文章编号:1001 - 9081(2007)10 - 2598 - 04

二阶段近似 KNN 离群挖掘算法与应用

林甲祥,樊明辉,陈崇成,江先伟
(福州大学 空间数据挖掘与信息共享教育部重点实验室, 福州 350002)
(linjx2000@gmail.com)

摘 要:针对高维大数据集,提出了二阶段近似最近邻离群挖掘算法(TPOM),在聚类的基础上,通过加速最近邻查询和改善剪枝效率,提高了循环嵌套 KNN 算法的离群检测效率。应用分析表明,该算法对于实际数据集有良好的适用性和可扩展性,具有近似线性的时间复杂度。

关键词:二阶段近似最近邻离群挖掘算法;基于距离的离群;近似最近邻;k 均值聚类
中图分类号: TP311.131 **文献标志码**:A

Algorithm of two-phase approximate KNN outliers mining and its application

LIN Jia-xiang, FAN Ming-hui, CHEN Chong-cheng, JIANG Xian-wei
(*Key Laboratory of Spatial Data Mining & Information Sharing of Ministry of Education, Fuzhou University, Fuzhou Fujian* 350002, *China*)

Abstract: In this paper, we targeted at high-dimensional datasets and presented Two-Phase Approximate KNN Outliers Mining (TPOM), an algorithm of two-phase based approximate KNN outliers mining. It processed clustering on the datasets firstly, and then improved the efficiency of nested loop KNN outliers detecting by accelerating the nearest neighbors search and improving the cut-off efficiency. Application and the result show that TPOM is well suit and extendible to real data, it scales log-linearly as a function of the number of data points and linearly as a function of the number of dimensions.

Key words: Two-Phase Approximate KNN Outliers Mining (TPOM); distance-based outlier; approximate KNN; k-means clustering

0 引言

离群检测是数据挖掘的一个重要研究方向,目标是从数据集中发现潜在有用的与其他大部分数据显著不同的数据。近年来研究人员提出了很多离群检测算法,按照算法采用的机制,大致可以分为基于统计的方法、基于深度的方法、基于距离的方法、基于密度的方法、基于偏离的方法和基于聚类的方法[1,2]。

自从文献[3]提出基于距离的离群定义后,基于距离的离群挖掘研究已经取得了一些重要的进展。作为一种实用的离群检测技术,基于距离的定义为离群提供了量化描述,不同的参数可以表示所有基于统计的离群[4]。目前,基于距离的离群已经成为很多离群挖掘问题研究的基础和出发点,被广泛应用于各种挖掘算法的构造中,一些优秀的算法在实际应用中已经取得了较好的成果。依据算法采用的策略,基于距离的方法大致分为循环嵌套的方法、索引的方法和单元划分的方法[1]。然而传统的基于距离的方法没有对占数据集大多数的正常数据进行必要的剪枝,因此算法的效率通常都不尽如人意,特别是面对高维大数据集,算法大多具有 $O(n^2 \times d)$ 或以上的时间复杂度。

本文在现有基于距离的离群检测方法的基础上,针对算法在高维大数据集上效率低下的问题,改进了传统的循环嵌套 KNN 算法,提出二阶段近似最近邻离群挖掘算法。算法分两个阶段实现离群数据的挖掘,首先对目标数据集进行大致聚类,然后在聚类数据上进行近似最近邻离群检测。本文最后利用福建省海洋环境质量监测数据对算法的效率和适用性进行验证。

1 基于距离的离群挖掘算法分析

基于距离的离群是指那些与数据集中其他大多数数据的距离大于某个阈值的数据,可量化描述为:d 维数据集 $D = \{x_1, x_2, \cdots, x_n\}$ 中的对象 x_i,如果至少存在 $p\%$ 的其他对象到 x_i 的距离超过阈值 δ,则 x_i 为基于距离的离群,记为 $DB(p,\delta)$。

大量文献中的离群挖掘研究表明,传统的基于距离的离群检测方法存在一些缺陷。首先,参数 p 和 δ 很难确定,因为 $DB(p,\delta)$ 与数据集本身的特性和具体的挖掘任务有关,对于不同参数值,挖掘结果具有很大的不稳定性,用户必须结合领域知识并反复测试,才能确定一个满意解;其次,基于距离的离群是一个二值属性,某个数据要么是离群,要么不是,无法描述离群数据的奇异程度;最后,算法的复杂度较高,若 n 表示 d 维数据集 D 中对象的数目,则基于距离的方法通常具有 $O(n^2)$ 的时间复杂度。其中,基于索引的方法的时间复杂度为 $O(n^2 \times d)$,但索引建立的开销很大,算法没有竞争性。基于循环嵌套的方法的时间复杂度为 $O(n^2 \times d)$,虽然不需要建立多维索引结构,但还是比较费时。基于单元的方法的时间复杂度为 $O(c^d + n)$,c 为取决于单元划分的一个常数,但算法不能处

收稿日期:2007 - 04 - 04;**修回日期**:2007 - 07 - 06。
基金项目:国家自然科学基金资助项目(60602052);福建省重点科技项目(2005H086);福建省自然科学基金资助项目(2006J0131)。
作者简介:林甲祥(1982 -),男,福建安溪人,硕士研究生,主要研究方向:网络共享技术、空间数据挖掘与信息可视化; 樊明辉(1974 -),男,湖北黄冈人,副教授,博士,主要研究方向:数据挖掘、交互式可视化技术; 陈崇成(1968 -),男,福建闽清人,教授,博士,主要研究方向:空间数据挖掘、智能决策支持系统; 江先伟(1975 -),男,福建连城人,硕士研究生,主要研究方向:聚类分析、数据挖掘。

理高维数据集,当数据维数 $d \leqslant 4$ 时,算法的优越性在 n 越大时越明显,而当数据维数 $d \geqslant 5$ 之后,循环嵌套算法开始显现其优势[5]。

针对基于距离的方法存在的上述问题,已有学者提出了一些解决措施。例如,利用特定的索引结构(KD-trees、R-trees、X-trees[6]等)对属性空间进行分区,从而加速对象的最近邻查询,但高维数据中,索引建立与遍历的效率非常差,索引结构被证明是无效的[7]。为解决高维数据集上算法的效率与可行性问题,文献[8]提出了基于距离的 k 最近邻(简称 KNN)离群检测算法。若以 $D_k(x)$ 表示对象 x 的第 k 个最近邻的距离,KNN 算法的主要思想是寻找 $D_k(x)$ 值最大的若干个对象作为离群。研究表明 KNN 算法是高维有效的离群挖掘方法,然而在高维大数据集上,算法的效率还是不尽如人意,具有 $O(n^2 \times d)$ 的时间复杂度[9]。因此,本文对传统的循环嵌套 KNN 算法进行改进,提出了二阶段近似最近邻离群挖掘算法(Two-Phase Approximate KNN Outliers Mining, TPOM)。

2　TPOM 算法及其构造

2.1　算法思想

对象的最近邻是指那些重要属性上最相似的数据,若把数据的属性特征个数作为维数,则离群点是指数据空间中不能与其他对象聚合在一起的对象。循环嵌套 KNN 算法中,为检测离群数据,需要找出每个对象的 k 最近邻,然而,对于占数据集绝大多数的正常数据,只须找出阈值 θ 范围内的 k 个邻近对象,而不必找出对象真正的 k 个最近邻,就可以确定数据为非离群点。若把对象的这些邻近点称为对象的近似最近邻[10],那么离群检测问题的关键在于如何高效地找出正常数据的近似最近邻,并把对象从可能的离群集中剔除。

本文借鉴 KNN 算法的思想,采用近似最近邻判别的策略,对简单的循环嵌套 KNN 算法进行改进,提出了二阶段近似最近邻离群挖掘算法(TPOM)。以简单的欧氏距离 $d(x,y)$ 作为对象间相似度的衡量标准。

$$d(x,y) = \sqrt{\sum_{i=1}^{d}(x_i - y_i)^2};\ x = (x_1,x_2,\cdots,x_d), y = (y_1,y_2,\cdots,y_d) \tag{1}$$

TPOM 算法分两个阶段对数据集进行离群检测。首先对数据集进行大致聚类,使得相似的对象尽可能处于同一个类别中;然后在聚类结果中进行对象的近似最近邻查询,通过正常数据的高效剪枝[11],实现改进的循环嵌套 KNN 离群检测算法。对于数据集中的大部分正常数据,由于算法只须对数据所在的类别进行近似最近邻搜索而不须扫描整个数据集,因此减少了数据集的扫描次数,使得 TPOM 算法具有近似线性的时间复杂度。

2.2　算法实现

以 $D = \{x_1,x_2,\cdots,x_n\}$ 表示待挖掘的目标数据集,n 表示数据对象的数目,d 表示属性特征个数,$Y = \{y_i \mid y_i = (y_{i1}, y_{i2},\cdots,y_{id})^T, \forall x_i \in D\}$ 表示 d 维数据对象的属性集,下文分别针对算法的两个阶段进行详细描述。

1)第一阶段

对数据集进行大致聚类,使得数据集中相似度高的对象尽可能处于同一个类别。

目前,文献中存在大量的聚类算法,如基于层次的算法(CHAMELEON,CURE,BIRCH)、基于划分的算法(K-MEANS,FREM)、基于密度的算法(DENCLUE,OPTICS,DBSCAN)、基于网格的算法(STING,CLIQUE,WAVECLUSTER)、以及基于模型的算法(COBWEB,CLASSIT,AUTOCLASS)等[12]。虽然 k-均值方法存在着一些缺陷,例如需要用户指定类别数、无法进行等频划分[1]等,但由于离群检测的目的在于挖掘离群而非聚类,并且第一阶段只须对数据集进行大致分类而不是最优分类,因此,本文借鉴 k-均值聚类的思想,提出了变形的聚类算法——VK-means,与 k-均值聚类算法不同的是,VK-means 算法采用"若对象与现有类别差异足够大,则分裂到新的分类;若类别之间的距离足够小,则合并两个类别"的机制,聚类结果的类别数目可能比用户指定的类别数 k 来得多,实际工作中,通常采用限制类别大小的方法来控制类别的扫描次数。VK-means 算法描述如下:

步骤 1　初始化聚类结果集 $R = \varnothing$;

步骤2　指定初始的类别数目 k,最终类别数 $k' = k$,并随机选择 k' 个数据作为中心;

步骤 3　对数据集 D 中未指定分类的对象 x_i,根据对象到各中心的最小距离 $\min(x_i,C)$,若 $\min(x_i,C)$ 足够大,则将对象 x_i 视为新分类的一个元素,$k' = k' + 1$,并跳转步骤3;否则把对象归入相应的类别中,并重新计算类别的中心及其类别之间的最小距离 $\min(C)$。若 $\min(C)$ 足够小,则合并相近的两个类别;否则跳转步骤3。

步骤4　计算目标函数 $f(D)$,若 $f(D)$ 的值小于用户指定的阈值 σ,则完成初始聚类,否则以各类别的中心为初始中心,跳转步骤3进行新一轮的归类;

步骤5　对每个类别进行判断,若其对象数目小于最大值 max,则把类别添加到结果集 R 中,否则递归调用算法进行类别细分;

步骤6　返回 R;

其中,k 表示用户指定的初始类别数目,$C = (c_1,c_2,\cdots,c_k)$,$c_k \in R^d$ 表示聚类的 k 个初始中心,$\min(x_i,C)$ 表示对象 x_i 与类别 C 之间的最小距离:

$$\min(x_i,C) = \min\{dist(x_i,c_j)\};\ j = 1,2,\cdots,k \tag{2}$$

$\min(C)$ 表示类别 C 之间的最小距离:

$$\min(C) = \min\{dist(c_i,c_j)\};\ i,j = 1,2,\cdots,k\&i \neq j \tag{3}$$

$f(D)$ 表示聚类的目标函数:

$$f(D) = \sum_{j=1}^{k'}\sum_{x_i \in C_j}(x_i - c_j)^2 \tag{4}$$

其中,C_j 为聚类中心 c_j 所代表的类。

VK-means 算法循环的过程是一个类别内直径变小、类别间差异性变大的过程。但选择不同的初始聚类中心,得到的聚类结果可能不用。对于海量数据,穷举所有可能的组合以获得最优的聚类结果是不现实的,实际工作中,通常在小样本数据中进行大致聚类,从而获得相对较优的初始聚类中心。

2)第二阶段

在聚类结果的基础上,采用基于循环嵌套的近似 KNN 算法对数据进行离群检测。传统的循环嵌套 KNN 算法通过遍历所有对象获得 k 最近邻,检测离群需要获得所有对象的 k 最近邻,因此循环嵌套 KNN 算法需要计算所有对象两两之间的距离,时间复杂度为 $O(n^2)$。本文对循环嵌套 KNN 算法进行改进,在第一阶段得到的聚类结果中,相似对象通常落在同一个类别中,因此第二阶段搜索对象的近似最近邻时,通常只须在对象所在的类别中进行。对于占数据集绝大多数的正常数据,一般在此步骤中被剪枝;对于少量的离群和边界数据,首先在离群数据所在的类别中进行近似 k 最近邻搜索,若没有

找到,则依次调入各个类别,直到找出对象的 k 个最近邻。由于离群数据相对较少,第二阶段为搜索对象的 k 最近邻而扫描整个数据集的次数是有限的,因此,算法具有近似线性的时间复杂度,最坏情况是每个对象需要扫描整个数据集才能确定其近似最近邻,最差时间复杂度为 $O(n^2)$。TPOM 算法描述如下:

步骤 1 初始化离群集 $OS = \varnothing$、剪枝阈值 $\theta = 0$;

步骤 2 对 D 中未被检测的对象 d,令 d 的 k 最近邻域 $Neighbor(d) = \varnothing$;

步骤 3 对 D 中除 d 以外未被检测的其他对象 b,计算 $dist(a,b)$;

步骤 4 若 d 的最近邻数目少于 k,则 $Neighbor(d) = Neighbor(d) \cup b$; 反之,若 $dist(d,b) < Maxdist(d, Neighbor(d))$,则更新对象的邻域 $Neighbor(d) = Closest(d, Neighbor(d) \cup b, k)$,此时,若 $Maxdist(d, Neighbor(d)) < \theta$,则 d 不是离群,被剪枝,跳转步骤 2;

步骤 5 若离群点数目少于 m,则 $OS = OS \cup d$;否则离群点数目已经达到 m,此时,若 d 的离群率比现有离群集 OS 中离群率最小的数据的离群率还大,即 $Maxdist(d, Neighbor(d)) > \theta$,则更新离群集 $OS = TopOutlier(OS \cup d, m)$,并更新剪枝阈值 $\theta = MinThreshold(OS)$;

步骤 6 返回离群集 OS;其中,函数 $Maxdist(d,S)$ 返回点 d 与对象集合 S 中的点 s_i 之间的最大距离值:

$$Maxdist(d,S) = \max\{dist(d,s_i)\};\ s_i \in S \tag{5}$$

函数 $Closest(d,S,k)$ 返回对象集合 S 中距离点 d 最近的 k 个点,若数据的个数少于 k,则返回整个集合:

$$Closest(d,S,k) = \{x_1, x_2, \cdots, x_k\} \tag{6}$$

其中 $x_j, j = 1 \cdots k$ 为 S 中与 d 距离最小的 k 个点。

函数 $TopOutlier(S,m)$ 返回对象集合 S 中数值最大的前 m 个点,若数据的个数少于 m,则返回整个集合:

$$TopOutlier(S,m) = \{x_1, x_2, \cdots, x_m\} \tag{7}$$

其中 $x_j, j = 1 \cdots m$ 为 S 中数值最大的 m 个点。

函数 $MinThreshold(S)$ 返回数值集合 S 中的最小值:

$$MinThreshold(S) = \min\{x_i\};\ x_i \in S \tag{8}$$

2.3 算法效率分析

TPOM 算法分两个阶段进行离群挖掘,第一阶段对数据进行大致聚类;第二阶段在聚类结果上进行对象的近似最近邻查询,并采用改进的循环嵌套 KNN 算法进行挖掘离群。以 n 表示数据集 D 中对象的数目,d 表示数据的属性特征个数,则第一阶段的平均复杂度为 $O(n\log n \times d)$[11]。由于离群的数目远远少于数据的总数,根据 TPOM 算法的主要思想,对于占数据集大多数的正常数据,离群检测时,算法只需对数据所在的类别进行扫描,而不须扫描整个数据集。因此,第二阶段具有近似线性的时间复杂度 $O(n \times d)$。最坏的情况是对于数据集中的大部分数据,算法需要扫描所有数据才能确定对象的近似最近邻,此时,TPOM 算法的时间复杂度为 $O(n^2)$。综合离群检测的两个阶段,TPOM 算法具有近似线性的时间复杂度。

3 算法应用及分析

3.1 应用背景

根据福建省海洋环境质量监测数据质量控制的需要,TPOM 算法应用于监测数据的分析处理中,以检测出监测数据中错误的、不一致或离群的值,并进一步分析异常出现的原因,为海洋环境信息管理部门的环境决策提供有用的信息。

本文仅取 2006 年连续采集的“福建省海洋环境质量生态浮标监测”数据集为例,说明 TPOM 算法挖掘的过程及结果,并结合领域知识对挖掘结果进行分析。生态浮标监测数据集共有 14 497 条记录,属性特征及意义如表 1 所示。由于自然或人为因素的影响,部分监测数据缺失,离群检测过程中,TPOM 算法将缺失的数据统一处理为 0。

表 1 海洋环境质量生态浮标监测数据属性特征表

字段名	含义
Y_TEMP	水温
Y_SPCND	电导率
Y_SAL	表层盐度
Y_DOM	单位溶解氧
Y_PH	Ph 值
Y_TURB	浊度
Y_CHL	叶绿素

3.2 结果分析与讨论

福建省海洋环境质量生态浮标监测数据离群检测中,设置对象的最近邻数目为 100,挖掘的离群数目为 10,变形的聚类算法 VK-means 中类别的最大对象数为最近邻数目的 3 倍即 300,则 TPOM 算法挖掘的结果如表 2 所示。结合生态浮标监测数据,分析得知,TPOM 算法与传统 KNN 具有一样的挖掘结果,离群集中出现在 2006 年 7 月中下旬,正好是 2006 年 7 月 14 日第四号热带风暴“碧利斯”从福建登陆后,2006 年 7 月 20 日再次受到第 5 号台风“格美”的打击,从而使得福建、广东、广西、江西、浙江、湖南等地陆续出现洪涝灾害。据悉,第四号强热带风暴“碧利斯”含水量比较大,行进速度忽快忽慢,因而导致包括福建在内的南方六省出现了连续的强降雨天气,再加上此前南方许多地区已发生过多次强降雨天气,从而导致了该时间段内的海面生态浮标监测明显偏离正常的监测值。

表 2 TPOM 算法离群检测结果

离群序号	数据 ID	KNN 距离	生态浮标数据采集时间
1	714	564.32	06-7-20 14:00
2	721	558.83	06-7-20 17:30
3	718	557.11	06-7-20 16:00
4	1060	554.00	06-7-19 21:00
5	708	553.28	06-7-20 11:00
6	893	552.74	06-7-21 9:30
7	686	552.67	06-7-21 23:30
8	904	552.40	06-7-18 21:30
9	717	550.87	06-7-20 15:30
10	712	550.29	06-7-20 13:00

采用 TPOM 算法进行离群挖掘得到的结果与传统的循环嵌套 KNN 算法是一样的。两者所不同的是,传统的 KNN 算法需要精确地搜索每个数据的 k 个最近邻,效率为 $O(n^2)$;而 TPOM 算法采用了有效的剪枝策略,对于多数正常数据,通常只须对所在的类别进行搜索,通过对每个类别的对象数的控制,可以有效地控制数据近似 k 最近邻搜索的时间,只有少数离群点和边界点的 k 最近邻搜索需要遍历整个数据集,因此

具有近似线性的时间复杂度。本文 TPOM 算法和传统的 KNN 算法都采用 Java 实现,实验平台为 P4 1.8 GHz,512M 内存的 PC,TPOM 算法和 KNN 算法在离群点数目 $m = 10,20$; 近邻数 $k = 20,50,100$ 时的效率对比如图 1 所示。

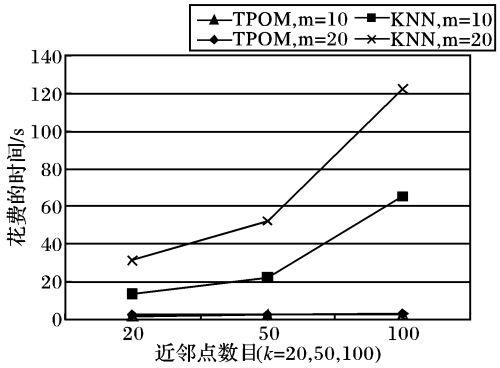

图 1　TPOM 和 KNN 的效率对比

实际应用研究发现,TPOM 算法在聚类的基础上,具有很好的可扩展性。一方面,算法可以充分利用先进的聚类技术,提高算法第一阶段的聚类效率;另一方面,采用距离的相似性度量方法,可以有效地度量高维数据之间的相似度,同时可以迅速地将数值型变量扩展到名称、顺序、比率以及混合型变量。正常情况下,TPOM 算法具有近似线性的时间复杂度,算法的时间复杂度随维数增多呈线性增长。但算法的效率有三个方面的影响因素,首先,数据集不应该是有序的,因为算法无法在有序数据集中实现近似最近邻的快速查询和正常点的有效剪枝;其次,数据必须是相对独立的,若数据相互依赖且具有大致一样的数值,则很可能处于同一个类别,近似最近邻查询需要扫描大部分数据才能达到目的;最后,算法不适于挖掘不包含离群的数据集,因为都是正常的数据无法实现正常点的有效剪枝。

4　结语

本文对基于距离的循环嵌套 KNN 算法进行了分析与改进,提出了二阶段近似最近邻离群挖掘算法,实现了分两个阶段完成的离群挖掘。第一阶段采用变形的 k-均值聚类方法对数据集进行大致聚类;第二阶段在聚类的基础上,采用基于循环嵌套的近似 k 最近邻算法来挖掘离群。由于聚类后,同一个类别中的数据的相似度较大,对于数据集中占大多数的正常数据,近似 k 最近邻查询只需对所在的类进行搜索而不须搜索整个数据集,因此,算法具有近似线性的时间复杂度。TPOM 算法应用在福建省海洋环境质量监测数据的分析与处理中,结果表明算法具有较好的适用性。进一步的工作主要包括相似性度量的改进、最近邻查询和剪枝效率的提高以及多源数据的离群检测。

参考文献:

[1] HODGE V, AUSTIN J. A survey of outlier detection methodologies [J]. Artificial Intelligence Review, 2004, 22(2): 85 - 126.

[2] 黄洪宇, 林甲祥, 陈崇成, 等. 离群数据挖掘综述[J]. 计算机应用研究, 2006, 23(8): 8 - 13.

[3] KNORR E, NG R. Algorithms for Mining Distance - Based Outliers in Large Datasets[A]// Proceedings of the 24th Int'l Conference on Very Large Databases. New York: ACM Press, 1998: 392 - 403.

[4] KNORR E M, NG R T. Finding intensional knowledge of distance-based outliers[C]// Proceedings of the 25th International Conference on Very Large Data Bases table of contents. San Francisco: Morgan Kaufmann Publishers Inc, 1999: 211 - 222.

[5] BARNETT V, LEWIS T. Outliers in Statistical Data[M]. 3rd ed. New York: John Wiley, 1994.

[6] KNORR E M, NG R T, TUCAKOV V. Distance-based outliers: algorithms and applications [J]. The VLDB Journal, 2000, 8(3): 237 - 253.

[7] BOLTON R J, HAND D J. Statistical fraud detection: A review[J]. Statistical Science, 2002, 17(3): 235 - 255.

[8] RAMASWAMY S, RASTOGI R, SHIM K. Efficient algorithms for mining outliers from large data sets[C]// Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data. New York: ACM Press, 2000: 427 - 438.

[9] KIM T-W, LI K-J. A distance - based packing method for high dimensional data[C]// Proceedings of the Fourteenth Australasian database conference on Database technologies. Adelaide, Australia: [s. n], 2003, 135 - 144.

[10] BAY S D, SCHWABACHER M. Mining Distance - Based Outliers in Near Linear Time with Randomization and a Simple[C]// Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. Washington: ACM Press, 2003: 29 - 38.

[11] JIANG M F, TSENG S S, SU C M. Two-phase clustering process for outliers detection[J]. Pattern Recognition Letters, 2001, 22(6 - 7): 691 - 700.

[12] HAN J W, KAMBER M. 数据挖掘概念与技术[M]. 范明, 孟小峰, 译. 北京: 机械工业出版社, 2001.

(上接第 2597 页)

5　结语

本文针对迁移实例服务和工作流服务的不同特点,设计了一种包括 MI Server 和工作流服务器的位置服务体系结构,其主要优点有:

1)对不同类型的服务采取了不同的处理方式,以达到功能和性能的最大优化。

2)多线程的工作流服务能同时处理多个 MI 的请求,提高了系统的并发度。

3)系统更加灵活,可以通过增加新线程实现工作流服务的动态扩展。

参考文献:

[1] CICHOCKI A, RUSINKIEWICZ M. Migrating Workflows [C]// Workflow Management Systems and Interoperability. Berlin, Heidelberg: Springer-Verlag, 1998: 339 - 355.

[2] 曾广周, 党妍. 基于移动计算范型的迁移工作流研究[J]. 计算机学报, 2003, 26(10): 1343 - 1349.

[3] 谢浩, 王晓琳, 曾广周. 面向服务的柔性迁移工作流停靠站设计[J]. 计算机应用, 2006, 12(3): 685 - 687.

[4] 纽约州立大学奥斯威戈分校网. Overview of package util.concurrent Release 1.3.4[EB/OL]. [2007 - 05 - 01]. http:// www.oswego.edu.

[5] 罗时飞. JBoss 管理与开发核心技术[M]. 3 版. 北京: 电子工业出版社, 2004.