面向线性文本的 K-means 聚类算法研究

文必龙,李 菲,马 强
(东北石油大学 计算机与信息技术学院,黑龙江 大庆 163318)

摘 要:鉴于线性文本内容组织形式的有序性,将有序的主题内容进行正确的划分,用于挖掘文本中隐藏的信息、知识,是一个值得研究的问题。同时,传统的 K-means 聚类算法在对线性文本进行聚类时,会造成计算复杂度增加以及无穷迭代或聚类结果混乱等一系列问题。针对以上问题,对传统的 K-means 算法进行研究,将随机初始化中心点的算法进行改进,提出一种随机均匀初始化中心点算法。该算法充分考虑线性文本的组织结构特性,随机化第一个中心点后,均匀地确定其他中心点,保证了文本子主题的完整划分;与此同时,又采用了设定约束规则的等距点归类法,实现文本迭代次数限制下的自动归类。实验结果表明,该算法在对线性文本进行聚类时,可以有效减少迭代次数并提高聚类精度,最终获得较好的聚类效果。

关键词:线性文本;组织结构;随机均匀取点;等距点归类;K-means 算法
中图分类号:TP301.6 **文献标识码**:A **文章编号**:1673-629X(2018)09-0053-06
doi:10.3969/j.issn.1673-629X.2018.09.012

Research on K-means Clustering Algorithm of Linear Text

WEN Bi-long,LI Fei,MA Qiang
(School of Computer and Information Technology,Northeast Petroleum University,Daqing 163318,China)

Abstract:In view of the orderliness of the organized form of linear texts,it is worthwhile studying to mine the hidden information and knowledge from the text by dividing the subject content correctly. At the same time,the traditional K-means clustering algorithm will conduce to a series of problems such as increasing computational complexity,infinite iteration phenomenon or clustering results confusion. For this,we research the traditional K-means algorithm and improve the algorithm of randomly initializing center,based on which we propose a random uniform initialization center algorithm. This algorithm gives plenty of considerations to the organizational structure of linear texts. After one central point is randomized,other central points are uniformly determined to ensure the sufficiently division of the subtopic. Meantime,we adopt an equidistant point categorization under the constraint rules to realize automatic classification under the limit of text iteration. The experiment illustrates that the proposed algorithm can effectively cut down iteration times and improve the clustering accuracy when clustering linear texts,obtaining the better clustering outcome at last.

Key words:linear text;organizational structure;random and even center point selection;isometric point classification;K-means algorithm

0 引 言

一篇具有明确主旨的文章,多采用一定的组织形式去组织文本内容。从文本内容中挖掘有用信息,是目前文本挖掘、文本信息抽取等相关领域研究的重点[1]。作为一种能将不同组织形式的文本根据内容聚集成簇的关键技术,聚类技术为文本内容的进一步分析挖掘提供了有力的支撑。K-means 算法是基于划分思想的经典聚类算法[2],是一种采取随机确定初始点作为中心点,然后不断循环迭代求得最大相似性的类别划分算法[3]。该聚类算法针对主题掺杂、内容组织无序的文本,具有简单、收敛速度快、处理大数据文本集有效等优点[4]。传统 K-means 算法随机初始化中心点,在迭代聚类时会有以下问题:需要输入最终结果的聚类个数 k[5],而判断一个未知数据集的划分个数通常是很困难的;k 个初始点的选择对最终的聚类结果影响很大[6];聚类过程中的迭代总次数增加使得聚类过程中的总耗时增加[7]。为解决以上问题,文献[8]从样本几何结构角度,设计一种新的聚类有效性指标,依此确定最佳聚类数。文献[4]和文献[9]在初始化中心点上分别采用最大距离积法、密度区域相距

收稿日期:2017-11-07 **修回日期**:2018-03-16 **网络出版时间**:2018-05-16
基金项目:国家重大专项(2016ZX05033-005-004)
作者简介:文必龙(1967-),男,博士,教授,研究方向为大数据、软件工程;李 菲(1991-),女,硕士研究生,研究方向为知识工程。

最远来确定初始化中心点。文献[10]和文献[11]分别提出了基于最近高密度点间的垂直中心点优化初始聚类中心和基于密度峰值优化的 K-means 文本聚类算法,解决了聚类效率低和局部最优解等问题。在对整篇文章的内容和组织结构进行分析时,发现文本具有基于某一主题下的有序组织的线性文本,对其采用传统的 K-means 算法会存在以下问题:(1)篇章主题内容划分的随意性较大。在不考虑线性文本具有的上下文内容划分的清晰界限,采取文本段落向量的相似性进行聚集分类时,聚类主题的侧移影响最终结果;(2)随机初始中心点的方式增大了聚类初始点的不确定性,在选择不当的情况下使得迭代次数增加或无穷迭代、延长运算时间等。同时,该算法在处理段落文本到各个中心点的距离相等时,归类不当也会造成聚类结果的不精确等问题。

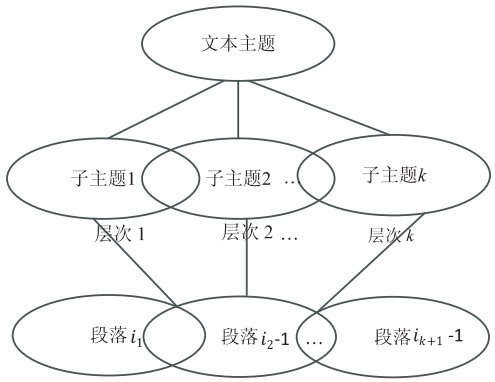
针对以上问题,文中深入分析线性文本内容的组织特性,提出一种随机均匀初始化中心点的 K-means 文本聚类算法,主要用来解决线性文本自身段落内容、层次、主题等的聚类问题。同时改进收敛函数,提出等距点归类法以解决特殊段落到中心点距离相同时无法准确归类的问题。

1 线性文本

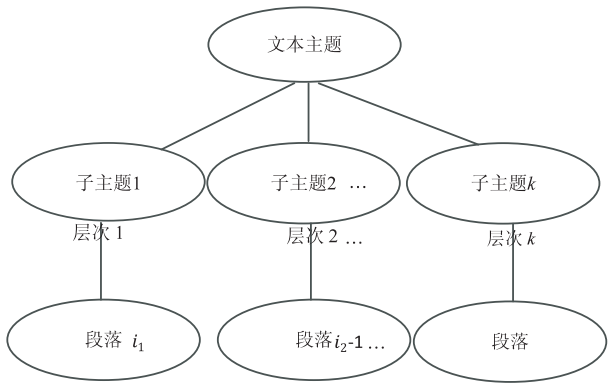
1.1 定 义

线性文本指的是阅读时有先后顺序,基于一个共同主题下划分各个相关子主题,子主题之间相互独立、均匀分散、段落在组织上具有线性结构的一类文本。传统的教材课文不管文字排列的方式如何,文章的写作和学习者的知识学习都要依靠一种相继的线性顺序进行,段落和章句之间必然依照逻辑、衔接和顺序来联结成一体,这是线性文本的特点。

线性文本具有较强的思维逻辑性和层次结构性[12]。与非线性文本相比,避免了让读者在阅读中肆意游荡。非线性文本中的各子主题[13]内容之间相互融合掺杂,文本段落在组织上杂乱无序、胡乱堆砌、毫无界限和标志之分(结构见图 1(a))。在采用传统的 K-means 文本聚类分析时,随机初始化中心点可保证杂乱主题被任意选取到,但是因为不确定性的存在,会使得聚类迭代次数增加或无穷迭代、文本中心意义的曲解和偏差等。线性文本从始至终是基于一个主题的,主题一般以抽象概括的语言显性或隐性地存在于整篇的篇章当中[14],并且以主题为轴心做逻辑导向划分子主题,实现文本内容的层次划分。表现层次的完整的单位是段落,文本最终形成一棵文本的结构树[15](结构见图 1(b))。文中把线性文本的逻辑结构表示为:文本={文本主题,层次主题,段落主题,句子,主题词}。



(a)非线性文本



(b) 线性文本

图 1 线性与非线性文本对比

在对线性文本进行结构分析时,其有序化的组织特性,决定了 K-means 聚类分析的有序性。文中基于一篇线性文本,对其内容进行 K-means 划分。具体定义如下:设文本 d 具有个 n 自然段,k 个子主题(也是 k 个内容层次,认为内容层次是依据子主题进行的划分),用 H 表示划分的文本内容,P 表示自然段。

定义 1:待分析文本 d。

$d=\{P_1,P_2,\cdots,P_n\}$

定义 2:文本聚类分析后的内容划分[14]。

$d=\{H_1,H_2,\cdots,H_k\}=\{P_{i_1}\cdots P_{i_2}-1\}\{P_{i_2}\cdots P_{i_3}-1\}\cdots\{P_{i_k}\cdots P_{i_{k+1}}-1\}$

其中,$i_1=1\leqslant i_2-1\leqslant\cdots\leqslant i_k\leqslant i_{k+1}-1=n$(为方便以后表示,$d=P_1,P_2,\cdots,P_n$ 简记为 $1,2,\cdots,n$)。

而在文本逻辑结构中更加强调的是文本所包含的思想内容(内容划分),段落单元是该段落的中心思想,作为文本结构树的叶子节点,段落间在表现主题时用词上会存在差异,也就支撑了段落中心思想的聚集程度。线性文本的有序聚类就是寻找一种分法使 k 个内容层次内的差异尽可能小,而层次间的差异尽可能大。

1.2 线性文本的空间向量模型

为了让计算机能对文本进行操作,采用向量空间

模型(VSM)对文本进行表示[16-17]。其基本思想是:将文本中不同的词语(一个词语是一个维度),按照它们的重要程度,赋予不同权重[17]。最后文档集合 D 中的任一文本 d_k 都表示成向量形式:$d_k = (W_{k1}, W_{k2}, \cdots, W_{kh})$,其中 W_{kg} 是文本 d_k 中第 g 个词语的权重,h 是 D 的维度,也称文本向量的基数[18]。那么,针对线性文本有:

定义 3:设文本 d 的特征项集为 $T = \{t_1, t_2, \cdots, t_m\}$(为了方便表示,亦可记为 $1, 2, \cdots, m$)。则设 $P_i = \{W_{i1}, W_{i2}, \cdots, W_{im}\}$ 为第 i 段的特征向量[19]。其中 W_{iq} 是特征项 $t_q (q \in [1, m])$ 在第 i 段中的权重,特征项计算的是词语的权重,形成如下文本空间矩阵[11]:

$$d = \begin{bmatrix} \text{段落编号} \\ P_1 \\ P_2 \\ \vdots \\ P_n \end{bmatrix} = \begin{bmatrix} t_1 & t_2 & \cdots & t_m \\ W_{11} & W_{12} & \cdots & W_{1m} \\ W_{21} & W_{22} & \cdots & W_{2m} \\ \vdots & \vdots & \vdots & \vdots \\ W_{n1} & W_{n2} & \cdots & W_{nm} \end{bmatrix}$$

在该模型中,使用 TF-IDF 作为特征词权重的度量[16-17]。

$$W_{kq} = \mathrm{TF}_q \times \log(N / \mathrm{DF}_q) \tag{1}$$

计算 TF(term frequency),有不同的归一化方式:

$$\mathrm{TF}_q = \frac{\mathrm{tf}_q}{\mathrm{sum}(\mathrm{doc\_length})} \tag{2}$$

$$\mathrm{TF}_q = \frac{\mathrm{tf}_q}{\max(\mathrm{tf}_d)} \tag{3}$$

其中,$\mathrm{sum}(\mathrm{doc\_length})$ 为文本总词频;$\max(\mathrm{tf}_d)$ 为文本 d 中的最大词频,文中选用的是单个段落的总词频;n 为自然段落总数;DF_q 为包含词语 q 的段落总数目。

1.3　K-means 聚类算法的不足

K-means 是一种基于迭代思想的聚类算法,从 v 篇预处理的文本集合 $D = \{d_1, d_2, \cdots, d_v\}$ 中选取 k 个初始簇中心,并依据相似程度将文本划分到最相似的簇中,最终形成 k 个簇的集合 $C = \{c_1, c_2, \cdots, c_k\}$。具体算法的实现步骤如下[20]:

定义 C 中元素的复合向量 $S(c_j) = \sum_{d \in C_j} W_d (j \in \{1, 2, \cdots, k\})$,即第 j 簇的文本向量之和,记第 t 次迭代产生的簇集合为 $C^{(t)}$,中心点为 $Z_j^{(t)}$,$t = 0, 1, 2\cdots$。

(1)任意选取 k 个中心点 $Z_1^{(0)}, Z_2^{(0)}, \cdots, Z_k^{(0)}$,迭代次数 $t = 0$;

(2)计算 $d_y (d_y \in D)$ 到中心点 $Z_j^{(t)}$ 的距离,d_y 归类到最近的中心 $Z_{j^*}^{(t)}$;

(3)计算 k 个簇的平均值 $S(c_j)/\mathrm{length}(c_j)$,更新中心点 $Z_j^{(t=t+1)}$;

(4) $Z_j^{(t)} ? = Z_j^{(t=t+1)}$,若二者不同,转步骤 2,反复迭代。否则转步骤 5;

(5)输出最终簇集合 C^*。

传统的 K-means 算法在处理线性文本时,采取随机挑选中心点并不断迭代的聚类方式,中心点的不确定性较大,在选择不当的情况下造成迭代次数增加、运算时间加长[21]。例如:初始化中心点时,在本属于同一簇的文本中选取多个中心点,以及忽略线性文本具有的上下文内容划分的清晰界限,在中心点选取上不均匀,使得聚类中心主题的偏移,影响聚类最终结果;同一个文本到多个中心点距离相等以及孤立点时,会干扰文本的聚类效果,最终无法准确归类(见表 1)。因此,急需改进中心点选取算法及处理等距点现象的归类方式。

表 1　文本到中心点距离对比

相似性	文本段落	中心点 1	中心点 2	中心点 3	中心点 4	中心点 5
文本 1	段落 3	0.334 236 302 646 757	0.115 977 683 518 284	0.334 236 302 646 757	0.003 642 825 487 315 88	无
文本 2	段落 7	1	0.002 842 603 536 854 92	0.001 192 346 813 615 37	无	无
文本 3	段落 9	0.536 554 779 311 918	0.032 816 121 379 961 7	0.002 861 322 007 718 47	0.015 677 054 021 242 8	0.007 399 789 347
文本 4	段落 1	0.317 918 870 784 118	0.004 996 853 763 414 34	0.002 095 959 772 674 11	无	无
文本 5	段落 11	0.003 104 590 875 046 88	0.059 778 467 808 891 2	0.018 173 687 296 230 2	无	无
文本 6	段落 23	0.018 150 226 730 43	0.003 172 751 443 743	0.000 343 946 221 45	无	无
文本 7	段落 7	0.002 842 603 536 854 92	1	0.012 649 522 917 928 9	0.015 519 371 829 305 9	无
文本 8	段落 3	0.001 425 258 531 636 22	0.144 587 718 756 867	0.003 297 636 635 963 78	0.001 425 258 531 636 22	无

2　随机均匀初始化中心算法

针对线性文本特性采取均匀初始化中心点的方式,可以精确地确定主题范围。因为线性文本的段落表意明确、集中,含有丰富的语义,在篇章当中段落间会存在并列、顺承等一些线性特征,也就使得表现主题内容的各个主题之间线性排列。

具体采用的随机均匀初始点算法(如图 2 所示)如下:

设具有 n 个自然段的文章 $d = \{P_1, P_2, \cdots, P_n\}$,$P$ 表示段落,共有 n 个自然段落,聚类数目为 k。

为使聚类结果有意义(过大或过小的 k 值都会影响聚类结果),在选定 k 值时,默认取值范围是 $[K_{\min}, K_{\max}]$,其中 $K_{\min} = 2$,$K_{\max} = \mathrm{sqrt}(n)$[22]。

一篇线性文本 W 可划分成具有 k 个子主题的簇集 C，k 个主题的内容在段落形式上呈线性排列，则选取初始化中心点也呈线性排列。其中，段落均匀间隔为 $\mathrm{dis} = (n/k)$。

(1)为了保证随机选取的中心点有意义，随机选择的第一个中心点为 $P_x(x \in [1,\mathrm{dis}])$。

(2)根据 P_x 及 dis 获取其他中心点 $p = P_{x+r*\mathrm{dis}}(r \in [1,k-1])$。

(3)形成初始点簇成员集 $C_{\mathrm{start}} = \{P_x, p\}$。

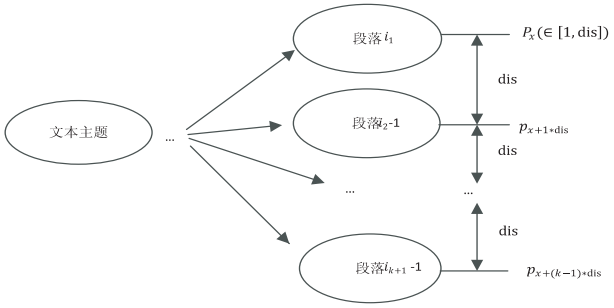


图 2 随机均匀初始化中心点

文本中，各子主题间为了突出各自内容，相互之间相似程度较小，从而在整篇文章上呈现主题间的并列或递进等线性排列特征。同时为避免文章冗余，主题内容的规模分布上多呈现出均匀分布特性。根据这种均匀特性，采用随机均匀初始中心点，可以更好地保证初始点间的相似度小。并且，该算法可使中心点均匀地分布到各个子主题内容中，避免随机性太大造成的初始点过于集中与分散的情况，有利于相似内容最快归类，提高聚类效果与速度。

3 等距点归类法

通过前面的模型，得到随机均匀选取初始点的 K-means 算法，但该算法在迭代时需要解决文本段落归类的问题。实验中发现，由于篇章内容少这个特性，使得对段落聚类时，每个段落向量有可能与其他内容都不相似或与多个簇的中心相似度相等，将这样的段落称为“等距点”，等距点可能即使多次迭代，仍不能将其划分到相近的类中。为解决该问题，提出如下归类处理方法。

定义 4：簇的平均值。

$$\mathrm{mean} = \frac{\sum_{h=1}^{k}\sum_{i_h=1}^{i_{h+1}-1}\sum_{q=1}^{m} W_{(i_h)q} + \cdots + W_{(i_{h+1}-1)q}}{\mathrm{length}(i_h)} \tag{4}$$

其中，h 为文本内容层次，i_h 属于文本内容 h 的一个自然段落。

该公式用于计算任意簇中所有自然段落空间向量坐标的平均值，计算结果作为簇更新后的中心点。

定义 5：最大迭代次数 $\max = \varepsilon$。

(1)计算非中心点 $p_i(i \leqslant (n-k))$ 到簇集 C_{start} 即 $(C_{\mathrm{start}}, p_i)$ 之间的相似度 $\mathrm{sim}(p_i, C_{\mathrm{start}})$，选取最大相似度的簇对 $\mathrm{sim}(p_i, C_z)(z \in \{1,2,\cdots,k\})$，将 p_i, C_z 合并成新簇，$C_{\mathrm{new}} = p_i \cup C_z$；当段落到多个中心点距离相等时，默认先不进行归簇(增加一次迭代)。

定义 6：计算任意两个段落之间的相似度-夹角余弦距离[19,23]。

$$\mathrm{sim}(p_i, p_e(p_e \in C_{\mathrm{start}})) = \left[\frac{\sum_{q=1}^{m}(W_{iq} * W_{eq})}{\sqrt{\sum_{q=1}^{m}(W_{iq})^2 * \sum_{q=1}^{m}(W_{eq})^2}}\right] \tag{5}$$

(2)计算新簇的平均值 $\mathrm{mean}(C_{\mathrm{new}})$，从而构成 $C_{\mathrm{new}} = \{C_{\mathrm{new1}}, C_{\mathrm{new2}}, \cdots, C_k\}$。

(3)判断 $C_{\mathrm{new}}? = C_{\mathrm{start}}$，若相等或者 $t = \varepsilon$，执行步骤 4，否则，进行赋值：$C_{\mathrm{start}} = C_{\mathrm{new}}$，$t = t + 1$。然后跳转到步骤 1。

(4)判断 $d = \{p_1, p_2, \cdots, p_n\}$ 都合并，未合并的，将其单独并为一类 C_{k+1}。

(5)输出聚类结果。

4 实验结果分析

将改进的 K-means 算法聚类结果进行评价研究的过程称为聚类有效性分析(cluster validity)。聚类有效性分析一般分为外部标准评价和内部标准评价[24]。外部标准评价(external criteria appraisal)，用于标定的聚类结果集，采用相应的评价指标来评价聚类质量。内部标准评价(internal criteria appraisal)，直接评价聚类算法的目标函数值，由该标准衍生出来的评价指标称为基于目标函数的指标[24]。

为验证该算法的有效性，以《人民日报》语料中的整篇文档作为实验文本，选取 7 个类别共 8 篇，每篇的段落数如表 2 所示：

表 2 文本段落数统计及分类

文本	段落数	分类
文本 1	18	体育
文本 2	13	经济
文本 3	26	经济
文本 4	14	国际
文本 5	15	军事
文本 6	25	文化
文本 7	15	教育
文本 8	23	政治

基于内部标准评价，采用类内类间相似性度量函数[25]，对聚类质量进行评判。

具体计算公式如下:

$$\mathrm{sim} = \frac{1}{c_n^2}\sum_{i=1}^{n}\sum_{j=1}^{i} d(X_i, X_j) \tag{6}$$

其中,$d(X_i, X_j)$ 为文本之间的余弦相似值。该值越大,文本的相似性越高,反之,相似性越低。

实验结果如表 3 所示。

表 3　聚类实验效果

算法比较	传统 K-means 算法			改进 K-means 算法		
	类间相似性	类内相似性	迭代次数	类间相似性	类内相似性	迭代次数
文本 1	0.021 080 07	0.127 865 1	4	0.009 457 514	0.487 947 5	30
文本 2	0.057 526 42	0.216 474 4	5	0.002 122 499	0.498 470 6	2
文本 3	0.025 415 18	0.432 052 3	3	0.018 708 5	0.459 363 3	2
文本 4	0.101 964 4	0.248 178 2	4	0.021 492 77	0.311 496 8	2
文本 5	0.066 270 1	0.242 471 4	4	0.010 012 34	0.254 462 1	3
文本 6	0.026 826 147	0.210 975 7	8	0.086 020 43	0.278 056	3
文本 7	0.113 873 01	0.220 940 7	6	0.061 705 59	0.394 327 7	4
文本 8	0.165 415 67	0.101 338 70	5	0.044 627 95	0.223 948 8	30

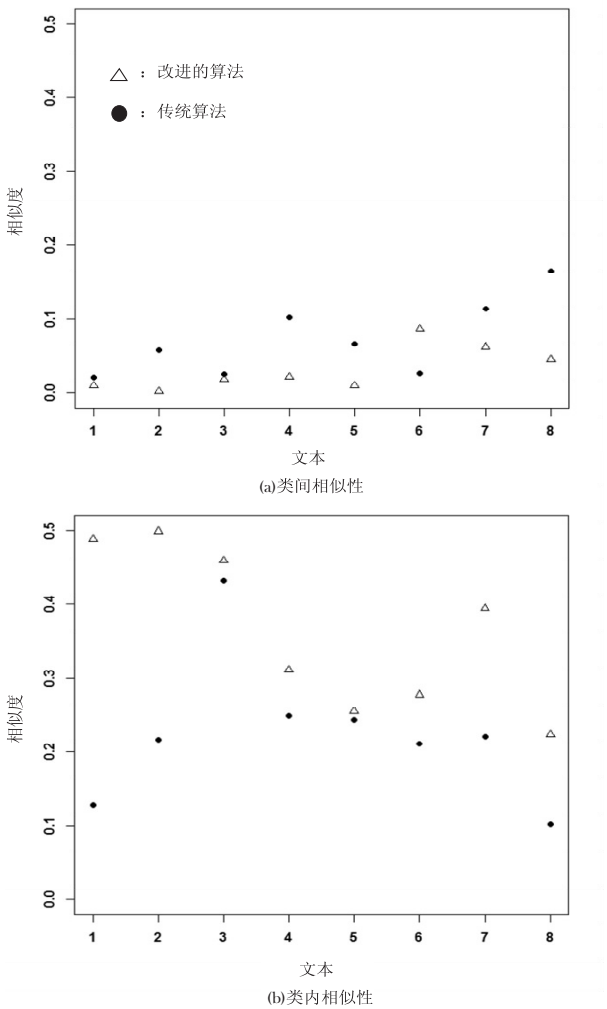


图 3　相似度对比

由表 3 可以看出,当未出现孤立点及文本段落到多个中心点距离相等时,改进算法降低了聚类迭代次数,缩短了聚类时间。相反的情况下,采取最大迭代限制并进行优化归类,提高了聚类结果的准确度。如图 3 的实验结果可以看出,传统 K-means 聚类算法类间相似度大于改进之后的算法结果,说明传统算法在簇间区分上不如改进算法的簇间区分性好,并且改进算法很大程度上降低了文本的耦合性[26];在类内相似性上,传统算法类内相似性小于改进之后的计算结果,说明簇内文本之间的紧凑程度要劣于文中算法。

5　结束语

针对组织有顺序的线性文本,考虑文本结构化特性,对传统 K-means 聚类算法在内容聚类上的不足进行改进,提出一种新的中心点确定方法—随机均匀选点;基于文本分布和迭代次数的等距点归类方法,构造了一种基于线性特征的自动文本内容分析算法,对深入理解文本、挖掘文本中的主题和有用信息,具有重要的意义。实验结果表明,该算法提高了线性文本的聚类效率,在形成以子主题为中心的簇集分类上优于传统的 K-means 聚类算法。下一步将在此基础上,依据文本的语义特性、相似度等特征自动确定 k 值,以期达到更好的聚类效果。

参考文献:

[1] 周雪忠. 文本挖掘在中医药中的若干应用研究[D]. 杭州:浙江大学,2004.

[2] DUBES R C, JAIN A K. Algorithms for clustering data [M]. [s. l.]:Prentice Hall,1988:60-65.

[3] HARTIGAN J A,WONG M A. A k-means clustering algorithm[J]. Applied Statistics,1979,28(1):100-108.

[4] 熊忠阳,陈若田,张玉芳. 一种有效的 K-means 聚类中心初始化方法[J]. 计算机应用研究,2011,28(11):4188-4190.

[5] 吴夙慧,成　颖,郑彦宁,等. K-means 算法研究综述[J]. 现代图书情报技术,2011,27(5):28-35.

[6] 马　帅,王腾蛟,唐世渭,等. 一种基于参考点和密度的快速聚类算法[J]. 软件学报,2003,14(6):1089-1095.

[7] 翟东海,鱼　江,高　飞,等. 最大距离法选取初始簇中心的 K-means 文本聚类算法的研究[J]. 计算机应用研究,2014,31(3):713-715.

[8] 周世兵,徐振源,唐旭清. K-means 算法最佳聚类数确定方法[J]. 计算机应用,2010,30(8):1995-1998.
[9] 袁 方,周志勇,宋 鑫. 初始聚类中心优化的 k-means 算法[J]. 计算机工程,2007,33(3):65-66.
[10] 邓 海,覃 华,孙 欣. 一种优化初始中心的 K-means 聚类算法[J]. 计算机技术与发展,2013,23(11):42-45.
[11] 田诗宵,丁立新,郑金秋. 基于密度峰值优化的 K-means 文本聚类算法[J]. 计算机工程与设计,2017,38(4):1019-1023.
[12] HIND J. Organizational patterns in discourse,syntax and semantics:discourse and syntax[M]. New York:Academic Press,1979.
[13] 刘秋水. Web 信息抽取与网页摘要的研究与应用[D]. 大连:大连理工大学,2008.
[14] 曾利沙. 主题与主题倾向关联下的概念语义生成机制—也谈语篇翻译意识与 TEM8 语段翻译教学[J]. 外语教学,2007,28(3):83-87.
[15] 林鸿飞,战学刚,姚天顺. 文本层次分析与文本浏览[J]. 中文信息学报,1999,13(4):7-15.
[16] 郭庆琳,李艳梅,唐 琦. 基于 VSM 的文本相似度计算的研究[J]. 计算机应用研究,2008,25(11):3256-3258.
[17] 黄 磊,伍雁鹏,朱群峰. 关键词自动提取方法的研究与改进[J]. 计算机科学,2014,41(6):204-207.
[18] 陈磊磊. 不同距离测度的 K-Means 文本聚类研究[J]. 软件,2015,36(1):56-61.
[19] 沈 斌. 基于分词的中文文本相似度计算研究[D]. 天津:天津财经大学,2006.
[20] MACQUEEN J. Some methods for clustering and analysis of multivariate observations[C]//Proceedings of 5th Berkeley symposium on mathematical statistics and probability. Berkeley:University of California Press,1967:281-297.
[21] 索红光,王玉伟. 一种用于文本聚类的改进 k-means 算法[J]. 山东大学学报:理学版,2008,43(1):60-64.
[22] REZAEE M R,LELIEVELDT B P F,REIBER J H C. A new cluster validity index for the fuzzy c-means[J]. Pattern Recognition Letters,1998,19(3-4):237-246.
[23] 姚清耘. 基于向量空间模型的中文文本聚类方法的研究[D]. 上海:上海交通大学,2008.
[24] HALKIDI M,BATISTAKIS Y,VAZIRGIANNIS M. On clustering validation techniques[J]. Journal of Information Systems,2001,17(2-3):107-145.
[25] 常兴治. 基于全局评价的文本分割技术研究[D]. 沈阳:东北大学,2004.
[26] 刘务华,罗铁坚,王文杰. 文本聚类算法的质量评价[J]. 中国科学院研究生院学报,2006,23(5):640-646.

(上接第 52 页)

遇到的许多技术细节。工作实践表明,该测试软件在使用过程中效果良好,可以提高软件开发与测试的效率,降低软件的开发成本。同时弥补了手工测试时重复劳动的缺点,减少了测试人员大量的重复测试验证工作;也有助于做好软件项目的管理工作。

参考文献:

[1] 龚智勇. 基于 Selenium 的 OpenStack Horizon 自动化测试的实现[J]. 国外电子测量技术,2017,36(5):45-49.
[2] 吴伶琳. 基于 Selenium 的软件自动化测试的研究与应用[J]. 计算机与现代化,2013(2):65-68.
[3] 曹 磊,董科军,袁博文. 一种基于 Selenium 的 Web 应用软件自动化测试平台设计与实现[J]. 科研信息化技术与应用,2014,5(6):44-52.
[4] 李 艳,任洪敏,刘 芳. 基于 Selenium JS UI 的自动化测试框架设计与实现[J]. 微型机与应用,2017,36(17):24-26.
[5] 卢 晨. 基于 Selenium 进行 Web 应用测试研究[J]. 软件导刊,2015,14(1):154-155.
[6] 赵金丹. 基于 selenium 的 web 自动化测试脚本设计研究[J]. 科技传播,2014(1):94.
[7] 柏 莹. 基于. NET 平台下 Web 自动化测试的研究与设计[D]. 西安:西安电子科技大学,2013.
[8] 李潇烨. 企业项目管理系统的 Web 自动化测试研究与实现[D]. 西安:西安电子科技大学,2015.
[9] 刘 军. 基于 Selenium 的网页自动化测试系统设计与实现[D]. 武汉:华中科技大学,2014.
[10] BRUNS A,KOMSTADT A,WICHMANN D. Web application tests with selenium[J]. IEEE Software,2009,26(5):88-91.
[11] LEOTTA M,CLERISSI D,RICCA F,et al. Repairing selenium test cases:an industrial case study about web page element localization[C]//IEEE sixth international conference on software testing,verification and validation. Luxembourg:IEEE,2013:487-488.
[12] XU Dianxiang,XU Weifeng,BAVIKATI B K,et al. Mining executable specifications of web applications from selenium IDE tests[C]//IEEE sixth international conference on software security and reliability. Gaithersburg,MD,USA:IEEE,2012:263-272.
[13] GUNDECHA U. Selenium testing tools coolcbook[M]. Birmingham,UK:Packt Publishing Ltd.,2012.
[14] 虫 师. Selenium 2 自动化测试项目实战:基于 Python 语言[M]. 北京:电子工业出版社,2016.
[15] 吴晓华. Selenium WebDriver 实战宝典[M]. 北京:电子工业出版社,2015.
[16] 张秋杰. 基于 pyUnit 框架的企业级软件自动化测试技术的研究[D]. 北京:北京邮电大学,2010.
[17] 孙 利. Java Web 案例教程[M]. 北京:电子工业出版社,2015.
[18] 齐 伟. 跟老齐学 Python 入门到精通[M]. 北京:电子工业出版社,2016.