新单元中增量式在线更新方法包括对已知用户进行预测和对新用户/物品进行预测。

10. 一种移动终端,其特征在于,包括权利要求6所述的基于自适应分布式计算的在线推荐系统。

## 基于自适应分布式计算的在线推荐方法、系统和移动终端

### 技术领域

[0001] 本发明涉及智能推荐技术领域,尤其涉及一种基于自适应分布式计算的在线推荐方法、系统和移动终端。

### 背景技术

[0002] 如何从海量大数据中找到用户感兴趣的信息,如何让信息受到广大用户的欢迎,是一件非常困难的事情。推荐系统的任务就是联系用户和信息,帮助用户发现对自己有价值的信息,让信息能够展现在对它有兴趣的用户面前,从而实现信息消费者和信息提供者的双赢。

[0003] 推荐系统主要是通过分析用户的行为,对其建模,通过模型来预测用户的兴趣从而做出推荐。主要的方法可分为内容过滤,协同过滤,和基于矩阵分解的模型。内容过滤是在基于物品内容的基础上给用户推荐和他们之前喜欢的物品在内容上相似的其他物品。协同过滤通过分析用户的行为数据来找到相似的用户和相似的物品做出推荐。基于矩阵分解的模型是通过发现隐含的特征(比如类别)来联系用户兴趣和物品。这种模型在用户的行为数据上通过矩阵分解的方法来确定物品在这个类别中的权重,然后计算出用户对物品的感兴趣程度,从而对用户进行推荐。内容过滤和协同过滤的算法大部分都是在物品的内容或用户行为数据上的一些统计方法,而基于矩阵分解的模型是一种机器学习的方法,能更好的学习出用户和物品之间的关系,因此这种模型已经被广泛地应用到了目前主流的推荐系统中。

[0004] 虽然基于矩阵分解的模型效果好,但通常是作为一种离线的计算模型。因为它要求在内存里面加载整个数据,而且时间计算复杂度也很高。在普通机器上很难利用矩阵分解模型对海量的大数据部署推荐系统。目前也有提出利用分布式计算模型比如MPI(Message Passing Interface)或者MapReduce在大型的集群系统中快速地进行分布式矩阵分解运算,同时通过增量式模型实现在线更新和推荐。然而这种方法需要搭建高性能计算机,同时部署集群系统和分布式计算的框架复杂度高,且不利于系统的维护和扩展。

[0005] 有鉴于此,如何针对大数据设计一种快速、稳定、可靠、有效的基于矩阵分解的在线模型对当前智能推荐系统起着至关重要的作用。

### 发明内容

[0006] 鉴于现有技术中的不足,本发明目的在于提供一种基于自适应分布式计算的在线推荐方法和系统。旨在解决现有技术中智能推荐系统利用矩阵分解模型处理海量大数据时面临的计算复杂度高、维护扩展困难等问题。

[0007] 本发明的技术方案如下:

一种基于自适应分布式计算的在线推荐方法,用于通过处理装置对海量数据信息处理后向用户进行推荐,其中,所述在线推荐方法包括以下步骤:

A、根据处理装置的处理能力,采用基于范数的矩阵抽样算法从海量数据信息中抽样后

进行分配,令每一处理装置能够独自处理分配的数据信息;

B、利用基于偏置量的矩阵分解模型对所分配的数据信息进行训练,得到一评分预测模型;

C、通过增量式在线更新方法更新所述评分预测模型;

D、通过加权集成更新后的评分预测模型获取对用户的最终推荐列表。

[0008] 所述的基于自适应分布式计算的在线推荐方法,其中,所述步骤A中,采用基于范数的矩阵抽样算法具体包括以下步骤:

A1、获取海量数据信息对应的数据矩阵;

A2、对所述数据矩阵的行和列同时进行采样,得到一子矩阵;并根据向量的第一或第二范数来确保采样后的子矩阵包含的数据和海量数据之间的近似度小于预定的误差阈值。

[0009] 所述的基于自适应分布式计算的在线推荐方法,其中,所述步骤C中增量式在线更新方法包括对已知用户进行预测和对新用户/物品进行预测。

[0010] 所述的基于自适应分布式计算的在线推荐方法,其中,所述步骤D中所述加权集成中的权重是根据每一处理装置的处理能力来进行分配。

[0011] 所述的基于自适应分布式计算的在线推荐方法,其中,所述处理装置为计算机,所述处理装置的处理能力包括计算机的内存和运算能力。

[0012] 一种基于自适应分布式计算的在线推荐系统,用于通过处理装置对海量数据信息处理后向用户进行推荐,其中,所述在线推荐系统包括:

自适应负载均衡单元,用于根据处理装置的处理能力,采用基于范数的矩阵抽样算法从海量数据信息中抽样后进行分配,令每一处理装置能够独自处理分配的数据信息;

分布式矩阵分解单元,用于利用基于偏置量的矩阵分解模型对所分配的数据信息进行训练,得到一评分预测模型;

增量式在线更新单元,用于通过增量式在线更新方法更新所述评分预测模型;

在线集成推荐单元,用于通过加权集成更新后的评分预测模型获取对用户的最终推荐列表。

[0013] 所述的基于自适应分布式计算的在线推荐系统,其中,所述处理装置为计算机,所述处理装置的处理能力包括计算机的内存和运算能力。

[0014] 所述的基于自适应分布式计算的在线推荐系统,其中,所述增量式在线更新单元中增量式在线更新方法包括对已知用户进行预测和对新用户/物品进行预测。

[0015] 所述的基于自适应分布式计算的在线推荐系统,其中,所述在线集成推荐单元中所述加权集成中的权重是根据每一处理装置的处理能力来进行分配。

[0016] 一种移动终端,其中,包括上述的基于自适应分布式计算的在线推荐系统。

[0017] 有益效果:

本发明的基于自适应分布式计算的在线推荐方法、系统和移动终端,其中,所述推荐系统不需要通过任何分布式计算模型去建立集群。而且能够在单台或者多台普通处理装置上有效处理的处理海量数据,不仅稳定性好,可扩展性高,而且还可以大大节约成本和开发效率。

附图说明

[0018] 图 1 为本发明的基于自适应分布式计算的在线推荐方法的流程图。

[0019] 图 2 为本发明的基于自适应分布式计算的在线推荐系统的结构框图。

[0020] 图 3 为现有技术的基于矩阵分解的推荐系统的框架示意图。

[0021] 图 4 为本发明的基于自适应分布式计算的在线推荐系统的框架示意图。

### 具体实施方式

[0022] 本发明提供一种基于自适应分布式计算的在线推荐方法、系统和移动终端，为使本发明的目的、技术方案及效果更加清楚、明确，以下对本发明进一步详细说明。应当理解，此处所描述的具体实施例仅仅用以解释本发明，并不用于限定本发明。

[0023] 请参阅图 1，其为本发明的基于自适应分布式计算的在线推荐方法的流程图。所述基于自适应分布式计算的在线推荐方法，用于通过处理装置对海量数据信息处理后向用户进行推荐，如图 1 所示，所述在线推荐方法包括以下步骤：

S1、根据处理装置的处理能力，采用基于范数的矩阵抽样算法从海量数据信息中抽样后进行分配，令每一处理装置能够独自处理分配的数据信息；

S2、利用基于偏置量的矩阵分解模型对所分配的数据信息进行训练，得到一评分预测模型；

S3、通过增量式在线更新方法更新所述评分预测模型；

S4、通过加权集成更新后的评分预测模型获取对用户的最终推荐列表。

[0024] 下面分别针对上述步骤进行详细描述：

所述步骤 S1 为根据处理装置的处理能力，采用基于范数的矩阵抽样算法从海量数据信息中抽样后进行分配，令每一处理装置能够独自处理分配的数据信息。

[0025] 通常对大数据处理的算法都是基于 MapReduce 模型。首先对数据进行分片，然后在多台机器（即处理装置）上同时对分片数据进行处理，最后将各个在 Map 中相关的数据归结在一起。这种模型的优点在于第一，通过 MapReduce 这个分布式处理框架，不仅能用于处理大规模数据，而且能将很多繁琐的细节隐藏起来，比如，自动并行化、负载均衡和灾备管理等，这样将极大地简化程序员的开发工作；第二，MapReduce 的伸缩性非常好，也就是说，每增加一台服务器，其就能将差不多的计算能力接入到集群中，而过去的大多数分布式处理框架，在伸缩性方面都与 MapReduce 相差甚远。但缺点是：第一，将输入数据分隔成固定大小的片段，再由 MapReduce 平台处理，这样处理延迟与数据片段的长度、初始化处理任务的开销成正比。小的分段会降低延迟，增加附加开销，并且分段之间的依赖管理更加复杂（例如一个分段可能会需要前一个分段的信息）；反之，大的分段会增加延迟。最优化的分段大小取决于具体应用；第二，为了支持流式处理，MapReduce 需要被改造成 Pipeline 的模式，而不是 Reduce 直接输出；考虑到效率，中间结果最好只保存在内存中等等。这些改动使得原有的 MapReduce 框架的复杂度大大增加，不利于系统的维护和扩展。

[0026] 而在本发明中，采用一种动态自适应性负载均衡技术，也就是根据各个计算机的处理能力来分配数据片，每台处理装置（在本实施例中，所述处理装置为计算机）都可以独立地在它所分配的数据上参与计算，而彼此间不需要通过消息机制来通信。这样做的好处是：第一，如果某台计算机出现了故障，其他的计算机依然可以独立的进行运算；第二，由于我们不需要建立集群，所以任何一台计算机都可以被充分地利用起来，这样可以大大节

约成本;第三,可扩展性好,任何一台计算机都可以无缝地接入在这个系统来,同时却没有带来额外的运营和管理的开销。

[0027] 另外,对于海量数据信息(即数据信息量足够大),采用基于矩阵范数的随机抽样理论上可以保证尽可能的近似原始数据,同时所抽样的数据也具有一定的代表性。以电影推荐网站为主,成千上万部电影中评价好的,被观看次数多的电影通常才几千而已,有统计显示活跃的用户平均观看的电影数目也不超过 3000 部,于是我们可以只在这些抽样出的电影中利用目前现有的矩阵分解的方法给用户做有效的推荐。在本实施例中,采用基于范数的矩阵抽样算法具体包括以下步骤:

S11、获取海量数据信息对应的数据矩阵;

S12、对所述数据矩阵的行和列同时进行采样,得到一子矩阵;并根据向量的第一或第二范数来确保采样后的子矩阵包含的数据和海量数据之间的近似度小于预定的误差阈值。

[0028] 具体来说,我们采用如下的基于范数的随机取样算法:

首先输入:原始矩阵  $A \in m \times n$ , 行和列的采样数目为  $p$  和  $q$ ;然后输出包含  $p$  行  $q$  列的子矩阵,同时根据第二范数计算行和列在整个矩阵范数中的比例,以列的抽样为例(行的

抽样类似)按照如下方法生成每一列的范数比例  $R = r_{A_1}, \dots, r_{A_2}, \dots, r_{A_n}$ , 比如  $r_{A_j} = \frac{\sqrt{\sum_i A_{ij}^2}}{\sqrt{\sum_i \sum_j A_{ij}^2}}$ 。对

$R$  进行归一化处理:  $R_j = R_j / \sum_j R_j$ , 归一化后的值就可以作为该样本取样的概率,按照如下方

法生成概率区间  $P: p_j = \sum_{k=1}^{j-1} p_k$  随机生成在  $[0, 1]$  之间的一个数,如果在  $P$  的某个区间,就抽取出对应的那个样本。

[0029] 所述步骤 S2 为利用基于偏置量的矩阵分解模型对所分配的数据信息进行训练,得到一评分预测模型。

[0030] 传统的基于 SVD (Singular Value Decomposition) 的矩阵分解方法需要将数据全部导入在内存中,而且计算复杂度是非线性的,根本无法处理海量的数据。在推荐系统中普遍使用的矩阵分解模型大部分采用的 SGD (Stochastic Gradient Descent) 来训练数据,通过这样的方法,矩阵分解模型能有效地解决数据的稀疏性问题,但对大数据处理能力依然有限。最近有文献提出了通过分割矩阵然后利用 MapReduce 的分布式计算模型来并行处理矩阵分解的问题,但这对硬件提出了很高的要求,同时也需要非常熟悉如何使用这些分布式计算模型来有效地进行计算。

[0031] 而在本发明中,通过自适应性负载均衡可以实现不需要建立集群系统就可以进行分布式计算的任务,而且每台机器都能根据自身的能力单独地进行有效的计算。因为我们利用矩阵抽样模型可以实现从数据矩阵中抽取出少量的行和列就可以近似的表达原始矩阵的信息。基于这个理论,可以在抽样后的数据矩阵上独立地进行矩阵分解运算。这样只需要在每台普通的机器上保存一定的样本数据既可以快速得到一个评分预测模型。这种评分预测模型时间和空间复杂度都不是很高,而且也可以保证推荐系统的有效性和稳定性。因此这种分布式矩阵分解模型对海量的大数据有着高效的处理能力。其具体做法如下:因为用户行为数据通常包含几个非常重要的信息:所有记录的评分的全局平均值,用户评分的偏置量和物品接受评分的偏置量。通过隐类和偏置量来关联用户和物品的模型称为基

于偏置量的矩阵分解。故此,在本发明的技术方案中采用一种基于偏置量的矩阵分解模型 SVDFeature,描述为:

$$y = \mu + \left( \sum_j b_j^{(g)} \gamma_j + \sum_j b_j^{(u)} \alpha_j + \sum_j b_j^{(i)} \beta_j \right) + \left( \sum_j p_j \alpha_j \right)^T \left( \sum_j q_j \beta_j \right)$$

$y$  是目标评分,  $\mu$  是平均分,  $b_j^{(g)}$  是全局特征偏置量,  $b_j^{(u)}$  是用户偏置量,  $b_j^{(i)}$  是物品偏置量,  $p_j$  是用户隐性特征,  $q_j$  是物品隐性特征。基于 SGD 的方法,可以求出这些偏置量的相关参数,如表 1 所示:

$$\begin{aligned} p_i &= p_i + \eta \left( \tilde{e} \alpha_i \left( \sum_j q_j \beta_j \right) - \lambda_1 p_i \right) \\ q_i &= q_i + \eta \left( \tilde{e} \beta_i \left( \sum_j p_j \alpha_j \right) - \lambda_2 q_i \right) \\ b_i^{(g)} &= b_i^{(g)} + \eta \left( \tilde{e} \gamma_i - \lambda_3 b_i^{(g)} \right) \\ b_i^{(u)} &= b_i^{(u)} + \eta \left( \tilde{e} \alpha_i - \lambda_4 b_i^{(u)} \right) \\ b_i^{(i)} &= b_i^{(i)} + \eta \left( \tilde{e} \beta_i - \lambda_5 b_i^{(i)} \right) \end{aligned}$$

表 1

所述步骤 S3 为通过增量式在线更新方法更新所述评分预测模型。

[0032] 在用户生成推荐列表时,需要计算用户对所有物品的兴趣权重,然后排名,返回权重最大的 N 个物品。那么,在物品数目很多时,这一过程的时间复杂度非常高,生成一个用户推荐列表的速度很慢,因此不能在线实时计算,而需要离线将所有用户的推荐结果事先计算好存储在数据库中。因此,矩阵分解模型通常都是静态的方法,不能进行在线实时推荐,也就是说,当用户有了新的行为后,他的推荐列表不会发生变化。而在实际的应用中,用户的数据是不断地增长的,因此在线推荐十分重要。目前文献提出了一些在线的矩阵增量式模型能够快速更新已经计算好的单个模型,但是如何在分布式计算的环境中,有效地处理更新问题对整个推荐系统效果起着很重要的作用。

[0033] 在本发明中,将单个矩阵分解模型的增量式在线更新扩展到分布式计算环境中。再次,需要考虑两种情况:第一,已知的用户;第二,新的用户或者新的物品。对已知的用户(已经在某机器上训练过的用户),本方案采用一个快速的更新策略直接进行在线更新;对新的用户(包括已知的用户但没有在该机器上训练过)和新的物品,则采用另外一个快速的更新策略来进行模型更新。这种在分布式计算环境中提出的在线更新模型能够更加有效地对模型进行快速更新,从而能够高效地加快在线的推荐的速度,同时还能有效地解决系统冷启动问题,包括如果对新加入的用户推荐物品以及对新加入的物品推荐给用户。

[0034] 其具体做法如下:结合步骤 S2 得到的基于偏置量的矩阵分解模型,再加入全局特征,用户特征和物品特征来对针对用户的喜欢做出个性化推荐,然而这个模型本身不支持增量式更新。也不能分布式地并行计算,因此该模型依然无法处理海量的数据。我们在这个基础上引入分布式计算模块就可以很好的解决这个问题,因为本发明的模型是建立在单个普通机器上对抽样数据进行训练的基础上,因此可以利用下两种在线更新的算法对每个



机器上的模型进行更新。更新有两种操作：第一，对已有的用户进行预测；第二，对新的物品和新的用户进行预测。如下表 2 所示，其中， $S$  为抽样后的子矩阵， $W$  为物品特征矩阵， $H$  为用户的特征矩阵。对上述的偏置量也可采用相似的方式来进行更新，如表 3 所示。

```

1: procedure ADDRATING( $S, W, H, r_{u,i}$ )
2:    $S \leftarrow S \cup \{r_{u,i}\}$ 
3:   return UPDATERATING( $S, W, H, r_{u,i}$ )
4: end procedure

5: procedure REMOVRATING( $S, W, H, r_{u,i}$ )
6:    $S \leftarrow S \setminus \{r_{u,i}\}$ 
7:   return UPDATERATING( $S, W, H, r_{u,i}$ )
8: end procedure

9: procedure UPDATERATING( $S, W, H, r_{u,i}$ )
10:  if  $P_u(\text{train}|r_{u,i}) > \text{RANDOM}$  then
11:     $(W, H) \leftarrow \text{USERRETRAIN}(S, W, H, u)$ 
12:  end if
13:  if  $P_i(\text{train}|r_{u,i}) > \text{RANDOM}$  then
14:     $(W, H) \leftarrow \text{ITEMRETRAIN}(S, W, H, i)$ 
15:  end if
16:  return  $(W, H)$ 
17: end procedure

```

[0035] 表 2

```

1: procedure USERUPDATE( $S, W, H, r_{u,i}$ )
2:    $S \leftarrow S \cup \{r_{u,i}\}$ 
3:   return USERRETRAIN( $S, W, H, u$ )
4: end procedure

5: procedure USERRETRAIN( $S, W, H, u^*$ )
6:   initialize  $u^*$ -th row in  $W$ 
7:   repeat
8:     for  $r_{u,i} \in C(u^*, \cdot)$  do
9:       for  $f \leftarrow 1, \dots, f$  do
10:         $w_{u,f} \leftarrow w_{u,f} - \alpha \frac{\partial}{\partial w_{u,f}} \text{Opt}(S, W, H)$ 
11:      end for
12:    end for
13:  until Stopping criteria met
14:  return  $(W, H)$ 
15: end procedure

```

表 3

所述步骤 S4 为通过加权集成更新后的评分预测模型获取对用户的最终推荐列表。

[0036] 目前的针对大数据的矩阵分解算法由于都是基于 MapReduce 模型,所以必须将所有数据归结在一起之后才能给出最终的预测评分,这样大大限制了它的功效。如果有一个节点出现故障,对最终的结果都会造成一定的影响。

[0037] 而本发明提出的集成推荐的模型就是为了解决如何在这个分布式计算框架中更加有效地获取推荐结果。比如说某台机器出现故障的时候,如何保证整个系统有效快速的给出推荐结果。本发明提出的集成推荐实际上是对每台机器预测评分的一个加权求和。因为每台机器处理数据的能力不一样,则相应训练出来的模型给出的推荐作用也会不一样,而且在增量式的在线更新中,每个模型更新的作用也有差别。而通过加权每台机器上获得的推荐结果可以最终得到一个比较合理的全局推荐结果。理想情况下,每台机器都能及时的给出推荐结果,这样加权后得到的结果最好,但是如果在对时效性要求很高的应用中,这种模型也可以获得相对好的推荐结果。因为每台机器上的矩阵分解模型都可以单独来做出推荐,这样可以保证只要有一台机器能正常运行都可以给出一定程度的推荐。而且针对某个用户还可以选择性的给出一些比较好的模型从而做出推荐。因此,本发明的集成推荐是一种可扩展性高,稳定性好的推荐模型。

[0038] 具体来说,集成推荐主要是对所有的机器上得到的推荐结果进行加权求和,权重主要是根据每台机器的处理能力来计算的,具体就是根据每台机器抽样的多少来赋予权重。假设每台机器抽样的行列个数分别为  $c_j$  和  $r_j$ ,则每台机器的权重可以简单计算为

$w_j = \frac{r_j}{\sum r_j} \times \frac{c_j}{\sum c_j}$ , 于是集成推荐的计算公式为:

$$R_i = \sum_j w_j \cdot y_j ;$$

其中,  $y_j$  为第 j 台机器做出的评分预测。最后只要对  $R_i$  进行排序就可以给出 Top K 的推荐了。集成推荐能够平衡各台机器的处理能力并且高效快速的在线给出推荐。如果为了进一步提高效率,还可以采用命中就推荐的方式,比如说某个已有的用户,只要在每台或者几台机器上抽样的数据中包含了这个用户,就可以略过哪些没有该用户抽样数据的机器。

[0039] 本发明还提供了一种基于自适应分布式计算的在线推荐系统,用于通过处理装置对海量数据信息处理后向用户进行推荐,如图 2 所示,所述在线推荐系统包括:

自适应负载均衡单元 100,用于根据处理装置的处理能力,采用基于范数的矩阵抽样算法从海量数据信息中抽样后进行分配,令每一处理装置能够独自处理分配的数据信息;

分布式矩阵分解单元 200,用于利用基于偏置量的矩阵分解模型对所分配的数据信息进行训练,得到一评分预测模型;

增量式在线更新单元 300,用于通过增量式在线更新方法更新所述评分预测模型;

在线集成推荐单元 400,用于通过加权集成更新后的评分预测模型获取对用户的最终推荐列表。

[0040] 上述系统中各个部分的功能都已经在上述方法中进行了详细介绍,这里就不再冗述了。

[0041] 通常基于矩阵分解模型的框架主要分为在线和离线计算两个部分,如图 3 所示。在线部分主要是根据已有的模型来对用户喜好进行计算、排序和过滤从而给出推荐结果。离线部分主要是在用户的行为数据、用户本身的信息以及物品本身的信息基础上生成矩阵分解的推荐模型,如果有新的用户行为数据、新增的用户或者物品的信息,都在离线部分进行更新然后重新调用矩阵分解的模型来进行模型的训练。可以看出这种模型不太适合在线推荐,因为它无法及时地对新增的用户行为数据,或者新增的用户,或者新增的物品给出有效的推荐。同时对已有的用户,也要通过计算出他对所有物品的喜好,然后排序、过滤才能给出最终的推荐结果,由于物品的数量庞大,这个计算会比较耗时。如果在线用户数量可观的话,这个计算量会非常大,系统的吞吐量也会越来越小。

[0042] 本发明提出的自适应性分布式计算的在线智能推荐系统,如图 4 所示。自适应性分布式计算主要是通过每台机器自适应性地将大数据抽样后训练矩阵分解模型来实现的,然后自发地进行增量式的模型在线更新,最后对各个模型进行加权集成推荐。和图 3 中现有技术中的推荐系统最大的不同在于分布式和在线化:矩阵分解和模型更新建立在自适应性分布式计算的框架上,而除了矩阵分解训练模型在离线来进行,模型的更新和推荐由于这个框架上的优越性而实现了在线化。

[0043] 概括来说,本发明提出了一种有效可行的自适应性分布式计算的在线智能推荐方案。其第一次创新性地提出了自适应性分布式计算框架,可自适应性分布式计算是指在由普通的机器组成的系统中每台计算机可以根据自己的处理能力自动去做负载均衡从而实现分布式计算。这种推荐系统不需要通过任何分布式计算模型去建立集群。而且能够在单台或者多台普通计算机上有效处理的处理海量数据,不仅稳定性好,可扩展性高,而且还可以大大节约成本和开发效率。同时这个方案有效地利用了矩阵分解模型,并且快速在分布式计算环境中对模型进行更新,最终通过集成各个模型给出比较理想的推荐结果。通过这个方案,只需要利用一台或者多台普通计算机给出一个高效处理海量数据的推荐解决方案。

[0044] 另外,本发明还提供了一种移动终端(如手机),其设置有上述的基于自适应分布式计算的在线推荐系统,令用户可以通过移动终端随时随地的获取推荐信息。

[0045] 综上所述,本发明的基于自适应分布式计算的在线推荐方法、系统和移动终端,其中,自适应性分布式计算主要是通过每一处理装置自适应性地将大数据抽样后训练矩阵分解模型来实现的,然后自发地进行增量式的模型在线更新,最后对各个模型进行加权集成推荐。这种推荐系统不需要通过任何分布式计算模型去建立集群。而且能够在单台或者多台普通计算机上有效处理的处理海量数据,不仅稳定性好,可扩展性高,而且还可以大大节约成本和开发效率。概括来说,本发明提出的基于矩阵分解的自适应性分布式计算的在线智能推荐系统有如下一些好处:

易部署性:矩阵抽样的自适应性负载均衡模型可以保证任何一台计算机都可以用来做推荐;

可扩展性和稳定性:我们不需要搭建集群系统,而且每台计算机之间不需要通信,如果一台机器出现故障,其他机器同样可以作出推荐。

[0046] 多样性:避免出现“长尾效应”,即如果只是提取出最受好评的几千部电影,那么不广泛流行的电影被观看的机会会越来越小。自适应性负载均衡模型可以在最大程度上保证

推荐的多样性。

[0047] 实时性 :为了保证推荐的实时性,我们引入了增量式在线更新模型,可以快速更新已有用户的推荐列表并作出推荐。

[0048] 准确性 :基于偏置量矩阵分解算法可以更好的关联物品和用户之间的关系,从而获取比较准确的推荐效果,而且通过集成推荐模型可以进一步优化推荐效果。

[0049] 应当理解的是,本发明的应用不限于上述的举例,对本领域普通技术人员来说,可以根据上述说明加以改进或变换,所有这些改进和变换都应属于本发明所附权利要求的保护范围。

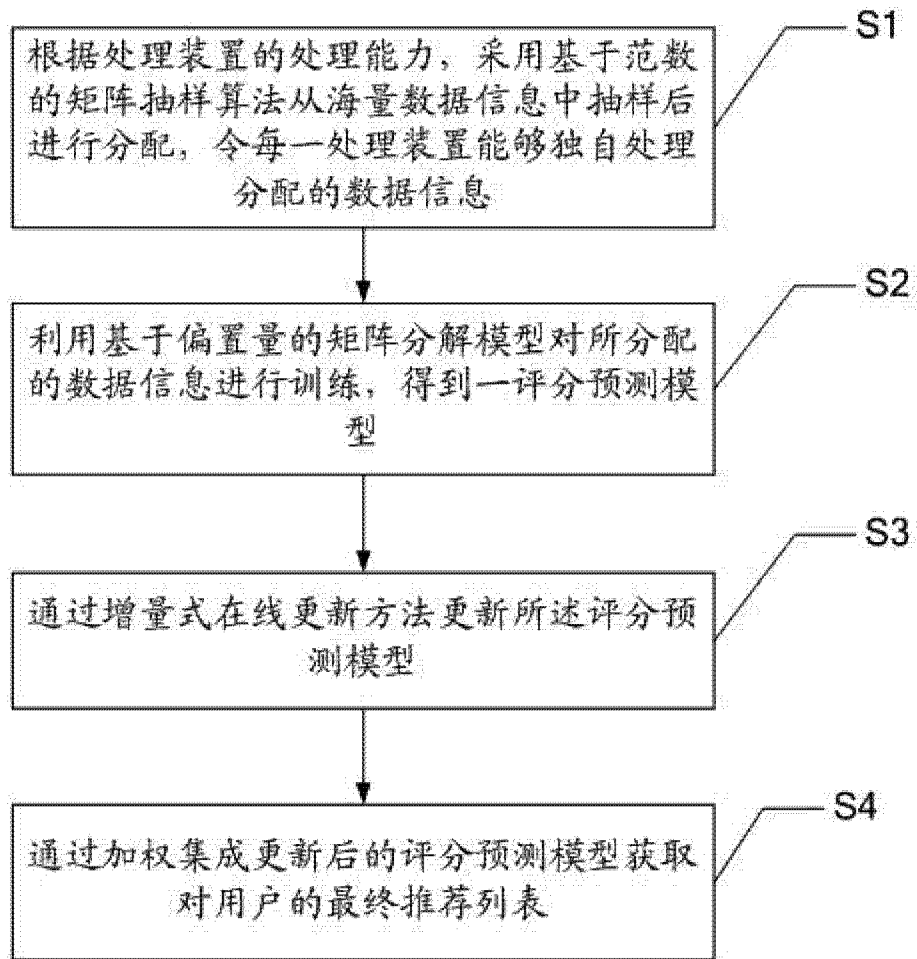


图 1



图 2

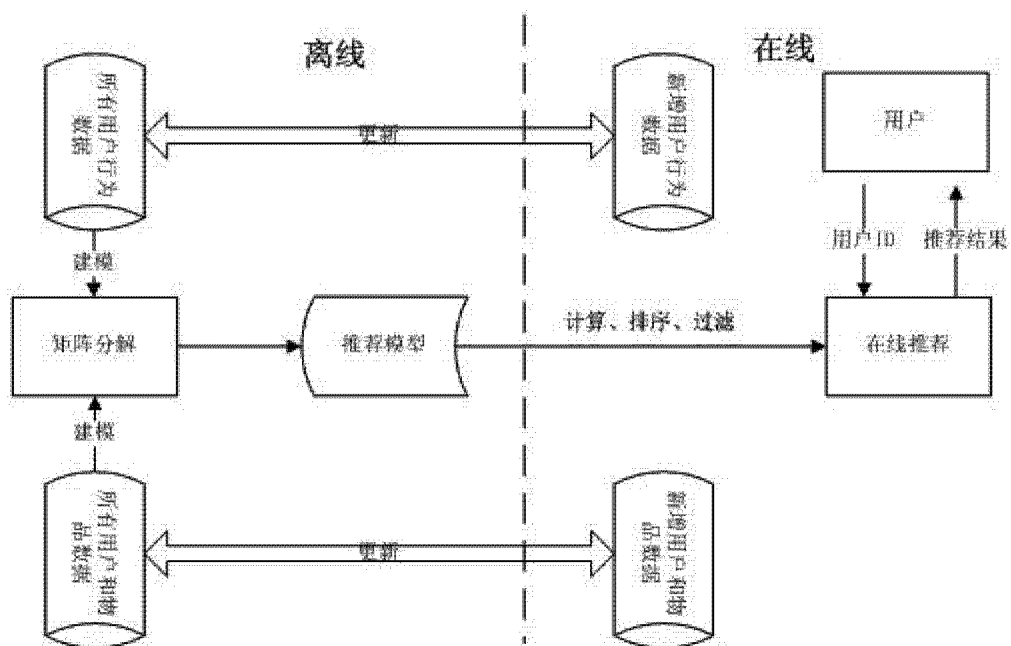


图 3

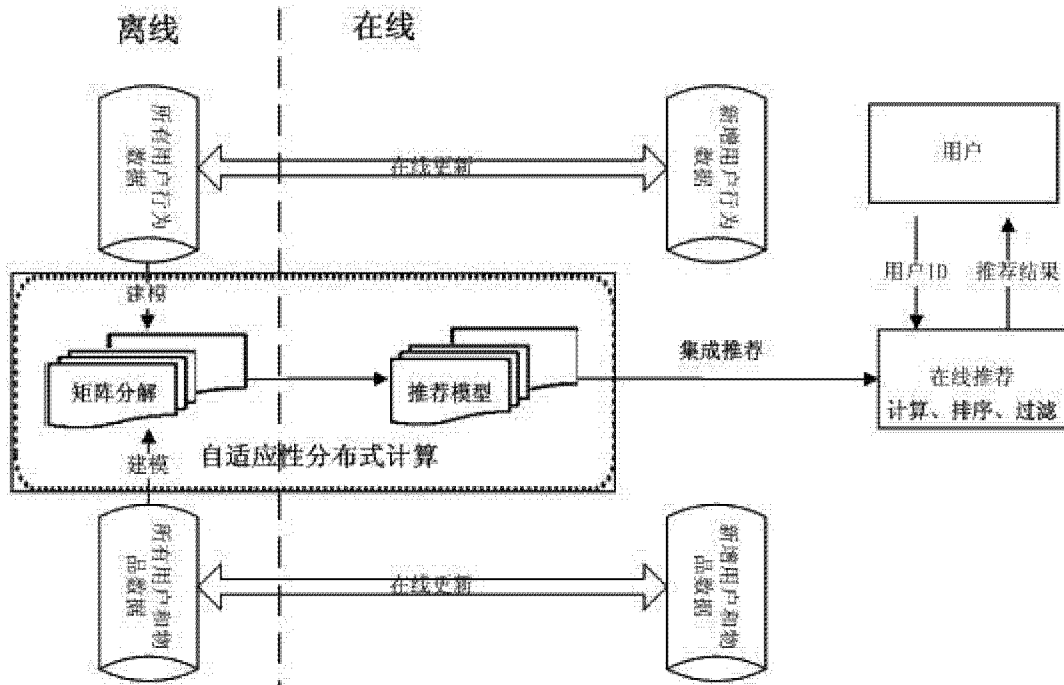


图 4