doi:10.3969/j.issn.1672 - 4348.2016.01.018

K-Means 聚类的多种距离计算方法的文本实验比较

林滨

(福州软件职业技术学院 计算机系, 福建 福州 350003)

摘要: 针对文本类型数据的分类进行研究,用 VSM 模型和 TF-IDF 技术对文本文件进行了数据样本抽取加权,得到文本相似度矩阵;采用不同样本距离计算方法和 K-Means 算法对数据进行了聚类实验,获得聚类结果并进行了分析和总结;基于实验结论,研究了不同距离计算方法之间的区别以及适用的数据类型。

关键词: 文本聚类; TF-IDF; K-Means; 距离计算

中图分类号: TP311.13　　**文献标志码**: A　　**文章编号**: 1672 - 4348(2016)01 - 0080 - 06

Experimental comparison of K-Means text clustering by varied distance calculation methods

Lin Bin

(Fuzhou Software Technology Vocational College, Fuzhou 350003, China)

Abstract: Text data samples were extracted and weighted and the text similarity matrices were obtained by vector space model (VSM) model and TF - IDF weighting technology. The data clustering was conducted via different distance calculation methods and K-Means algorithm. The clustering results were analysed. The differences among the distance calculation methods and the applicable data types were studied.

Keywords: text clustering; TF - IDF; K-Means; distance calculation

聚类是无监督机器学习算法的一个重要组成部分。聚类(clustering)是将物理或抽象对象的集合分成由类似的对象组成的多个类的过程[1],使得每一个类内的数据尽可能相似而不同组内的数据尽可能不同。对样本数据集进行分类计算后形成多个簇(cluster),同属于一个簇中的数据具有比较高的相似度,而属于不同簇的对象之间差异度相对较高,簇的分类通常是根据数据样本间的距离来进行区分。

文本聚类在信息检索等诸多方面有较广泛的应用,对文本文档进行聚类可以采用 K-Means 算法。这种聚类方法的特点是聚类过程的无指导性,无需事先对数据样本进行标记设定,属于典型的无监督(unsupervised)机器学习问题。搜索引擎返回的的结果进行信息分类,电子商务网站的用户偏好商品推荐,社交网络的用户分类采用的也是文本聚类方法。此外,文本聚类方法在文档自动归类整理,图书馆资料分类等方面也起着重要的作用。

文中先从整体上对数据样本的处理流程进行介绍;接着对文本数据的预处理进行介绍;然后对样本间距计算方法和 K-Means 算法进行了研究;之后,使用 MATLAB 数学软件,以不同的间距计算方法,对转换后的 Reuters-21578 新闻数据集进行 K-Means 聚类实验,并对聚类结果进行分析和总结;最后,在不同距离计算方法之间的区别,以

收稿日期: 2016 - 01 - 25

作者简介: 林滨(1978 -),男,福建福州人,助教,硕士,研究方向:机器学习、大数据。

及适用的数据类型方面进行了研究。

1　文本数据预处理

1.1　文本数据相似性计算

计算文本数据的相似性的目的在于为文本聚类提供依据，相似的文本数据被归为同一类。因此，需要寻找一个可以用来计算两个数据样本相似性的函数。具体应用到文本文档，两篇文档中相同词的个数，就成为判断是否相似的关键。所以，通过收集各个文档中每个单词的使用频率，然后计算文档的词汇交集中的用词频率是否接近，从而用来判断这些文档的相似度。基于这种思想，可以计算出文本数据的相似性，以便用来聚类。

1.2　文档样本向量化

为了提取收集到的文档样本的特征，并将其转换成计算机可以处理的数据，需要将包括词频在内的这些特征进行数字化成向量（vector）。这种将特征数字化为向量的过程一般称为向量化（vectorization）。

判断一个句子是否与问题相关可以通过计算该句子与问题的相似度来完成，信息检索中常用的相似度计算方法是向量空间模型（vector space model，VSM）[2]。列出所有的需要向量化的文档可能遇到的单词的集合，每一个词被赋予一个数字，该数字就代表其在向量词典中所占的维度。有些文档的维度很大，最大值可以达到向量的基数（cardinality），故文档通常被假定有未定义的维度。

文档向量中的元素值被设定为每个单词在文档中出现的次数，同时该值会按照顺序保存在各个单词所属于的维度。这个数值一般称为（term frequency，TF）权重。

我们一般通过计算文本文档中所有单词权重的距离来进行聚类，从而来判断两个文档的相似性。可是影响距离计算结果的词，往往是一些使用最频繁的词，比如 a、that、the、it、is、are 等，这些词被称作停用词。不管选择使用什么距离计算方法，聚类结果进行总是会被这些词影响。为此，我们必须修改这些词的权重，通过采用新的加权方法来降低影响。

1.3　TF－IDF 加权方法

根据向量空间模型，单词在伴随文本中的重要性用该单词的 tf－idf 值来度量[3]。改进前文档向量保存的是词频，改进后保存的是词频乘以文率的倒数，称为逆文档频率（inverse document frequenty，IDF）。这意味着，词的重要程度不仅受益于它在文档中出现的次数，而且同时导致它在语料库中出现的频率成反比下降。

词频（TF）是单词在文档中的次数除以该文档的总单词数。

文档频率（document frequency，DF）是某个单词出现在特定语料库中文档的个数统计。假设单词 W_i 在文档中出现的频率表示为 DFi，则逆文档频率 IDFi 便为：

$$\mathrm{IDF}_i = 1/\mathrm{DF}_i \tag{1}$$

如果包含该词的文档数量很多，则该词 IDF 值会变得太小，文档向量中该词相应的权重值也随之变得太小。解决方法可以是乘一个常数，用文档个数（N）来增大 IDF 值。则 IDF 公式可改为：

$$\mathrm{IDF}_i = N/\mathrm{DF}_i \tag{2}$$

那么，文档向量中单词权重 W_i 为：

$$W_i = \mathrm{TF}_i * \mathrm{IDF}_i = \mathrm{TF}_i * N/\mathrm{DF}_i \tag{3}$$

上述公式 IDF 值作用仍不够理想，不能突显单词词频（TF）在权重中的重要程度。为此，一般要对 IDF 值取对数：

$$\mathrm{IDF}_i = \log(N/\mathrm{DF}_i) \tag{4}$$

由上述 4 个公式可总结出对于 TF－IDF 方法中计算单词 w_i 权重 W_i 公式为：

$$W_i = \mathrm{TF}_i * \log(N/\mathrm{DF}_i) \tag{5}$$

因此，在文档矩阵中，给予某个单词 w_i，该词在对应矩阵中存放的便是上述权重。这样的计算导致停用词权重被降低，罕见词的权重被放大。一般来说，当某词有很大的 TF 值和比较大 IDF 值时，便会被认为是文档的重要单词或者关键词。

2　K-Means 算法

2.1　算法思想

K-Means 算法是一种通用的聚类算法[4]，很多场合都用它来做文本聚类。K-Means 算法的基本思想是使用一种迭代机制进行分类，其具体算法过程如下：

假设有 n 个向量，需要划分成 k 个簇。随着指定 k 个值作为簇中心，然后进行多次迭代来重新计算中心，达到最大迭代值，或者中心收敛在固定位置后停止。

每一次迭代过程分为两步操作。首先,找到距离中心最近的向量,并将这些向量划归给相应的簇。然后,计算用各簇中所有向量的坐标平均值用来更新中心。

算法过程如下:

(1)从 n 个文档随机选取 k 个单词作为簇中心;

(2)对剩下的每一个单词计算它到每个中心的距离,把其划归为最接近的中心的簇;

(3)再次计算已获得的各个簇单词的中心;

(4)迭代 2 ~3 步直到新获得的中心和原中心重合或低于设定阈值,整个算法结束。

2.2 距离计算方法

在文本聚类的具体应用中,恰当选择特定的距离计算的方法是决定 K-means 文本聚类整体质量的关键性因素。距离最远的样本点最不可能分到同一个簇中[5]。K-means 中一般采用下面几种距离计算方法来描述数据个体之间的相似度。

2.2.1 欧氏距离

欧氏距离(Euclidean distance)法是在这 4 种距离计算方法中最简单直观的一种方法,符合我们对距离感性的认识。在数学中,两个 n 维向量 $(\boldsymbol{s}_1,\boldsymbol{s}_2,\boldsymbol{s}_3,\cdots,\boldsymbol{s}_n)$ 和 $(\boldsymbol{t}_1,\boldsymbol{t}_2,\boldsymbol{t}_3,\cdots,\boldsymbol{t}_n)$ 之间的距离表示为:

$$d_{st}^2 = \sqrt{(x_s - x_t)(x_s - x_t)'} \tag{6}$$

2.2.2 曼哈顿距离

曼哈顿距离(Manhattan distance)为标准坐标系上的绝对轴距总和,结果是两个坐标差的绝对值之和,是一种城市区块距离。两个 n 维向量 $(\boldsymbol{s}_1,\boldsymbol{s}_2,\boldsymbol{s}_3,\cdots,\boldsymbol{s}_n)$ 和 $(\boldsymbol{t}_1,\boldsymbol{t}_2,\boldsymbol{t}_3,\cdots,\boldsymbol{t}_n)$ 之间的距离应表示为:

$$d_{st} = \sum_1 |x_{sj} - x_{tj}| \tag{7}$$

2.2.3 夹角余弦距离

夹角余弦距离(cosine distance, CD)法可以通过从原点出发,并指向两个位置坐标所产生的两条向量间的夹角来测距。当夹角值较小时,两个向量的指向大致一样,便可以认为这两个位置坐标比较相似。夹角余弦越小,两向量的夹角越大。因此,用 1 减去余弦值来得到一个有效的距离来作为余弦距离。

两个 n 维向量 $(\boldsymbol{s}_1,\boldsymbol{s}_2,\boldsymbol{s}_3,\cdots,\boldsymbol{s}_n)$ 和 $(\boldsymbol{t}_1,\boldsymbol{t}_2,\boldsymbol{t}_3,\cdots,\boldsymbol{t}_n)$ 之间的余弦距离计算公式可以表示为:

$$d_{st} = 1 - \frac{x_s x_t'}{\sqrt{(x_s x_s') - (x_t x_t')}} \tag{8}$$

2.2.4 相关距离

相关距离(correlation distance)中采用的相关系数,是衡量给定随机变量 X 与 Y 相关性的一种方法,取值范围是[-1,1]。绝对值越大,表明 X 与 Y 相关度越高。X 与 Y 线性相关时,相关系数取值为 1 或 -1。

两个 n 维向量 $(\boldsymbol{s}_1,\boldsymbol{s}_2,\boldsymbol{s}_3,\cdots,\boldsymbol{s}_n)$ 和 $(\boldsymbol{t}_1,\boldsymbol{t}_2,\boldsymbol{t}_3,\cdots,\boldsymbol{t}_n)$ 之间的相关距离计算公式可以表示为:

$$d_{st} = 1 - \frac{(x_s x_s')(x_s x_s')}{\sqrt{(x_s x_s') - (x_t x_t')}\sqrt{(x_s x_s') - (x_t x_t')}} \tag{9}$$

3 实验与分析

3.1 实验介绍

为研究不同距离计算法在 K-Means 聚类中的表现,实验分别使用欧式距离、曼哈顿距离、夹角余弦距离和相关距离,运用这些不同的距离计算法,对 Reuters-21578 新闻数据集进行 K-Means 聚类实验,再从实验数据中比较、分析聚类效果。

Reuters-21578 数据集包含有 22 个 SDML 文件,文件的内容为纯文本,每一个文件包含有新闻稿件。

实验环境为 Windows 系统, MATLAB 2015b 环境,Eclipse 编程工具。

系统流程如图 1 所示。

在 MATLAB 计算环境下,4 个 K-means 聚类函数的入口均为:

[IDX,C,sumd,D] = kmeans(matrix, cluster, distance, emptyaction, onlinephase, options, replicates, start);

其中,参数 distance 为不同距离计算方法。

本实验在数据集 135 种类别的文章中,重点关注 5 个方面的文章:就业(jobs)、粮食(grain)、外贸(trade)、并购(acquisition)和利率(interest rate)。分别采用不同的距离计算公式分别对数据集进行了多次 K-means 聚类,并显示簇中每一类文章个数和聚类准确率。

3.2 实验结果

3.2.1 使用欧式距离的平方计算

[IDX, C, sumd, D] = kmeans(R, 7, 'Distance', 'sqEuclidean', 'Replicates', 3, 'Op-

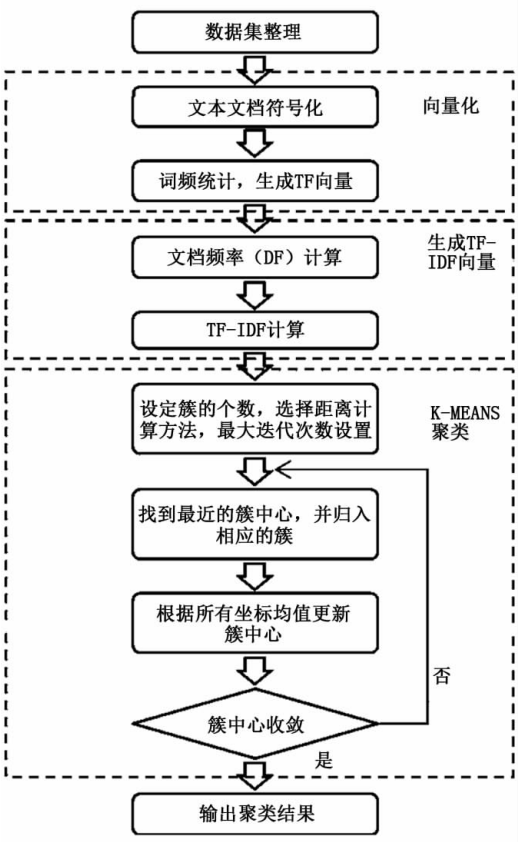

图 1　系统流程图

Fig. 1　Text data clustering system flowchart

tions', opts)。

聚类结果如表 1 所示。

表 1　欧式距离聚类结果

Tab. 1　Euclidean type distance clustering result

聚类组别	文章数/篇	聚类准确率/%
1 组	29	29
2 组	6	11
3 组	43	8
4 组	21	42
5 组	25	21
6 组	111	31
7 组	2	1

3.2.2　使用曼哈顿距离计算

[IDX, C, sumd, D] = kmeans(R, 7, 'Distance', 'cityblock', 'Replicates', 5, 'Options', opts)。

聚类结果如表 2 所示。

表 2　曼哈顿距离聚类结果

Tab. 2　Mahattan type distance clustering result

聚类组别	文章数/篇	聚类准确率/%
1 组	20	39
2 组	5	11
3 组	2	1
4 组	15	31
5 组	1	1
6 组	2	2
7 组	190	23

3.2.3　使用夹角余弦距离计算

[IDX, C, sumd, D] = kmeans(R, 7, 'Distance', 'cosine', 'Replicates', 3, 'Options', opts)。

聚类结果如表 3 所示。

表 3　夹角余弦距离聚类结果

Tab. 3　Cosine distance clustering result

聚类组别	文章数/篇	聚类准确率/%
1 组	21	42
2 组	42	68
3 组	32	63
4 组	37	72
5 组	46	61
6 组	29	11
7 组	28	7

3.2.4　使用相关距离计算

[IDX, C, sumd, D] = kmeans(R, 7, 'Distance', 'correlation', 'Replicates', 5, 'Options', opts)。

聚类结果如表 4 所示。

表 4　相关距离聚类

Tab. 4　Related distance clustering (result)

聚类组别	文章数/篇	聚类准确率/ %
1 组	31	6
2 组	22	46
3 组	42	8
4 组	38	71
5 组	26	52
6 组	33	68
7 组	43	70

3.3　结果分析

上述 1 ~ 7 组聚类相关的结果，是采用使用不

同距离计算方法作为相似性度量的 K-Means 聚类的文章数和准确率[6-7]。

比较发现,采用欧式距离平方计算方法和 Manhattan 距离计算方法的结果中,本应属于不同组的大量文章被划分成同一组,如欧式距离平方算出的第 6 组聚类以及 Manhattan 距离计算出的第 7 组聚类,聚类质量不高;而在夹角余弦距离和相关距离计算方法的聚类结果中,各个聚类结果与实际文章分类差别不大。由图 2 可以看出,明显后两者在本次文本聚类实验中的表现胜过前面两种距离计算方法。

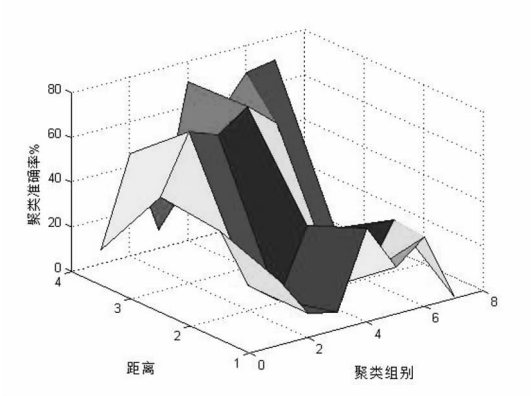

图 2 不同距离计算方法聚类准确率比较

Fig. 2 Comparison of clustering accuracy among different distance calculation methods

以上 4 种距离计算方法从数学上分析主要分为两类,一类体现的是具体位置坐标上的差异,另一类体现的是方向上的差异。这里选取有代表性的欧氏距离和夹角余弦距离,并把维度降低到三维来进行讨论。由图 3 可以看出,欧氏距离计算的是三维空间各个点的绝对距离,这与各个点具体所在的三个位置坐标直接相关;而夹角余弦距离计算的是空间向量间的夹角值,体现的是方向上的差异,而不是具体的位置坐标的差异。如果保持 A 点坐标位置不变,将 B 点朝远离坐标轴原点的方向移动,这时夹角余弦距离是保持不变的,这是因为夹角没有发生改变,但是 A、B 两点间的坐标距离显然发生了改变,这就是欧氏距离和夹角余弦距离计算时的不同之处。对于文本类型的数据的相似性分析,要从方向上区分差异,适当降低对绝对数值的敏感度,因而可以选择夹角余弦距离作为计算方法。对于其它要从维度的数值大小中体现差异的数据分析,欧氏距离计算方法更能体现出数值特征的绝对差异。

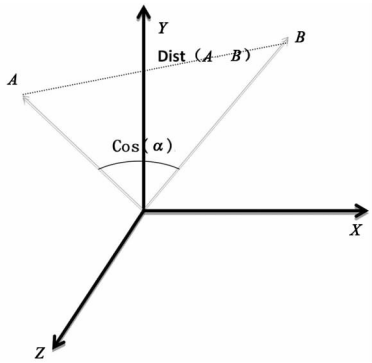

图 3 欧氏距离和夹角余弦距离比较

Fig. 3 Comparison of distance between the results gained by Euclidean method and by cosine method

算法、距离计算方法和数据类型都可能在一定程度上影响文本聚类的结果[8-9]。对于 K-means 算法,聚类质量与初始中心的选择以及提供的聚类的数目有很大关系。通过多次聚类取最佳结果这种方法可以减少聚类质量对初始中心选择的依赖。文本类的数据集,通过改进向量空间模型特征加权和采用更合适的距离计算方法,可在一定程度上改善聚类的输出质量。

经过上述比较后可以得出初步结论,夹角余弦距离和相关距离计算方法,更适合用作文本数据的相似性计算[10]。

4 结论

距离计算方法的选择对聚类形状会产生显著的影响。不同样本之所以会聚成一类,是因为它们的类内变差要小于类间变差。选择合适的距离计算方法可以在一定程度上减小类内变差或者增大类间变差,使得大部分相似文档的类内变差小于类间变差。本文对由文本数据元素组成的 Reuters-21578 文本分类标准数据集,分别采用欧式距离、曼哈顿距离、夹角余弦距离和相关距离计算方法共进行了多次 K-Means 文本聚类尝试。根据实验结果的分析、比较和归纳得出实验结论,夹角余弦距离和相关距离更适合用于文本数据聚类的相似性计算。最后,分析了有助于提高聚类输出质量的相关重要因素以及对应的解决办法。总之,针对不同的数据集和不同的聚类要求,应该从多个方面来考虑,以便寻求与之适合的距离计算方法。

参考文献：

[1] 孙吉贵,刘杰,赵连宇. 聚类算法研究[J]. 软件学报,2008,19(1):48 - 61.

[2] 余正涛,樊孝忠,郭剑毅,等. 基于潜在语义分析的汉语问答系统答案提取[J]. 计算机学报,2006,29(10):1889 - 1893.

[3] 吴飞,韩亚洪,庄越挺,等. 图像 - 文本相关性挖掘的 Web 图像聚类方法[J]. 软件学报,2010,21(7):1561 - 1575.

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

[5] 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 - 719.

[6] Jain A K. Data clustering: 50 years beyond k-Means[J]. Pattern Recognition Letters,2010,31(8):651 - 666.

[7] Song Q B, Ni J J, Wang G T. A fast clustering-based feature subset selection algorithm for high-dimensional data[J]. IEEE Trans on Knowledge and Data Engineering,2013,25(1):1 - 14.

[8] Aldahdooh R T, Ashour W. Distance-based initialization method for K-means clustering algorithm[J]. International Journal of Intelligent Systems and Applications,2013,5(2):41 - 51.

[9] 李法运,农罗锋. 基于向量语义相似度的改进 K-Means 算法[J]. 情报科学,2013,31(2),34 - 37.

[10] 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, 2011,34(5):28 - 37.

(责任编辑：肖锡湘)

(上接第 79 页)

可以仍然处于 Wait 状态,从而形成循环。

由此可知,Fischer 协议无法避免进程饥饿现象,对于单个申请访问临界区的进程,不能保证其一定能够最终获取临界区资源。

4　结语

形式化方法是用于保障系统安全性与可靠性的重要手段,也是当前系统分析研究领域的前沿与热点,并在多类不同系统的分析中得到了成功运用。实时互斥协议作为一种时间敏感的系统协议,采用具有严格数学语义的形式方法进行性质分析,是十分有效且必要的。使用时间自动机构建了 Fischer 协议的形式化模型,并通过模型检测技术验证其满足互斥与无死锁两个重要性质,但同时也验证出其不满足进程活性。由此表明,形式化技术尤其是模型检测技术是一种有效的协议分析技术,能够成为保障各类复杂协议正确性及可靠性的有力手段。

参考文献：

[1] Dijkstra E W. Solution of a problem in concurrent programming control[J]. Communications of the ACM,1965,8(1):289 - 294.

[2] Hesselink W H. Mutual exclusion by four shared bits with not more than quadratic complexity[J]. Science of Computer Programming,2015,102(5):57 - 75.

[3] Lynch N, Shavit N. Timing-based mutual exclusion[C]//Proc of IEEE Real-Time Systems Symposium. Phoenix, America: IEEE Computer Society,2002:2 - 11.

[4] Machin M, Dufosse F, Blanquart J P, et al. Specifying safety monitors for autonomous systems using model-checking[J]. Lecture Notes in Computer Science,2014, 8666:262 - 277.

[5] Behrmann G, David A, Larsen K G. Formal methods for the design of real-time systems[M]. Berlin: Springer-Verlag, 2004:200 - 236.

[6] Lamport L. A fast mutual exclusion algorithm[J]. Acm Transactions on Computer Systems Tocs Homepage,1987,5(1):1 - 11.

[7] Beatrice B. An introduction to timed automata[J]. Lecture Notes in Control and Information Science,2013,433:169 - 187.

[8] Behrmann G, David A, Larsen K G, et al. Developing UPPAAL over 15 years[J]. Software: Practice and Experience, 2011,41(2):133 - 142.

[9] Behrmann G, David A, Larsen K G, et al. UPPAAL 4.0[J]. Quantitative Evaluation of Systems,2006,4(12):125 - 126.

(责任编辑：肖锡湘)