基于量子遗传算法的 XML 聚类集成*

蒋 勇[1]，谭怀亮[2]，王祖析[1]，张朝霞[1]
(1. 湖南化工职业技术学院 信息系，湖南 株洲 412004；2. 湖南大学 计算机与通信学院，长沙 410082)

摘 要：为了改善单一聚类算法的聚类性能，提出一种基于量子遗传算法的 XML 文档聚类集成解决方法。该方法首先利用 KNN 分类算法将 XML 文档划分成 k 个差异性的聚类成员；其次根据聚类成员的关系获得内联相似度矩阵，并通过多次分割、向下、向上、双向收缩的 QR 算法分解特征值对应的特征向量来实现矩阵的维数缩减；然后在映射空间上用量子遗传算法实现聚类集成，把每一个样本判别到最优的聚类类别中。这样减少了数据差异性对聚类结果的影响，提高了聚类质量。实验结果表明，在真实的数据集上，该聚类集成算法比其他聚类集成算法具有更好的效果。

关键词：XML 文档；KNN 分类；量子遗传算法；聚类集成；聚类质量
中图分类号：TP301.6 **文献标志码**：A **文章编号**：1001-3695(2012)06-2200-05
doi:10.3969/j.issn.1001-3695.2012.06.052

XML clustering ensemble based on quantum genetic algorithm

JIANG Yong[1], TAN Huai-liang[2], WANG Zu-xi[1], ZHANG Zhao-xia[1]
(1. *Dept. of Information, College of Hunan Chemical, Zhuzhou Hunan* 412004, *China*; 2. *School of Computer & Communication, Hunan University, Changsha* 410082, *China*)

Abstract: To improve the clustering performance of a single clustering algorithm, this paper proposed an approach of the XML document clustering ensemble algorithm based on quantum genetic algorithm. Firstly, it divided the XML document into the difference of the k-members clustering using the K-nearest neighbour classifier algorithm. Next according to the relationship between the clustering members of the datasets was obtained Co-occurrence similarity matrix, and through a multi-segment and upward and downward double-direction shrink QR algorithm decomposition a large-scale matrix of eigenvalue to achieve the corresponding eigenvector matrix of dimensionality reduction. Finally in mapping space, using the quantum genetic algorithm to complete clustering ensemble, and discriminate the optimal clustering category from each sample. For to do it that would be reduced the data differences on the impact of clustering effects, and improved the clustering quality. Experiments on real-world data sets indicate that it has better clustering effects than clustering ensemble algorithms.

Key words: XML document; K-nearest neighbor partitioning; quantum genetic algorithm; cluster ensemble; clustering quality

0 引言

XML 聚类集成[1,2]是首先对 XML 原始数据进行聚类，然后对这些结果加以组合，最终获得比原始数据的聚类更好的结果，以方便查询和检索。若对 XML 采用聚类集成算法聚类，它比单聚类算法如 K-均值聚类算法、谱聚类算法[3,4]等有更高的聚类精度和稳定性，而且在进行初始聚类时可以使用不同的聚类算法分布式地处理数据，解决不可能先集中起来的数据聚类问题，同时也使噪声、孤立点对结果的影响较小。近年来的几百种聚类算法已经证明，K-均值聚类算法是一种最简单、使用最普遍的基于中心的单聚类算法，在紧凑的超球形分布的数据集合上有很好的性能，但对 XML 这种半结构化的文档没有优势；谱聚类算法虽然克服 K-均值聚类算法的缺点，具有识别非凸分布聚类的能力，算法具有全局最优解，但它的时间复杂度达到 $O(n^3)$，当 XML 文档数据集的规模较大时求解特征值因数据量太大要耗费 CPU 太多的资源，所以采用此方法求解大规模的 XML 文档数据聚类也不妥。

若采用如下几类聚类集成算法对 XML 文档进行聚类也存在一些缺陷：a)采用基于图形分割算法[5,6]，这类算法是先把初始聚类结果转换成图的顶点和边，或者超图的顶点和超边，然后使用图划分算法进行分割，所以结果会受到图划分算法的影响；b)采用基于特征方法的自适应聚类集成选择算法[7]，这类算法是根据不同的特征进行初始聚类，然后采用多样性测度方法进行聚类集成，这样对 XML 文档来说，就会存在把具有不同特征的同一元素划分到不同类别的问题出现；c)基于概率统计的半监督聚类集成算法[8]，这类算法虽然克服 b)类算法的缺陷，元素聚类也服从概率统计结果，但同样存在划分不准

收稿日期：2011-11-01；**修回日期**：2011-12-12 **基金项目**：国家教育部博士点基金资助项目(200805321029)；湖南省自然科学基金资助项目(07JJ6139)

作者简介：蒋勇(1967-)，男，湖南邵阳人，副教授，硕士，主要研究方向为图形图像、人工智能、XML 数据库(hunanlaojiang@163.com)；谭怀亮(1970-)，男，湖南双峰人，副教授，硕导，博士，主要研究方向为嵌入式系统、工业过程先进控制、网格计算、图形图像；王祖析(1965-)，男，湖南双峰人，高级讲师，主要研究方向为计算机网络网格、XML 数据库、计算机实验；张朝霞(1984-)，女，湖南永州人，硕士，主要研究方向为数字通信、人工智能、XML 数据库.

确的问题;d)基于内在关联的层次聚类算法[9],这类算法虽然用于聚类时考虑元素的结构信息,但没有考虑元素在不同层的信息,因此也不太适合 XML 聚类集成。

综上所述,对 XML 文档来说,由于文档中有许多不同的 XML 元素及其属性,它们的差异性很大,即使同一个元素在不同节点层其含义也不一样。所以对 XML 文档进行聚类,要综合考虑结构信息和元素信息这些因素,本文的基于量子遗传算法[10,11]的 XML 文档的聚类集成方法就是综合考虑了这些因素。其方法是先把 XML 文档集看成图的顶点集合,其中每个 XML 文档是图的一个顶点,图的边是顶点之间的连线,它表示文档之间的相似度,其相似度采用内容与结构相结合的方法计算,按照它们组成的边的权重最小、一条路径的加权之和最大的原则把图划分成不同的路径,路径的划分采用 K-邻近法[12],这样把划分的每一条路径组成一个聚类,所有不同的划分路径组成 K 个初始聚类;其次从初始聚类结果出发构造映射关系,获得内联相似度矩阵,对内联相似度矩阵使用多次分割、向下、向上双向收缩的 QR 算法[13]求解其特征值对应的特征向量,把获得的特征向量进行空间映射来实现降维;最后用量子遗传算法来实现最终的聚类集成。

1　XML 文档聚类问题简介

1.1　内容与结构相结合的XML文档相似度

设有 n 个 XML 文档集合 $D=(d_1,d_2,\cdots,d_n)$,若 d_i、d_j 为两个 XML 文档,则它们的相似度定义为

$$\begin{aligned}\mathrm{dist}(d_i,d_j)=&(\mathrm{contSim}(d_i,d_j)\times\lambda)+\\&(\mathrm{structSim}(d_i,d_j)\times(1-\lambda)\end{aligned}\tag{1}$$

其中:$\mathrm{contSim}(d_i,d_j)$ 表示 XML 文档内容相似度;$\mathrm{strucSim}(d_i,d_j)$ 是结构相似度;λ 是[0,1]之间的调节结构相似度与内容相似度的阈值参数。为了求出每对 XML 文档相似度,首先要对文档集进行预处理,求出每个文档的关键词特征向量和所有不同的路径向量。在这里把属性值也看成是关键词。若一个 XML 文档 d_i 的关键词特征向量表示为 $d_i=(w(w_1,d_i),w(w_2,d_i),\cdots,w(w_m,d_i))$,则 $w(w_k,d_i)$ 为关键词 w_k 在文档 d_i 出现的带权频率,且

$$w(w_k,d_i)=tf(w_k,d_i)\times\ln(N/N_k)$$

其中:$tf(w_k,d_i)=\sum_{i=1}^{m}((w_k^j,d_i)\times(1/\mathrm{level}(w_k^j,d_i)))$ 为关键词 w_k 在 d_i 中的频率,(w_k^j,d_i) 为关键词 w_k 在文档 d_i 的第 j 层出现的次数,$\mathrm{level}(w_k^j,d_i)$ 为关键词 w_k 在文档 d_i 所处的路径序列中第 j 层的层深;N 为 XML 文档集的关键词数目;N_k 为关键词 w_k 在所有文档集出现的数目。若 XML 文档集 D 中的路径集合为 $P=(p_1,p_2,\cdots,p_f)$,则文档 d_i、d_j 的路径分别为 $(p_{i,1},p_{i,2},\cdots,p_{i,f})$ 和 $(p_{j,1},p_{j,2},\cdots,p_{j,f})$,其中 d_i 中的一条路径 $p_{i,1}$ 是指从根元素到叶子元素的节点序列。文档的结构距离用欧氏距离来表示其相似度:$\mathrm{structSim}(d_i,d_j)=\|p_i-p_j\|_2$,它正则化为[0,1],内容相似度采用 cos 余弦来表示:

$$\mathrm{contSim}(d_i,d_j)=\frac{d_i\cdot d_j}{\|d_i\|_2\times\|d_j\|_2}$$

其中:d_i、d_j 为两文档对应的关键词特征向量。

1.2　k-临近划分及聚类

对有 n 个 XML 文档集合 $D=(d_1,d_2,\cdots,d_n)$,把每个文档看成一个图的顶点,这些顶点组成一个无向图 $G=(V,E)$。其中:$V=(d_1,d_2,\cdots,d_n)$ 是顶点的集合,E 是连接顶点的边集合。若两顶点有边相连接,则表示该两顶点是邻接的。为了求取图中的 k 条路径及聚类,使用 k-临近方法来划分其路径并构造其初始聚类。其策略是:

a)从图中选取 k 个差异性大的顶点中的一个作为路径的起始点,计算它到 k 个邻近点之间的相似度,相似度按式(1)计算。根据相似度的大小,把所有相似度大的顶点组成一个集合 A,相似度小的顶点组成一个集合 B,再执行如下步骤:

(a)从集合 A 中选取最邻近的且相似度大的点,其相似度记为 dist_{ij}。

(b)计算经过顶点 v_i 到达顶点 v_j 的所有路径相似度集合,从中选取最大最小相似度,即

$$\mathrm{dist}'_{ij}=\max_{p\in p_{i,j}}(\mathrm{dist}''_{i,j})=\max_{p\in p_{i,j}}\min_{1\leqslant l<|p|}(\mathrm{dist}'_{p[l]p[l+1]})$$

这里 $p\in p_{ij}$,$p[l]$ 表示从顶点 i 到 j 的路径中第 l 层顶点。

(c)若 $\mathrm{dist}_{ij}>\mathrm{dist}'_{ij}$,则选用经过顶点 v_i 到达顶点 v_j 的路径的相似度来代替该对顶点的相似度;否则该对顶点的相似度就是该条路径上连接该对顶点可达的路径相似度。

(d)判断该连接的顶点对是否属于同一路径(设定一个阈值 θ 作为判断条件)划分中,若是则执行(e);否则须从集合 A 中找出下一对相似度次最近的顶点。重复执行(a)~(d),直至找到要连接的顶点对属于同一路径为止;若没有找到,则须重新选择路径的起始点。

(e)把属于同一路径的顶点组合起来构造成为初始聚类的样本点,同时从访问过的路径邻接点出发,广度优先搜索其 k 个邻近点,求取满足条件的属于同一路径的顶点,并把它们归于初始聚类为止。

b)对集合 B 中相似度小的顶点进行补充。这些相似度小的顶点,有些是两类的边缘点,有些是孤立点(噪声点)。若是两类的边缘点,则比较其到两类之间距离的远近,根据距离,将其划分到离更近的那一类中;若是孤立点,可以将其剔除掉而不影响其聚类,或者单独看成一类。

c)计算划分类别的聚类中心。

d)运行步骤 a)~c)k 次,求得 k 个初始聚类为止。

2　量子遗传算法的聚类集成

量子遗传算法的 XML 文档聚类集成分成两步:a)生成阶段,把 XML 数据集作为输入,通过运行 KNN 分类算法 k 次,得到 k 个基聚类和 k 个聚类中心;b)聚类组合阶段,把聚类成员作为输入,通过量子遗传聚类算法寻找该样本集的最优聚类组合,把每一个样本判别到最优的聚类类别中,从而完成聚类集成并输出最终的结果。

2.1　基于内联相似度矩阵及分解

通过对原始 XML 文档进行 KNN 分类算法 k 次后,得到 k 个长度为 n 的基聚类向量 $\Pi=\{\pi^1,\pi^2,\cdots,\pi^k\}$ 和聚类中心 $V=\{v_1,v_2,\cdots,v_k\}$。其中:π^i 是由 KNN 分类算法产生的在文本集 D 上的一个划分,$\pi^i=\{d_1^i,d_2^i,\cdots,d_k^i\}$,$\cup_k d_k^i=D$,$d_i\rightarrow\{\pi^1(d_i),\pi^2(d_i),\cdots,\pi^k(d_i)\}$ 表示第 i 个样本对应的所有聚类成员划分结果,$\pi^i(d_i)$ 是第 i 个样本在聚类成员 π^i 中的标签。

构造一个矩阵 W,则该矩阵由 k 维向量组成,每个向量有 n 个元素,且每个元素是其所在的聚类成员中的类标签,不再是文档样本在原始特征空间下的属性值,这个矩阵就是内联相似度矩阵,如表 1 所示。

表 1 数据集新特征空间矩阵

	π^1	π^2	π^3	…	π^k
d_1	$\pi^1(d_1)$	$\pi^2(d_1)$	$\pi^3(d_1)$	…	$\pi^k(d_1)$
d_2	$\pi^1(d_2)$	$\pi^2(d_2)$	$\pi^3(d_2)$	…	$\pi^k(d_2)$
d_3	$\pi^1(d_3)$	$\pi^2(d_3)$	$\pi^3(d_3)$	…	$\pi^k(d_3)$
…	…	…	…	…	…
d_n	$\pi^1(d_n)$	$\pi^2(d_n)$	$\pi^3(d_n)$	…	$\pi^k(d_n)$

若对该矩阵采用矩阵分解的方法来获得其特征值和其对应的特征向量,它的时间复杂度达到 $O(n^3)$,此方法不适合大矩阵的特征值求解。为了获得好的求解目的,本文采用多次分割、向下、向上双向收缩的 QR 算法来很好地解决求特征值的问题。其基本思想是:首先对 W 矩阵进行 Householdler 变换,将其化为上双对角阵 A;然后对矩阵 A 分割为一些低阶上双对角子方阵,即 $A=\begin{bmatrix} A_1 & & & 0 \\ & 0 & \ddots & \\ & & & A_i \end{bmatrix}$,其中每个子方阵具有形式 $A_i=\begin{bmatrix} \times & \times & & 0 \\ & \ddots & \ddots & \\ & & \ddots & \ddots \\ 0 & & \times & \times \\ & & & \times \end{bmatrix}$,且双对角带 × 的元素均不为 0,依次对这些子方阵进行一次 QR 迭代,在迭代过程中,使用 Givens矩阵与子方阵左右相乘,直接驱逐出子方阵产生新的非 0 元素及不满足条件的次对角线的元素;在完成对所有子方阵的一次 QR 迭代后,就对矩阵 A 向下和向上收缩多行,重复对收缩后的矩阵 A 进行分割、子方阵 QR 迭代、向下和向上收缩,就能求出其特征值。其算法描述如下:

a)初始化。对 W 矩阵,进行 Householdler 变换,将其变换为上双对角阵 A,并设它的首行 main_pre = 1,末行 main_rea = k。其中 $k=\min(m,n)$ 为上双对角阵 A 的阶数。每个子方阵 A_i 的首行为 pre,末行为 rea。

b)对 A 矩阵进行分割,即采用如下方法求得子方阵的首行 pre 和末行 rea:当主对角元为 0、次对角元不为 0 时,利用一系列 Givens 左变换将次对角元从左至右驱逐出矩阵,当主对角元不为 0、次对角元不为 0 时,得到所求的子方阵首行;同理,当主对角元为 0、次对角元为 0 或不为 0 时,利用一系列 Givens 右变换将次对角元从下至上驱逐出矩阵,当主对角元不为 0、次对角元为 0 时,得到所求的子方阵末行。

c)对子方阵进行一次 QR 迭代。

d)判断 rea 是否小于 main_rea,若小于则返回 b),继续分割下一个子方阵并进行 QR 迭代;否则对矩阵 A 进行双向收缩。

e)对矩阵 A 进行双向收缩,判断首行与末行是否相遇,若不相遇,则执行步骤(a)~(f);否则执行步骤 f)。

(a)i = main_pre,j = main_rea;

(b)如果 $a_{i,i+1}$ 次对角元素等于 0,矩阵向下收缩一行,$i=i+1$,main_pre = i;

(c)如果 $a_{i,i+1}$ 次对角元素不等于 0、$a_{i,i}$ 主对角元等于 0,则利用 Givens 左变换将 $a_{i,i+1}$ 从左至右驱出矩阵 A,然后矩阵向下收缩一行,$i=i+1$,main_pre = i;

(d)如果 $a_{i,i}$ 主对角元不等于 0 、$a_{i,i+1}$ 次对角元素不等于 0,则收缩到此行为止,跳出循环转步骤 f);

(e)如果右下角次对角元 $a_{j-1,j}$ 等于 0,则 A 向上收缩一行,$j=j-1$,main_rea = j;

(f)如果右下角次对角元 $a_{j-1,j}$ 不等于 0,则收缩到此行为止,跳出循环转步骤 f)。

f)如果 main_rea 大于 main_pre,则返回步骤 b),对收缩后的矩阵进行下一轮分割。

g)把分离的特征值按照从大到小排序,计算它的前 k 个特征值对应的特征向量 $x_1,x_2,\cdots,x_k$,构造矩阵 $X=[x_1,x_2,\cdots,x_k]\in R^{n\times k}$,规范化 X 为矩阵 $Y=X_{ij}/(\sum_j X_{ij})^{1/2}$。

2.2 量子遗传聚类集成及算法

2.2.1 量子遗传算法与量子染色体编码

量子遗传算法是量子计算与遗传算法结合的产物。它建立在量子态矢量表示的基础上,用量子比特编码来表示染色体,以量子旋转门实现染色体基因的调整。在量子遗传算法中,一个染色体可以表达多个态的叠加,一个量子比特可能处于|0>、|1>,或两者之间的中间态,即一个量子比特可以为一个“0”态或“1”态,或它们的任意叠加态。因而它们可表示为 $|\psi\geqslant\alpha|0>+\beta|1>$。其中 α、β 是两个复数,满足 $|\alpha^2+\beta^2|=1$,$|\alpha^2|$和 $|\beta^2|$分别是|0>和|1>状态的概率,其量子比特用概率幅表示为$[\alpha,\beta]$。若有 n 个体的量子种群 $p(t)=\{p_1^t,p_2^t,\cdots,p_n^t\}$,则 p_j^t 为第 t 代种群中的第 j 个量子染色体。

而在本文的量子遗传聚类集成算法的量子染色体编码中,若有 m 个体的种群 $p=\{p_{11},p_{12},\cdots,p_{mn}\}$,则 $p_{ij}(i,j=1,2,\cdots,n)$表示种群的第 i 个体第 j 个量子比特,每一个量子染色体表示一个聚类中心,所有的聚类中心集合 $V=\{v_1,v_2,\cdots,v_k\}$ 表示一个初始种群。

2.2.2 适应度函数

适应度是决定整个算法的优劣,反映个体好坏的关键。在本文中,为了获得最优的聚类中心,使得同类样本点到该聚类中心的距离尽可能的小,不同类的样本点之间的簇间距离尽可能地大,因此选择适应度函数为 $f(s)=J/D$。其中:D 为各聚类中样本点到对应聚类中心距离的总和 $D=\sum_{i=1}^{k}\sum_{j}^{n}D_{ij}$;$J$ 为聚类之间的加权距离和 $J=\sum_{i=1}^{k}\sum_{k=1}^{k}J_{ik}$,$D_{ij}$是第 i 个数据点到第 j 个聚类中心的欧式距离,J_{ik}是第 i 个聚类中心到第 k 个聚类中心的距离。这样的定义,满足了聚类目标函数的最小值,保证了最优聚类中心的获得。

2.2.3 量子旋转门

量子遗传算法的染色体的更新主要是通过量子旋转门来实现的,且量子旋转门的更新决定染色体的最优搜索方向,加快搜索的速度。在本文中采用进化方程来实现染色体的自动更新,其进化方程的数学表达式为

$$\theta=k_1(f(p)_{\text{opt}}-f(x_i)_{\min})+k_2(f(p)_{\text{opt}}-f(x_i)_{\max})+k_3(f(p)_{\text{opt}}-f_{\text{avg}})$$

其中:当前染色体 x_i 测量的适应度最大值、最小值分别为 $f(x_i)_{\max}$、$f(x_i)_{\min}$;该次测量种群的平均值为 f_{avg};$f(p)_{\text{opt}}$ 为种群

全局最佳适应度值;$f(p)_{opt} - f(x_i)_{min}$ 值指明搜索方向;$f(p)_{opt} - f_{avg}$ 值是为了实现局部精化;$f(p)_{opt} - f(x_i)_{max}$ 值指明调节的距离;k_1、k_2、k_3 为影响因子,满足 $k_1 + k_2 + k_3 = 1$ 条件。这样计算能快速得到 θ,无须像传统量子遗传算法既要查表又要控制它的步长那样繁琐,而且对 $\Delta\theta_i$ 的取值太大,则适应度值高的个体大量繁殖而出现早熟现象,算法容易陷入局部最优;取值太小则染色体更新缓慢,算法处于停滞状态。

2.3　量子遗传算法的 XML 聚类集成

XML 聚类集成阶段是组合阶段,是把通过 KNN 划分算法求得初始聚类中心 $V = \{v_1, v_2, \cdots, v_k\}$ 和求解内联相似度矩阵获得的特征值对应的特征向量映射成空间的点作为输入,利用 QGA 算法快速搜索求得全局最优的解作为 XML 文档聚类集成的输出。其算法具体步骤描述如下:

输入:给定一个 XML 文档集 D,分类数目 K。

输出:最终聚类集成结果和聚类中心 C'_i。

a)对 XML 文档集预处理,得到每个文档的路径和关键词特征向量;

b)运行 K 次 KNN 划分算法,得到 k 个基聚类 $\Pi = \{\pi^1, \pi^2, \cdots, \pi^k\}$ 和 k 个聚类中心 $V = \{v_1, v_2, \cdots, v_k\}$;

c)利用初始聚类结果构造内联相似度矩阵 W,运行多次分割、向下、向上双向收缩的 QR 算法获得它的前 k 个特征值对应的特征向量 $x_1, x_2, \cdots, x_k$,构造矩阵 $X = [x_1, x_2, \cdots, x_k] \in R^{n \times k}$,规范化 X 为矩阵 $Y = X_{ij} / (\sum_j X_{ij})^{1/2}$;

d)将 Y 的每一列作为 R^k 空间的一点,用所有初始聚类产生的 k 个聚类中心代表一个初始化种群,用每个聚类中心代表为一个量子染色体,即种群个体为 m、量子比特位数为 k 的种群表示为 $p(t) = \{p_{11}, p_{12}, \cdots, p_{mk}\}$;

e)调用量子遗传算法获得最佳聚类中心 $c'_1, c'_2, \cdots, c'_k$;

f)计算 Y 中的每一个点到最终得到的所有聚类中心的距离,将 Y 判别到具有最近距离的那一类聚类中心所在的类别中;

g)将文档集中的每个文档 d_i 根据 Y 的判别结果判到与之对应的相应类别中,输出 XML 文档集的聚类集成结果和聚类中心。

3　实验结果与分析

3.1　实验设计

为了评价本文提出的聚类集成算法,在实验中使用的计算机配置为 Intel 酷睿 i5-600 CPU,4 GB 主存;开发环境及编程语言为 Windows 2003 Server、Visual. NET;选用实验数据为 DBLP 归档数据集、PubMed 数据集和 Wikipedia 数据集,它们是差异性极大的真实 XML 数据集,其中 DBLP(http://dblp. uni-trier. de/xml)包含期刊、会议、书及其章节、计算机科学等方面的论文;PubMed(http://www. ncbi. nlm. nih. gov/entrez/)包含超过二千一百万引用和检索,其内容涉及到医学、护理、牙科、兽医、卫生保健、临床科学等多个方面,本文只选择部分有关蛋白质的文章作为实验数据;Wikipedia 是一个包含新闻、科技、文化、博客等内容的 Web 网站,选用 Index 2007 年的 XML 文档作为实验数据。表 2 显示三个数据集的统计情况,其中 size、#docs、#class、#terms 表示从三个数据集中抽取的文档大小、文档个数、文档类别及词条数等。

表 2　DBLP、PubMed、Wikipedia 数据集的统计

数据集	size/MB	#docs	#class	#terms
DBLP	830	4 000	5	7 329
PubMed	580	2 000	11	16 380
Wikipedia	360	48 305	73	535 351

3.2　聚类评价措施及结果

为了评价本文提出的算法,在实验中,使用信息检索常用评估指标召回率(recall)、精确率(precision)和聚类的计算时间来与单一聚类算法 K-means、谱聚算法和核聚算法的聚类实验进行对比,并假定给定的类别 c_i,A_i 为正确聚类到 c_i 的文档个数,B_i 为错误聚类到 c_i 的文档个数,C_i 为本应属于类 c_i 却聚类到其他类别的文档个数,则精确率和召回率分别为 $P = (\sum_i A_i)/(\sum_i A_i + \sum_i B_i)$ 和 $R = (\sum_i A_i)/(\sum_i A_i + \sum_i C_i)$。精确率和召回率越高,说明聚类质量越好。为了与 MCLA、HGPA 和 CSPA 三种典型的聚类集成算法进行对比,使用聚类质量来衡量聚类集成质量的好坏,其表达式为 $Q(C) = D - J$。其中:D 为各聚类中样本点到对应聚类中心距离的总和,J 为聚类之间的加权距离和。使用 NMI 来量化聚类结果和已知类别标签的匹配程度。使用 ANMI 来度量最终聚类结果和 k 个聚类标签之间的平均标准互信息。表 3(a)~(c)显示与单聚类算法和本文算法的聚类时间、聚类精度和召回率比较。表 4(a)和(b)显示与三种聚类集成算法的 NMI、ANMI 的百分比比较。图 1 显示不同的阈值参数对聚类集成质量的影响。

表 3　单聚类算法与本文算法的比较

(a)单聚算法与本文算法的聚类时间比较　/s

数据集	算法			
	K-means	谱聚算法	核聚算法	本文算法
DBLP	208.438	198.617	197.547	234.783
PubMed	232.641	227.693	223.674	258.426
Wikipedia	246.749	234.751	230.372	274.671

(b)单聚类算法与本文算法的聚类精度比较　/%

数据集	算法			
	K-means	谱聚算法	核聚算法	本文算法
DBLP	87.51	98.24	98.56	98.79
PubMed	85.46	96.62	97.57	97.86
Wikipedia	84.36	95.23	95.67	96.73

(c)单聚类算法与本文算法的聚类召回率比较　/%

数据集	算法			
	K-means	谱聚算法	核聚算法	本文算法
DBLP	97.37	98.46	98.69	98.77
PubMed	90.13	92.36	93.07	94.69
Wikipedia	89.71	90.47	91.25	92.46

表 4　不同聚类集成算法的百分比比较

(a)不同聚类集成算法的 NMI 比较

数据集	算法			
	CSPA	HGPA	MCLA	本文算法
DBLP	0.836	0.750	0.778	0.908
PubMed	0.769	0.796	0.756	0.832
Wikipedia	0.597	0.533	0.534	0.783
Avg	0.734	0.693	0.689	0.841

(b)不同聚类集成算法的 ANMI 比较

数据集	算法			
	CSPA	HGPA	MCLA	本文算法
DBLP	0.863	0.879	0.865	0.976
PubMed	0.727	0.632	0.567	0.789
Wikipedia	0.763	0.572	0.723	0.895
Avg	0.784	0.694	0.718	0.887

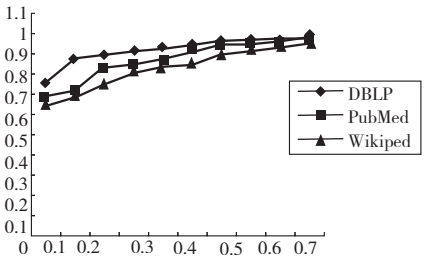

图1 不同阈值参数下的初始聚类对聚类集成质量的影响

3.3 实验结果分析

从表 3 中的聚类实验结果可以看出,对比单一聚类算法,本文算法除在聚类时间比 K-means、谱聚算法、核聚算法要长外,其聚类精确率和召回率都要高,说明本文算法的聚类质量高。

从表 4 中的实验结果来看,比较 CSPA、HGPA 和 MCLA 的 NMI 和 ANMI 值,CSPA 得到最好的聚类结果,而 HGPA 和 MCLA 的 NMI 值互有高低,它们都比本文算法要差;同时比较它们的平均 NMI 值,CSPA 总体要比 HGPA 和 MCLA 高 0.04% 左右,但比本文算法差 0.1% 左右,同样在 AMNI 的值上,HGPA 和 MCLA 的值比 CSPA 差 0.09% ~0.07%,比本文算法差 0.1% 左右。

在聚类质量上,图 1 反映不同的阈值参数的初始聚类结果在使用本文算法进行聚类集成时对聚类质量的影响,说明 XML 文档的结构影响聚类集成的质量,这正也说明了聚类成员的差异性影响聚类集成的质量。

4 结束语

本文提出的量子遗传算法的 XML 文档聚类集成比 CSPA、HGPA、MCLA、HBGA 等聚类集成算法有更好的聚类质量;与单一的 K-means、谱聚算法和核聚算法等相比有更高聚类精度和较高的召回率,但在计算时间上高于单一聚类算法。该算法的创新点有:a)在内容与结构相结合的 XML 文档相似度中引入阈值参数,这样在不同阈值参数的情况下,使用 KNN 划分算法就可以获得不同的初始聚类成员,解决了不同的初始成员的差异性对聚类集成质量的影响;b)在获得一致性内联矩阵后,调用多次分割、向下、向上双向收缩的 QR 算法求解矩阵的特征值和对应的特征向量,避免因数据规模大、数据的计算量成倍增加而增加算法的时间复杂度;c)解决了高维、大数据集的聚类集成问题;d)使用量子遗传算法快速实现了 XML 聚类集成。

参考文献:

[1] 唐伟,周志华. 基于 Bagging 的选择性聚类集成[J]. 软件学报,2005,16(4):496-502.

[2] 王红军,李志蜀,成飚,等. 基于隐含变量的聚类集成模型[J]. 软件学报,2009,20(4):825-833.

[3] YAN Dong-hui, HUANG Ling, JORDAN M I. Fast approximate spectral clustering[EB/OL]. (2009). http://www. citeseerX. ist. psu. edu/.

[4] LUXBURG U V. A tutorial on spectral clustering[J]. Statistics Computing,2007,17(4):395-416.

[5] STREH1 A, GHOSH J. Cluster ensembles: a knowledge reuse framework for combining partitionings[J]. Journal of Machine Learning, 2002,3:583-617.

[6] FERN X Z, BRODLEY C E. Solving cluster ensemble problem by bipartite graph partitioning[C]// Proc of the 21st International Conference on Machine Learning. New York: ACM Press,2004.

[7] AZIMI J, FERN X L. Adaptive cluster ensemble selection[EB/OL]. (2009). http://www. citeseerX. ist. psu. edu/.

[8] 王红军,李志蜀,戚建雄,等. 基于贝叶斯网络的半监督聚类集成模型[J]. 软件学报,2010,21(11):2814-2825.

[9] FRED A L N, JAIN A K. Combining multiple clusterings using evidence accumulation[J]. IEEE Trans on Pattern Analysis and Machine Intelligence,2005,27(6):835-850.

[10] 周珠,潘炜,罗斌,等. 一种基于粒子群优化方法的改进量子遗传算法及应用[J]. 电子学报,2006,34(5):897-901.

[11] 蒋勇,谭怀亮,李光文. 基于量子遗传算法的 XML 聚类方法[J]. 计算机应用,2011,34(2):446-449.

[12] BURKHOLDER J J, SQUIRE K, KOLSCH M. Nearnest neighbor classification using sensitive distance mearsurement[EB/OL]. (2009). http://edocs. nps. edu/npspubs/scholarly/theses/2009/sep/.

[13] 赵学智,叶邦彦,陈统坚. 大型矩阵奇异值分解的多次分割双向收缩 QR 算法[J]. 华南理工大学学报:自然科学版,2010,38(1):1-8.