(19)中华人民共和国国家知识产权局

(12)发明专利申请

(10)申请公布号 CN 107908624 A
(43)申请公布日 2018.04.13

(21)申请号 201711321280.6

(22)申请日 2017.12.12

(71)申请人 太原理工大学
地址 030024 山西省太原市万柏林区迎泽西大街79号

(72)发明人 谢珺 邹雪君 杨云云 续欣莹

(74)专利代理机构 太原市科瑞达专利代理有限公司 14101
代理人 卢茂春

(51)Int.Cl.
G06F 17/27(2006.01)
G06K 9/62(2006.01)

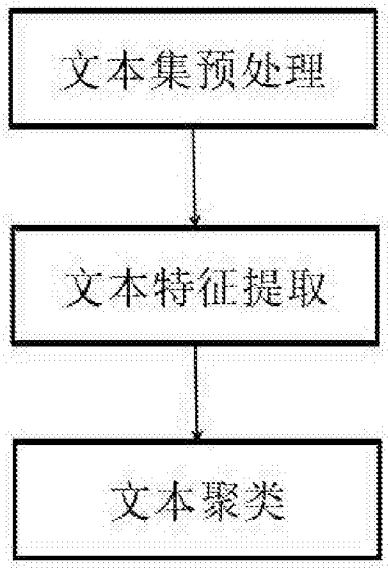
权利要求书2页 说明书5页 附图2页

(54)发明名称

一种基于全覆盖粒计算的K-medoids文本聚类方法

(57)摘要

一种基于全覆盖粒计算的K-medoids文本聚类方法，该方法包括以下步骤：1)对文本进行预处理，包括中文分词，去停用词；2)对文本进行特征提取，设置高频词与低频词阈值，滤除区分度不够的高频词和代表性不强的低频词，然后利用TF-IDF算法建立词向量空间模型；3)对文本进行聚类，首先利用single-pass对文本进行粗聚类，利用全覆盖粒计算理论的粒度重要性概念计算初始聚类中心候选集，然后基于密度和最大最小距离算法计算初始聚类中心，最后利用k-medoids算法进行文本聚类。本发明有效的解决了传统K-medoids聚类算法随机选取初始聚类中心，存在迭代次数增加、聚类结果波动较大的问题，也解决了当前改进K-medoids聚类算法中初始聚类中心位于同一类簇的问题。



CN 107908624 A

1.一种基于全覆盖粒计算的K-medoids文本聚类方法,其特征在于包括下述内容:

(1)对文本进行预处理,包括中文分词,去停用词;

(2)对文本进行特征提取,设置高频词与低频词阈值,滤除区分度不够的高频词和代表性不强的低频词,然后利用TF-IDF算法建立词向量空间模型;

(3)对文本进行聚类,首先利用Single-Pass对文本进行粗聚类,利用全覆盖粒计算理论的粒度重要性概念计算初始聚类中心候选集,然后基于密度算法和最大最小距离算法计算初始聚类中心,最后利用k-medoids算法进行文本聚类。

2.根据权利要求1所述的一种基于全覆盖粒计算的K-medoids文本聚类方法,其特征在于对文本进行特征提取中的滤除区分度不够的高频词和代表性不强的低频词,具体操作包括以下步骤:假设词j的频率为m,M_1为低频词频率,M_2为高频词频率,若$M_1<m<M_2$则保留该词,否则剔除,达到降维的目的。

3.根据权利要求1所述的一种基于全覆盖粒计算的K-medoids文本聚类方法,其特征在于对文本进行聚类中的single-pass粗聚类,包括以下步骤:

(1)从文档集n中输入第一篇文档d_1作为第一类中的中心,n为正整数;

(2)输入第二篇文档与第一篇文档做相似性处理,得到相似结果θ,若θ>σ,则第二篇分到第一类中并重新计算中心,否则第二篇作为新的一类;

(3)输入第i篇文档,分别与已有类别中的中心文档做相似性处理,得到与d_i相似度最大的类别m且记录相似结果θ,若θ>σ,则d_i分配到类别m中并重新计算中心,否则成为新的一类;

(4)重复第三步,直至最后一篇文档分配类别,即整个聚类过程结束。

4.根据权利要求1所述的一种基于全覆盖粒计算的K-medoids文本聚类方法,其特征在于对文本进行聚类中的全覆盖粒计算理论的粒度重要性概念,具体如下:

设$\boldsymbol{C}=\{C_i \mid C_i \in \boldsymbol{C}, C_i \subseteq U\}$是非空论域U上的一个全覆盖,全覆盖$P \subseteq \boldsymbol{U}$,P={$C_j$:j=1,…,n},定义粒$G_x$的中心、全覆盖粒C的中心、P的全覆盖粒度熵分别为:

$center_C(x)=\cap\{N_C(x) \mid x \in N_C(x), N_C(x) \in G_x\}$

$center(C)=\{center_C(x) \mid x \in U\}$

$$I(P)=\sum_{x\in U}\frac{1}{|U|}\left(1-\frac{|center_P(x)|}{|U|}\right)=1-\frac{1}{|U|^2}\sum_{x\in U}|center_P(x)|$$

$$Sig_{\boldsymbol{C}-C_i}(C_i)=I(\boldsymbol{C})-I(\boldsymbol{C}-C_i)$$

其中,$|center_P(x)|$表示$center_P(x)$的基数。

基于上述全覆盖粒计算模型的相关基础概念,定义全覆盖平均粒度重要性,设C={C_i:1,...,m}是非空论域U上的一个全覆盖,定义平均粒度重要性为:

$$\overline{Sig(C)}=\frac{1}{m}\sum_{i=1}^{m}Sig_{\boldsymbol{C}-C_i}(C_i)。$$

5.根据权利要求1所述的一种基于全覆盖粒计算的K-medoids文本聚类方法,其特征在于对文本进行聚类中的基于密度算法和最大最小距离算法,包括如下步骤:

(1)n个样本分为$C_1,C_2,..,C_p$共P类,P>K,计算每类的中心($z_1,z_2,...,z_P$)并选取$C_1,C_2,..,C_p$中包含样本数最多的类的中心作为第一个聚类中心v_1;

(2)选取距离第一个聚类中心v_1最远的中心作为第二个聚类中心v_2;

(3)计算其余中心与v_1、v_2之间的距离,并求出它们中的最小值,即:

$d_{ij}=||z_i-v_j||,j=1,2$

$d_i=\min(d_{i1},d_{i2}),i=1,2,...P$

(4)若$d_1=\max(d_i)$则相应的中心z_1作为第三个聚类中心v_3;

(5)依此类推若存在k个聚类中心,计算各中心到各个聚类中心的距离d_{ij},并算出:

$d_k=\max(\min(d_{i1},d_{i2},...,d_{i(k-1)})),i=1,2,...P$

z_k为第k个聚类中心。

6.根据权利要求1所述的一种基于全覆盖粒计算的K-medoids文本聚类方法,其特征在于对文本进行聚类中的k-medoids算法,包括如下步骤:

(1)从n个样本中随机选取K个样本作为初始聚类中心;

(2)分别计算剩余样本到K个初始聚类中心的距离,该剩余样本并入到距离最小的类簇中,所有剩余样本计算完毕后n个样本被分为K类;

(3)重新计算每类的聚类中心,计算每类中的样本中心,距离该中心最近的样本成为新的聚类中心;

(4)重复第(2)、(3)步骤,直到所有的聚类中心不变时算法结束,

其中,更新的聚类中心公式:$v_i=\{x_{ij}\,|\min_{j=1}^{K}|x_{ij}-\sum_{k=1}^{M}x_{ik}/M|\}$,$x_{i1},x_{i2},...x_{iM}$表示K类中第i类所包含的M个样本。

一种基于全覆盖粒计算的K-medoids文本聚类方法

技术领域

[0001] 本发明涉及全覆盖粒计算和文本挖掘技术，特别是涉及全覆盖粒计算的粒化以及文本聚类的方法。

背景技术

[0002] 互联网快速发展带来的信息过载、缺乏结构性等问题，使得人们很难在海量的信息中快速、准确地获取用户感兴趣的、潜在有用的内容，依靠手工对这些信息进行处理是不可能的。目前，绝大多数的网络信息都表现为文本形式，文本数据作为非结构化的数据，不像结构化的数据便于处理，因此大大降低了文本数据的利用率，而且大多数传统的信息检索技术不能处理海量的文本数据。数据挖掘是一门从大量有效数据中挖掘隐藏信息的有效技术，文本挖掘则是对文本信息进行数据挖掘的过程，随着文本数据的增长，文本挖掘成为数据挖掘领域中一个重要的研究方向，文本聚类是文本挖掘的预处理步骤，是文本进一步挖掘与分析的关键环节。文本聚类主要是对样本文档集计算相似性，并根据相似性程度将样本划分成若干类簇，同类簇的文档间相似性较大，不同内簇间的文档相似性则较小。

[0003] 文本聚类一直是国内外研究学者关注的热点、难点问题，研究已经取得巨大的成果，但是还是存在一系列亟待解决的关键问题，如样本的词向量空间维度过大，聚类中心的随机选取问题和计算复杂度大等。如何对数据降维，提高聚类质量，降低计算复杂度等都需要我们做进一步的研究。

发明内容

[0004] 本发明为了解决传统聚类方法随机选取聚类中心和文本聚类方法准确率较低的问题，提供一种基于全覆盖粒计算的K-medoids文本聚类方法，该方法包括以下步骤：

[0005] 1.对文本进行预处理，包括中文分词，去停用词；

[0006] 2.对文本进行特征提取，设置高频词与低频词阈值，滤除区分度不够的高频词和代表性不强的低频词，然后利用TF-IDF算法建立词向量空间模型；

[0007] 3.利用SinglePass算法对文档聚类，得到粗聚类集$C_1,C_2,..C_p$，构成全覆盖计算$C=\{C_i:i=1,\cdots,p\}$，按照全覆盖粒计算的相关定义分别计算粒度重要性和平均粒度重要性，选择$Sig_{c-C_i}(C_i)\geq\overline{Sig_{c-C_i}(C_i)}$的粒子放入到集合S中。不妨假设S中含有N个粒子(N＜p)，若N≥K则进行第4步，若N＜K则返回第3步，在SinglePass中调整阈值δ直至N≥K，然后进行第4步；

[0008] 4.按照公式(1)计算S中每个粒子的中心，任意两个中心粒子间的欧式距离记为矩阵D；

[0009] 5.选择包含更多粒子对应的中心作为第一个聚类中心v_1，选择距v_1最远的粒子中对应的中心作为第二个聚类中心v_2；对于S中剩余粒子，根据矩阵D分别求出其中心到v_1,v_2距离为d_{i1},d_{i2}，取$d_i=\min(d_{i1},d_{i2})$，$d=\max(d_i)$对应的粒子中心为v_i，依此类推计算v_k，此时找到K个初始聚类中心

[0010] 6.对于任意$x_i\in U$，首先寻找与其最近的类心v_m(m＝1,2,...k)，此时样本分为K类；

[0011] 7.选每个类簇中与该簇其他对象距离之和最小的对象作为新的聚类中心，在K类中用新中心代替原始中心；

[0012] 8.重新分配每个对象到距离最近的中心点，获得聚类结果；

[0013] 9.计算所有对象到其类簇中心的距离之和，如果该值不变或者达到最大迭代次数则算法结束，否则转到第8步。

[0014] 所述的文本特征提取，具体包括以下操作：首先滤除区分度不够的高频词和代表性不强的低频词，即假设词j的频率为m，M_1为低频词频率，M_2为高频词频率，若$M_1<m<M_2$则保留该词，否则剔除，达到降维的目的。

[0015] 所述的TF-IDF算法，具体包括以下操作：

[0016] $$w_{i,j}=\frac{tf_{ij}\cdot idf_{i,j}}{\sqrt{\sum_{j=1}^{N}(tf_{ij})^2\cdot(idf_{i,j})^2}}$$

[0017] $$tf=\frac{x_{ij}}{|x_i|}$$

[0018] $$idf=\log(\frac{n}{|x_j|}+0.01)$$

[0019] x_{ij}表示第i篇文档中词j出现的频率，$|x_i|$表示该篇文档中所有词的词频总数，n表示样本总数，$|x_j|$表示词j包含的样本总数，N表示文档所有词的数量。

[0020] 所述的single-pass聚类，具体包括以下操作：

[0021] 1)从文档集n中输入第一篇文档d_1作为第一类中的中心；

[0022] 2)输入第二篇文档与第一篇文档做相似性处理，得到相似结果θ，若θ＞σ，则第二篇分到第一类中并重新计算中心，否则第二篇作为新的一类；

[0023] 3)输入第i篇文档，分别与已有类别中的中心文档做相似性处理，得到与d_i相似度最大的类别m且记录相似结果θ，若θ＞σ，则d_i分配到类别m中并重新计算中心，否则成为新的一类；

[0024] 4)重复第三步，直至最后一篇文档分配类别，即整个聚类过程结束。

[0025] 所述的全覆盖粒计算理论的粒度重要性概念，具体包括以下操作：

[0026] 设$\boldsymbol{C}=\{C_i|C_i\in\boldsymbol{C},C_i\subseteq U\}$是非空论域U上的一个全覆盖，全覆盖$\boldsymbol{P}\subseteq\boldsymbol{U}$，P＝{$C_j$:j＝1,…,n}，定义粒$G_x$的中心、全覆盖粒C的中心、P的全覆盖粒度熵分别为：

[0027] $center_C(x)=\cap\{N_C(x)|x\in N_C(x),N_C(x)\in G_x\}$

[0028] $center(C)=\{center_C(x)|x\in U\}$

[0029] $$I(P)=\sum_{x\in U}\frac{1}{|U|}(1-\frac{|center_P(x)|}{|U|})=1-\frac{1}{|U|^2}\sum_{x\in U}|center_P(x)|$$

[0030] $$Sig_{\boldsymbol{C}-C_i}(C_i)=I(\boldsymbol{C})-I(\boldsymbol{C}-C_i)$$

[0031] 其中，$|center_P(x)|$表示$center_P(x)$的基数。

[0032] 基于上述全覆盖粒计算模型的相关基础概念，定义全覆盖平均粒度重要性，设C＝{C_i:1,...,m}是非空论域U上的一个全覆盖，定义平均粒度重要性为：

[0033] $$\overline{Sig(C)}=\frac{1}{m}\sum_{i=1}^{m}Sig_{\boldsymbol{C}-C_i}(C_i)$$

[0034] 全覆盖粒计算是信息处理的一种新概念和计算范式，主要通过建立合适的粒度来

寻找解决问题的有效方法，降低问题的求解难度。全覆盖粒计算的基本问题归纳为两个方面，即粒化和粒的计算。粒化是求解空间的一个构造性过程，处理粒度的形成、粗细、表示和语义解释，粒的计算主要是指如何有效的利用粒度去解决复杂问题。

[0035] 本发明引入全覆盖粒计算模型，对文档集进行合理的粒化，利用粒的计算解决文本聚类问题。

[0036] 具体的文档粒化对应关系如表1所示：

文档集	全覆盖粒计算模型
文档集N	论域U
文档d	元素x
粗聚类集D	全覆盖C
每类的文档集D_i	基本粒C_i
粗聚类类数p	基本粒个数m

[0037]

[0038] 所述的密度算法和最大最小距离算法，具体包括以下操作：

[0039] 1).n个样本分为$C_1,C_2,..,C_p$共P类(P>K)，计算每类的中心$(z_1,z_2,...,z_P)$并选取$C_1,C_2,..,C_p$中包含样本数最多的类的中心作为第一个聚类中心v_1；

[0040] 2).选取距离第一个聚类中心v_1最远的中心作为第二个聚类中心v_2；

[0041] 3).计算其余中心与v_1、v_2之间的距离，并求出它们中的最小值，即：

[0042] $d_{ij}=||z_i-v_j||$,j=1,2

[0043] $d_i=\min(d_{i1},d_{i2})$,i=1,2,...P

[0044] 4).若$d_1=\max(d_i)$则相应的中心z_1作为第三个聚类中心v_3；

[0045] 5).依此类推若存在k个聚类中心，计算各中心到各个聚类中心的距离d_{ij}，并算出：

[0046] $d_k=\max(\min(d_{i1},d_{i2},...,d_{i(k-1)}))$,i=1,2,...P

[0047] z_k为第k个聚类中心；

[0048] 所述的k-medoids算法，具体包括以下操作：

[0049] 1)从n个样本中随机选取K个样本作为初始聚类中心；

[0050] 2)针对剩余的每一个样本，分别计算该样本到K个初始聚类中心的距离，该样本并入到距离最小的类簇中，所有样本计算完毕后n个样本被分为K类；

[0051] 3)重新计算每类的聚类中心，计算每类中的样本中心，距离该中心最近的样本成为新的聚类中心；

[0052] 4)反复重复上述第2)、3)步骤，直到所有的聚类中心不变时算法结束。其中，更新的聚类中心公示：$v_i=\{x_{ij}|\min_{j=1}^{K}|x_{ij}-\sum_{k=1}^{M}x_{ik}/M|\}$，$x_{i1},x_{i2},...x_{iM}$表示K类中第i类所包含的M个样本。

[0053] 本发明基于全覆盖粒计算的K-medoids文本聚类方法，通过Single-Pass方法以及全覆盖粒计算的相关理论，找到有效的初始聚类中心，降低聚类方法的复杂度，提高聚类方法的准确率。

附图说明

[0054] 图1为本发明整体示意图；

[0055] 图2为本发明中基于全覆盖粒计算的K-medoids文本聚类方法的流程图。

具体实施方式

[0056] 为更进一步阐述本发明为实现预定发明目的所采取的技术手段及功效，以下结合附图及较佳实施例，对依据本发明的具体实施方式、特征及其功效，详细说明如后。

[0057] 如图1所示，本发明的整体流程详述如下：

[0058] 步骤1：使用jieba分词对中文文本分词，对“哈工大停用词词库”、“四川大学机器学习智能实验室停用词库”、百度停用词表“等各种停用词表整理去重后提取新的中文词词表。

[0059] 步骤2：对步骤1去停用词后的分词结果进行TF-IDF特征提取。TF-IDF是一种统计加权方法，公式为

[0060] $$w_{i,j}=\frac{tf_{ij}\cdot idf_{i,j}}{\sqrt{\sum_{j=1}^{N}(tf_{ij})^2\cdot(idf_{i,j})^2}}$$

[0061] $$tf=\frac{x_{ij}}{|x_i|}$$

[0062] $$idf=\log(\frac{n}{|x_j|}+0.01)$$

[0063] x_{ij}表示第i篇文档中词j出现的频率，$|x_i|$表示文档i中所有词的词频总数，n表示样本总数，$|x_j|$表示词j包含的样本总数，N表示文档所有词的数量。

[0064] 这样就得到由样本的所有特征词组成的“样本—特征”矩阵。

[0065] 步骤3：对步骤2的“样本—特征”矩阵进行聚类，首先利用SIngles-Pass粗聚类，接着利用全覆盖粒计算理论的粒度重要性概念计算初始聚类中心候选集，然后基于密度和最大最小距离算法计算初始聚类中心，最后利用k-medoids算法进行文本聚类。

[0066] 步骤4：通过步骤3得到所有的聚类结果，利用聚类精度检测聚类效果。使用查全率(Recall)、查准率(Precision)以及值三个指标来衡量聚类的效果，具体公式

[0067] 如下：

[0068] $$P_i=\frac{|A_i\cap B_i|}{|A_i|}\times 100\%$$

[0069] $$R_i=\frac{|A_i\cap B_i|}{|B_i|}\times 100\%$$

[0070] $$F_i=\frac{2\times P_i\times R_i}{P_i+R_i}\times 100\%$$

[0071] $|A_i\cap B_i|$表示聚类类别A_i中包含对应人工类别B_i的文本个数，$|A_i|$表示聚类类别A_i包含的样本个数，$|B_i|$表示人工类别B_i包含的样本个数。

[0072] 在该实施例中，利用本发明的方法分别对1400篇复旦语料库。语料库的具体分布和统计结果如下表2和表3：

[0073] 表2:样本类别信息

数据集	标签/编号	样本数
Data Set1	艺术/1	198
	历史/2	200
	计算机/3	200
	太空/4	200
	环境/5	202
	经济/6	200
	体育/7	200

[0074]

[0075] 表3:样本统计信息

数据集	类别数	文本数	分词结果	最小样本数	最大样本数
Data Set1	7	1400	172324	198	202

[0076]

[0077] 表2中分词结果经过简单的降维后得到特征词集,样本集的“文档—特征”矩阵分别为1400×172324。

[0078] 表3实验对比结果

编号	K-medoids 算法			本文算法		
	P	R	F	P	R	F
1	59.78%	54.04%	56.76%	74.86%	67.68%	71.09%
2	57.80%	50.00%	53.62%	75.14%	65.00%	69.71%
3	57.55%	61.00%	59.22%	70.28%	74.50%	72.33%
4	63.05%	64.00%	63.52%	70.44%	71.50%	70.97%
5	71.43%	59.41%	64.86%	77.98%	64.85%	70.81%
6	54.42%	58.50%	56.39%	67.91%	73.00%	70.36%
7	53.75%	64.50%	58.64%	64.40%	80.50%	71.56%

[0079]

[0080] 根据表3的实验对比结果,本文算法的准确率、召回率与F值均高于K-medoids算法,表明聚类结果受初始聚类中心选取的影响,且K-medoids算法的正确率波动范围较大,易陷入局部最优。本文算法先是采用Single-Pass算法对文本集粗聚类,相关的文本集分别聚成簇,根据初始聚类中心一定在形成的大簇中的原则,利用全覆盖粒度重要性和平均粒度重要性选出初始聚类中心,也克服了初始聚类中心容易位于同一类簇的缺陷,取得较好的聚类结果。

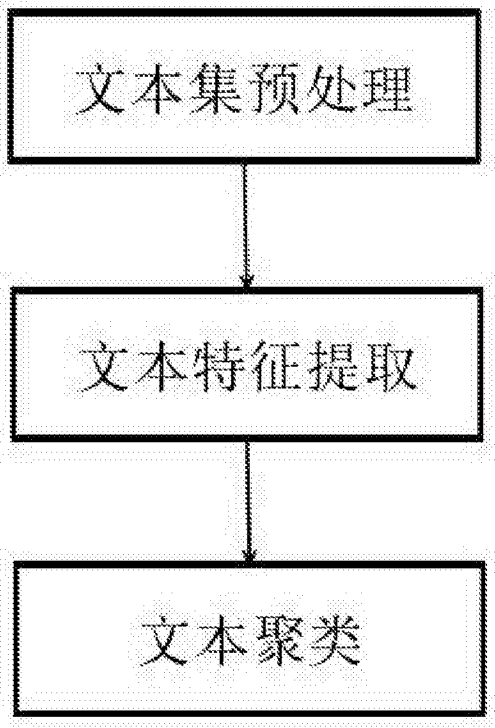


图1

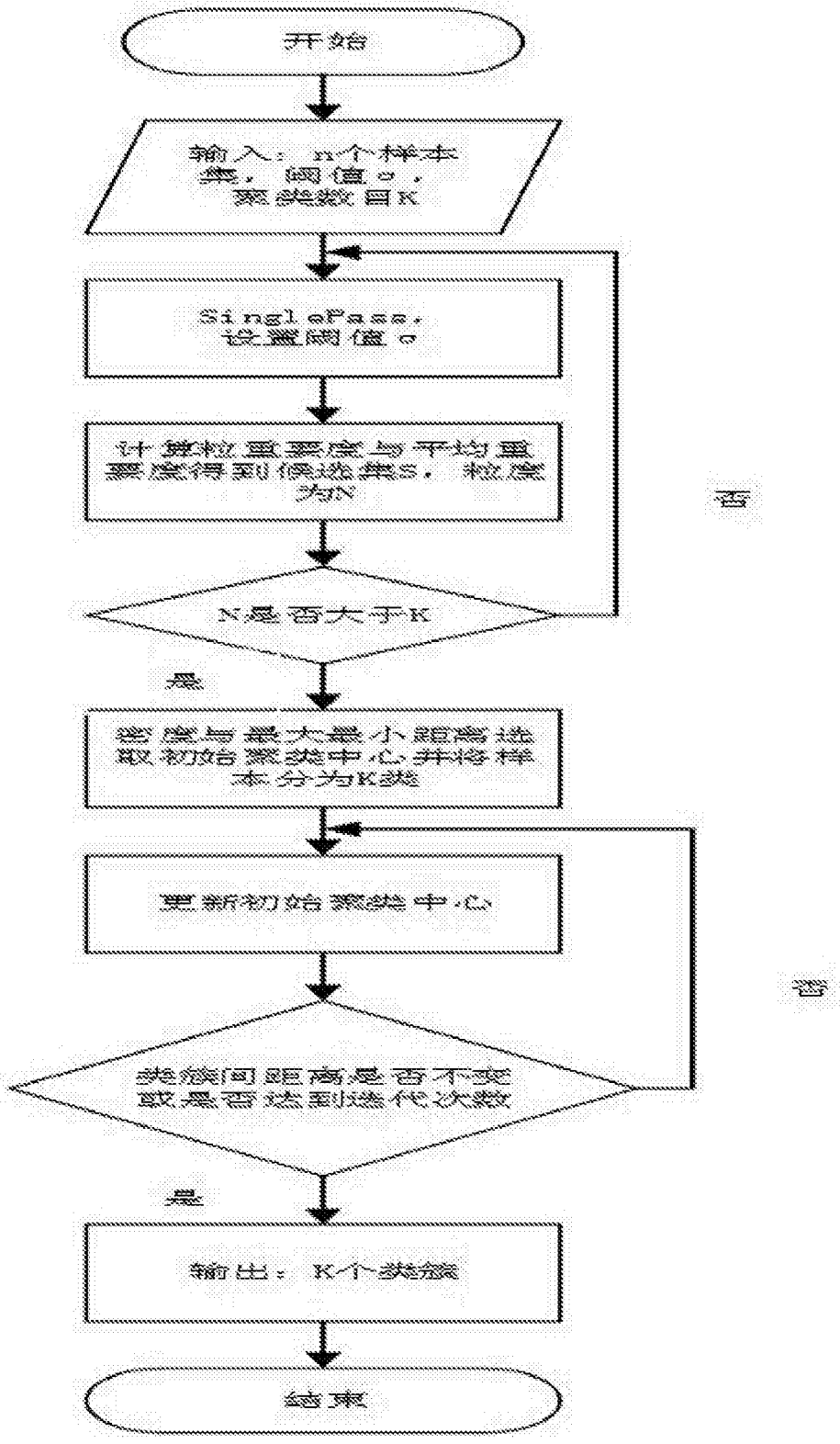

开始
输入：n个样本集，阈值σ，聚类数目K
Singlepass，设置阈值σ
计算粒重要度与平均重要度得到候选集S，粒度为N
否
N是否大于K
是
密度与最大最小距离选取初始聚类中心并将样本分为K类
更新初始聚类中心
否
类簇间距离是否不变或是否达到迭代次数
是
输出：K个类簇
结束

图2