doi: 10.3969/j.issn.1005-3026.2015.01.005

一种面向医学短文本的自适应聚类方法

栗 伟[1], 许洪涛[2], 赵大哲[1,3], 刘积仁[3]
(1. 东北大学 医学影像计算教育部重点实验室, 辽宁 沈阳 110819; 2. 郑州市人力资源和社会保障数据管理中心, 河南 郑州 450000; 3. 东软集团股份有限公司, 辽宁 沈阳 110179)

摘 要: 针对电子病历中疾病诊断文本同义词识别和命名标准化问题,提出了一种自适应的文本聚类方法.首先提出了一种新的基于集合的文本相似性度量算法;然后采用基于相似度分布的文本聚类算法实现同义文本识别,该算法能够自动确定类簇个数;最后采用基于序列模式的中心概念提取算法实现了疾病命名的标准化,同时对聚类簇进行合并和优化,进一步提升了聚类的准确性.测试结果表明,所述方法具有较高的准确率和聚类效率,在病历文本的预处理、分类和分析中具有广泛意义.

关 键 词: 聚类分析;相似性度量;频繁序列模式;电子病历;相似度分布
中图分类号: TP 391 **文献标志码**: A **文章编号**: 1005-3026(2015)01-0019-05

An Adaptive Clustering Method on Medical Short Text

LI Wei[1], *XU Hong-tao*[2], *ZHAO Da-zhe*[1,3], *LIU Ji-ren*[3]
(1. Key Laboratory of Medical Image Computing, Ministry of Education, Northeastern University, Shenyang 110819, China; 2. The Zhengzhou Municipal Human Resources and Social Security Data Management Center, Zhengzhou 450000, China; 3. Neusoft Group Ltd., Shenyang 110179, China. Corresponding author: LI Wei, E-mail: l-w@neusoft.com)

Abstract: An adaptive clustering method on short text was presented for synonyms text recognition and disease naming standardization of diagnosis in electronic medical record. Firstly, a new set based text similarity measure algorithm was proposed. Then, a similarity distribution based text clustering algorithm which could automatically determine the number of clusters was applied to recognize the synonymous disease texts. Finally, the disease naming texts were standardized by the central concept extraction algorithm based on frequent sequence pattern, while clusters were merged and optimized to further improve the clustering accuracy. The results showed that the proposed approach has a high accuracy and clustering efficiency which is of great significance for medical application such as medical text preprocessing, classification and analysis.

Key words: clustering analysis; similarity measurement; frequent sequence pattern; electronic medical record; similarity distribution

电子病历(electronic medical record,EMR)记录了患者住院期间的完整诊疗信息,通常包含多个疾病诊断.然而,这些诊断文本存在着领域和医生特定的用语、同义词表达、缩略语以及拼写和打字错误等造成诊断文本不一致问题[1].这些问题严重影响了医学临床文本处理与分析的准确性.因此,电子病历记录中疾病命名同义词识别和标准化是本文要讨论的问题.

ICD-10(International Classification of Diseases, version 10)是医学应用最广的疾病分类系统,且在很多国家的医疗保险报销、健康统计和报告系统中应用.但该标准在实施中,不同的国家和医院根据实际需求进行了扩充[2],且存在版本问题[3]、编码准确性问题[1].由于电子病历诊断文本太短,将文本映射到ICD-10标准会匹配到多个分类,计算机无法自动区分正确的匹配结果.因

收稿日期: 2013-12-05
基金项目: 国家自然科学基金资助项目(61172002); 国家科技支撑计划项目(2014BAI17B01); 国家高技术研究发展计划项目(2012AA02A607).
作者简介: 栗 伟(1980-),男,河南驻马店人,东北大学博士研究生; 赵大哲(1960-),女,辽宁沈阳人,东北大学教授,博士生导师; 刘积仁(1955-),男,辽宁丹东人,东北大学教授,博士生导师.

此,基于 ICD - 10 标准,采用文本匹配方法无法解决上述疾病名称同义词识别和标准化的问题.

另外一种解决方法是采用聚类分析方法. 当前研究已经有很多聚类分析的成果,如层次聚类方法[4]、概念聚类方法[5]等. 但是这些方法存在以下问题:①由于无法预先知道数据集中总的疾病类别,因此,无法设置聚类簇的个数;②算法需要基于同义疾病名称,自动创建出一个标准的疾病命名,而上述算法无法自动生成.

本文针对上述疾病诊断文本中出现的问题和当前已有聚类算法存在的局限性,提出了一种面向医学短文本的自适应聚类方法(adaptive clustering method on medical short text,以下简称 ACT),该方法能够自动确定聚类簇个数,实现同义疾病文本的识别,同时为每一个类簇自动抽取产生标准的疾病命名. 本方法为病历数据的预处理、分类和分析等应用提供了疾病命名分类标准. 该分类标准来源于实际数据,因此具有更广泛的实际应用.

1 基于集合的短文本相似性度量

计算两个短文本之间的相似度是聚类算法运行的前提. 目前已有的研究工作中,研究者已经提出了一些文本相似性度量方法,一般分为两类:基于词特征的相似度量方法和基于词匹配的相似度量方法. 基于词特征的文本相似度量方法主要是对文本提取特征,然后根据特征进行相似性计算,目前主要采用词频特征,如:计算每个词的 TF/IDF 特征[6],然后使用 LSA[7]、余弦相似度等方法计算文本间相似度. 基于词匹配的相似性度量方法一般通过计算两个文本之间的距离实现,如:Hamming 距离、Levenshtein 距离、Jaccard 指数等. 由于疾病诊断文本长度比较短,通常还会出现简化缩略或者顺序错位等问题,因此这些计算方法计算效果并不理想. 依据疾病诊断文本特点,本文提出了一种新的基于集合的相似度(set based similarity,以下简称 SBS)计算方法.

假设两个短文本 s_1 和 s_2,其中,$s_1 = \{w_{11}, w_{12}, \cdots, w_{1p}\}$,$s_2 = \{w_{21}, w_{22}, \cdots, w_{2q}\}$,$w$ 是疾病短文本包含的字,那么 s_1 与 s_2 之间的相似度 $d(s_1, s_2)$ 定义如式(1)所示.

$$d(s_1,s_2)=\begin{cases}0, & \text{if } |\mathrm{set}(s_1)|\cdot|\mathrm{set}(s_2)|=0;\\ \dfrac{|\mathrm{set}(s_1)\cap\mathrm{set}(s_2)|}{|\mathrm{set}(s_1)\cup\mathrm{set}(s_2)|}, & \text{其他};\\ 1, & \text{if } \mathrm{set}(s_1)\subseteq\mathrm{set}(s_2)\\ & \text{或 } \mathrm{set}(s_2)\subseteq\mathrm{set}(s_1).\end{cases} \tag{1}$$

其中:set(s)表示获取文本 s 中的字集合;|set(s)|表示集合 set(s)的大小.

表 1 显示了 SBS(S)和 Hamming(H),Levenshtein(L),Jaccard(J)不同计算方法的对比. 其中,Hamming 距离要求输入的两个文本长度必须相等,Levenshtein 距离在大数据集上存在计算性能低的问题,Jaccard 距离没有考虑两种特殊的情况,而 SBS 方法解决了上述问题,计算效果较好.

表 1 几种相似度计算方法对比

Table 1 The comparison of different similarity computings

同义短文本对	H	L	J	S
“双眼糖网”和“双眼糖尿病视网膜病变”	—	0.5	0.29	1.0
“双眼视网膜动脉硬化”和“双眼动脉硬化性视网膜病变”	—	0.42	0.75	1.0
“双眼动脉硬化性视网膜病变”和“双眼动脉硬化性视网膜变化”	0.83	0.83	0.85	0.85

2 基于相似度分布的自适应聚类

基于上述短文本相似度量方法,本文聚类算法首先进行同义短文本的识别,然后针对同义文本类簇提取标准的疾病命名.

针对短文本集合 D,计算集合中所有文本与其他文本之间的相似度,并按照从大到小顺序排列,如图 1 所示. 每一个文本对象的相似距离序列表明其他文本和该文本的相似程度. 因此,确定该文本对象的同义文本类簇,只需要获取该相似度曲线的截断阈值,所有大于该阈值的文本对象作为该文本对象的初始同义词类簇,并将这些文本对象从集合中移除. 针对移除后新的文本集合进行同样的操作,直至该集合为空结束.

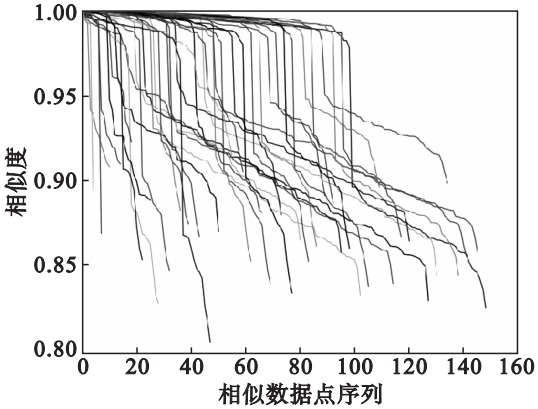

图 1 相似度序列曲线

Fig. 1 Curves of similarity sequence

上述过程定义为 SplittingCluster 子算法. 假设一个类簇 c 用元组结构(header, members)表示,输入数据集为 D,两个文本 s 和 s'之间的相似度通过 SBS 获取,记作 SBS(s,s'). SplittingCluster 定义如下.

算法 1　SplittingCluster(D,SBS)

输入：　D 为数据集,SBS 为相似性计算过程

输出：　类簇集合 $C=\{c_1,\cdots,c_m\}$,m 为类簇个数

步骤：

1)　初始化集合 $C=\{\}$

2)　计算 D 中每个数据点的相似度序列,对 D 中数据点排序

3)　While 集合 D 不为空 do：

4)　　从 D 中顺序选择一个数据对象 s

5)　　计算 s 与 D 中所有数据对象 s'之间相似度 SBS(s,s'),得到一个相似度列表 L

6)　　按照降序将 L 进行排序,计算该相似序列截断阈值 $T_{\text{sim}}=\text{CT}(L)$

7)　　获取类簇 $c=(s,\{s' \mid \text{SBS}(s,s')<T_{\text{sim}}\})$,将 c 加入集合 C

8)　　将包含在集合$\{s' \mid \text{SBS}(s,s')<T_{\text{sim}}\}$中对象从集合 D 中删除

9)　返回类别集合 C

SplittingCluster 算法中 T_{sim}表示计算相似度序列的截断阈值. 依据图 1 中相似度序列曲线特点,CT(L)的计算过程就是计算离散序列的第一个二阶导数为 0 的点对应的相似度值. 为提高算法的运行效率,本文利用领域先验知识,按照解剖部位/组织对数据集进行类别预划分,将大数据集划分为若干个相对较小的数据集,提高聚类准确率和效率.

容易证明上述 SplittingCluster 子算法复杂度在$[O(n\lg 2n),O(n^3)]$区间变动,依赖于数据个数 n.

证明　在最坏的情况下,假设最终的数据集每个类簇的成员个数为 1,那么总计算次数为 $n(n+1)/2+n(n+1)(2n+1)/6\sim O(n^3)$;最好的情况下,假设最终数据集只有一个类簇,即类簇成员为 n,那么只需要计算 n 次,理想情况下排序时间复杂度为 $O(n\lg 2n)$.

3　基于序列模式的中心概念抽取

运行上述 SplittingCluster 子算法,数据集 D 被分成了若干个类簇,类簇集合 C 表示为 $C=\{c_1,\cdots,c_m\}$. 但这个结果存在两个问题:①类簇包含了同义文本,但每一个类簇缺乏标准疾病命名. ②类簇之间可能具有相同的标准疾病命名,需要合并类簇. 为了解决该问题,这里给出中心概念的定义.

定义　给定一个同义文本集合 S,从集合 S 中提取新的文本 c,如果 c 和集合 S 中的语义相同,那么 c 称为集合 S 的中心概念. 如:集合 $S_1=\{$“双眼糖网”,“左眼糖尿病视网膜病变”,“双眼糖尿病视网膜病变”$\}$,从 S_1 中提取的新文本 $c_1=$“糖尿病视网膜病变”,那么 c_1 就是集合 S_1 的中心概念.

因此,通过提取每个类簇的中心概念,并根据中心概念进行类簇合并,即可解决 SplittingCluster 算法运行结果存在的两个问题. 本文采用基于序列模式挖掘方法获取每一个类簇的中心概念,即:将类簇中每一个短文本看作字序列,提取类簇中最长子序列. 表 2 中文本集合挖掘出的最长子序列为“眼糖尿病视网膜病变”,作为表中文本集合的中心概念. 序列模式挖掘有相对比较成熟的方法,本文采用基于 PrefixSpan 的方法提取最长频繁子序列[8].

表 2　基于频繁序列的中心概念提取

Table 2　Frequent sequence based central concept extraction

ID	文本序列集合	频繁序列
1	[[右],[眼],[糖],[尿],[病],[视],[网],[膜],[病],[变]]	眼糖尿病视网膜病变
2	[[左],[眼],[糖],[尿],[病],[视],[网],[膜],[病],[变]]	眼尿病视网膜病变
3	[[双],[眼],[糖],[尿],[病],[视],[网],[膜],[病]]	眼糖病视网膜病变
4	[[双],[眼],[糖],[网]]	眼糖尿病网膜病变
5	[[双],[眼],[糖],[尿],[病],[视],[网],[膜],[病],[变],[(],[V],[期],[)]]	眼糖尿病视膜病变
6	[[双],[眼],[增],[殖],[性],[糖],[尿],[病],[视],[网],[膜],[病],[变]]	眼糖尿病视网病变 ……

基于中心概念定义和获取方法,对集合 C 中各个类簇集合提取中心概念后,将类簇和类簇成员逐一迭代且合并对应成员,并将空类簇删除,直到类簇的个数不再变化,达到稳定的状态. 由此可以得到类簇合并算法 MergingCluster,定义如下.

算法 2 MergingCluster(C)

输入: C 为集合 D 对应的类簇集合

输出: 合并后的类簇集合 C'

步骤:

1) Do{

$m = |C|$

2) 遍历类簇集合 C 中每一个类簇 c{

3) 遍历类簇 c 中每一个成员 s{

4) 如果 s 与类簇 c'有最大相似度($c' \neq c$),则将 s 从 c 中移除,加入到 c'中}}

5) 删除 C 中成员为空的类簇

6) 重新计算 C 中每一个类簇的中心概念

7) }Until 集合 C 中类簇个数 m 稳定

8) 返回 $C' = C$

针对表 2 所示的例子数据,运行 SplittingCluster 算法得到如图 2a 所示结果,运行 MergingCluster 算法得到如图 2b 所示结果.

MergingCluster 子算法的最高计算复杂度为 $O(c^2n)$,n 为数据集大小,c 为类簇个数. 证明:最坏情况下,假设有 n 个类簇,即每个类簇只有一个成员,c 等于 n,那么需要计算 $c[1\times(c-1)+\cdots+1\times(c-1)]=c[n(c-1)]\sim O(c^2n)$;最好情况就是只有一个类簇,不需要合并和移动类簇成员.

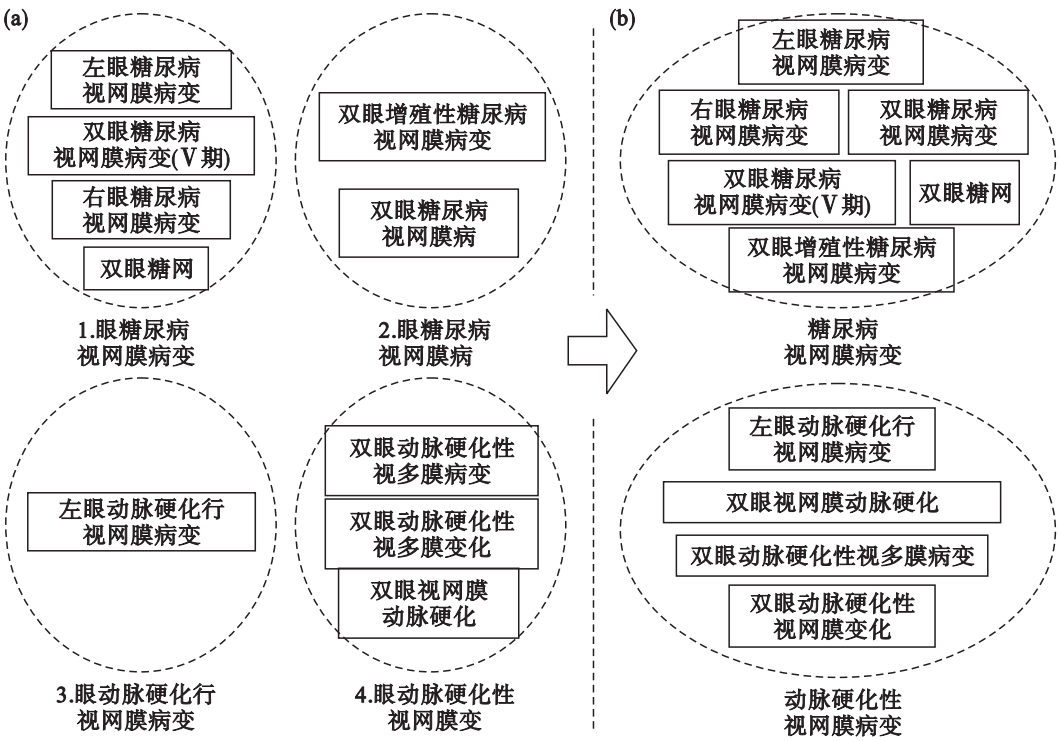

图 2 ACT 运行结果

Fig. 2 ACT running results

(a)—运行 SplittingCluster 算法;(b)—运行 MergingCluster 算法.

ACT 算法与 K-means 等传统聚类算法不同点在于:①ACT 算法不需要一个预先设定的类簇个数,而传统聚类算法则需要预先设定;②ACT 算法通过获取文本集合的中心概念作为该类簇的"中心点",而传统聚类算法是通过计算距离计算数据的逻辑中心点/质心;③ACT 算法的类簇在合并过程会逐步减少,而传统聚类算法类簇个数不变.

4 实验与分析

本文使用的测试数据集来自中国医科大学第二附属医院. 电子病历数据经过匿名处理后,从中提取出疾病诊断文本,并以此作为研究的短文本数据集. 数据集中文本总数 52 957,平均每个病历包含 3 个疾病文本,平均文本长度为 6,覆盖 62 个科室. 该数据集包含了常见的临床疾病诊断,涵盖的疾病集合是 ICD-10 包含的医学领域疾病的子集.

ACT 算法使用 Python v2.7 实现,实验在台式机进行,配置为 Intel Core 2,CPU 2.80 GHz,4 GB内存和 Windows7 操作系统. 表 3 列出了数据集中数据最多的 33 个数据子集运行的类簇个数和运行时间. 通过将运行结果和临床标准进行对比评估,平均聚类准确率为 96.4% .

表 3　自适应聚类算法测试结果
Table 3　Testing results of ACT algorithm

ID	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
数据集	眼	骨	肠	血	腺	子宫	心	肺	肝	肾	脑	腹	皮	胆	动脉	胃	颈
文本数	201	188	187	156	155	141	137	135	119	114	112	104	96	84	81	68	65
类簇数	77	114	106	104	85	42	84	53	44	60	57	60	59	48	48	36	49
时间/s	4.02	3.46	2.81	2.36	1.89	0.10	2.06	1.56	0.83	0.88	0.77	0.55	0.94	0.36	0.36	0.30	0.02

ID	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33
数据集	神经	细胞	静脉	发	胰	胸	鼻	肢	脊	肌	颅	耳	淋巴	尿	食管	口
文本数	61	58	47	43	42	40	38	38	37	36	35	34	34	33	32	32
类簇数	31	44	34	19	27	25	23	23	24	16	26	20	28	19	26	20
时间/s	0.23	0.25	0.10	0.10	0.04	0.04	0.04	0.04	0.04	0.04	0.15	0.03	0.04	0.02	0.02	0.03

5　结　　论

本文针对疾病文本数据特点，提出了一种新的自适应的文本聚类算法 ACT. 该算法采用了一种基于集合的文本相似性度量方法，弥补了已有相似度计算方法的缺陷；同时能够自动确定聚类簇个数，不需要人工交互设置先验数值；算法采用基于序列模式的中心概念提取方法实现了聚类簇的合并和优化，进一步提升了聚类准确性.

本文使用医院病历疾病诊断文本数据测试发现，ACT 聚类算法平均聚类准确率为 96.4%，运行效率高，为病历数据预处理、分类等应用提供了标准的疾病分类标准.

参考文献：

[1] Jensen P B, Jensen L J, Brunak S. Mining electronic health records: towards better research applications and clinical care [J]. *Nature Reviews Genetics*, 2012, 13(6): 395 – 405.

[2] Utter G H, Cox G L, Owens P L, et al. Challenges and opportunities with ICD – 10 – CM/PCS: implications for surgical research involving administrative data [J]. *Journal of the American College of Surgeons*, 2013, 217(3): 516 – 526.

[3] DeAlmeida D R, Watzlaf V J, Firouzan M, et al. Evaluation of inpatient clinical documentation readiness for ICD – 10 – CM [EB/OL]. [2014 – 07 – 08]. http://perspectives. ahima. org/evaluation-of-inpatient-clinical-documentation-readiness-for-icd-10 – cm/#. VAQvAU3loeE.

[4] Ansari S, Chetlur S, Prabhu S, et al. An overview of clustering analysis techniques used in data mining [J]. *International Journal of Emerging Technology and Advanced Engineering*, 2013, 3(12): 284 – 286.

[5] Poelmans J, Ignatov D I, Kuznetsov S O, et al. Formal concept analysis in knowledge processing: a survey on applications [J]. *Expert Systems with Applications*, 2013, 40(16): 6538 – 6560.

[6] Zhang W, Yoshida T, Tang X. A comparative study of TF/IDF, LSI and multi-words for text classification [J]. *Expert Systems with Applications*, 2011, 38(3): 2758 – 2765.

[7] Nakov P, Popova A, Mateev P. Weight functions impact on LSA performance [C]// Recent Advances in Natural Language Processing. Granada, 2001: 187 – 193.

[8] Pei J, Han J, Mortazavi-Asl B, et al. Prefixspan: mining sequential patterns efficiently by prefix-projected pattern growth[C]// IEEE the 29th International Conference on Data Engineering (ICDE). Brisbane, 2001: 215 – 224.