软件学报 ISSN 1000-9825, CODEN RUXUEW
Journal of Software,2011,22(8):1761−1770 [doi: 10.3724/SP.J.1001.2011.03893]

E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel/Fax: +86-10-62562563

基于流信息距离的多文本流热点挖掘*

杨 宁+, 唐常杰, 王 悦, 陈 瑜, 郑皎凌, 李红军

(四川大学 计算机学院,四川 成都 610065)

Mining Hotspots from Multiple Text Streams Based on Stream Information Distance

YANG Ning+, TANG Chang-Jie, WANG Yue, CHEN Yu, ZHENG Jiao-Ling, LI Hong-Jun

(College of Computer Science, Sichuan University, Chengdu 610065, China)

+ Corresponding author: E-mail: yneversky@gmail.com

Yang N, Tang CJ, Wang Y, Chen Y, Zheng JL, Li HJ. Mining hotspots from multiple text streams based on stream information distance. *Journal of Software*, 2011,22(8):1761−1770. http://www.jos.org.cn/1000-9825/3893.htm

Abstract: This paper characterizes the local and global hotspots in text streams and elaborates their correlation. The paper then applies Kolmogorov complexity to mining the hotspots in multiple text streams. The Redundant Information is defined based on Kolmogorov complexity, and it has been demonstrated that the Redundant Information exceeding a threshold is necessary for the local hotspots. Secondly, a similarity metric, termed as Stream Information Distance (SID), is suggested based on the conditional Kolmogorov complexity to quantify the similarity between different text streams. Borrowing ideas of Phylogeny originated from Computational Biology, a heuristic algorithm based on hierarchical clustering is proposed to mine the global hostspots from multiple text streams. Finally, the convergency, effectiveness, and scalability of this algorithm are validated by the extensive experiments over synthetic and real data set.

Key words: hotspot mining; multiple text streams; stream information distance; redundant information; Kolmogorov complexity

摘 要: 把文本流中的热点区分为局部热点和全局热点,分析了二者的相关性,并将 Kolmogorov 复杂度应用于多文本流中的热点挖掘.首先,定义了基于 Kolmogorov 复杂度的冗余信息的概念,并论证了文本流存在局部热点的必要条件是冗余信息超过某个阈值;其次,基于条件 Kolmogorov 复杂度提出了一个相似性度量指标——流信息距离(stream information distance,简称 SID),以衡量不同文本流之间的相似度;并借鉴计算生物学领域中的种系发生树的思想,提出了一种基于层次聚类的多文本流全局热点挖掘启发式算法.在合成和真实数据集的实验,验证了算法的收敛性、有效性和规模可伸缩性.

关键词: 热点挖掘;多文本流;流信息距离;冗余信息;Kolmogorov 复杂度

中图法分类号: TP311 **文献标识码**: A

近年来,从文本流中挖掘热点(有些文献称为突发模式),在信息检索、舆情分析、科技情报分析等应用中,

* 基金项目: 国家自然科学基金(600773169); 国家科技支撑计划(2006BAI05A01)
收稿时间: 2009-10-12; 修改时间: 2010-03-29; 定稿时间: 2010-06-10

扮演着越来越重要的角色[1−3].例如,分析新闻文本流可以发现在某一个时期来自不同新闻媒体中大量地、集中地出现一些相近或者相关的主题,反映出不同时期的舆论焦点.热点可以分为局部热点和全局热点.局部热点是单一文本流中出现的热点,而全局热点是同时存在于多个文本流中的热点.热点根据相似程度的高低,可以组织成层次结构,图 1 中给出了一个例子.

文本流热点挖掘已经取得了一批重要成果[1−13].但是,现有工作仍然存在下述局限:

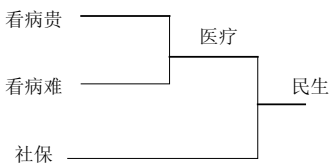


Fig.1 Hierarchy of hotspots

图 1 热点的层次结构

(1) 在评价文本流的语义相似程度时,主要采用基于统计分布的方法,其主要缺点是擅长发现“形似”(即具有相似的概率分布曲线),很难发现“神似”(即信息意义上的相似).

例 1:考察以{A,B,C,D}为字母的 4 个文本流片段 X_1=(ABAAC),X_2=(BABBC),X_3=(ADABC),X_4=(BABAC)组成的样本集合,由于条件概率 P(A|C)=P(B|C)=1/2,因此,传统的方法将认为 A 和 B 在语义上是相似的,但这往往与实际不相符.例如,在本文实验中引用的新闻文本流数据中,P(“民生”|“总理”)和 P(“科学发展”|“总理”)是基本一致的,但是,“民生”和“科学发展”在语义上显然是不一样的.因此,仅仅依赖统计的方法,只能判断信息的“量”是否接近,难以判断信息的“质”是否相似.

(2) 现有方法忽略了对文本流自身是否存在冗余信息的判断,从而难以发现有意义的全局热点.

例 2:考察 X_5=X_6=(ABCDE),尽管 X_5 和 X_6 的内容完全一致,但是由于 X_5 和 X_6 自身没有冗余信息,没有局部热点,所以(ABCDE)是平凡的、没有意义的全局热点.

(3) 缺乏对热点层次结构的挖掘,以及无需先验参数且过程可解释的方法.在同一个时期,不同文本流之间可能存在不同的局部热点.

例 3:考虑如下 4 个以{A,B,C,D}为字母的文本流片段:X_7=(AAABC),X_8=(AABAD),X_9=(CDCCC),X_{10}=(BCCBC).直观来说,根据共享信息的不同,可以发现有两个热点,其中,X_7 和 X_8 存在热点 A,X_8 和 X_9 之间存在热点 C.这里,热点的个数并不应该先验地设定,而是根据信息的近似程度自动地在文本流之间进行划分得到的,因而也是可解释的(信息是近似的).

本文对上述问题进行了研究.首先有如下重要观察:

观察 1. 当某个全局热点出现时,从时间纵向来看,相同的字母(代表一个主题、事件等分类型数据)在某个时间区间内密集出现在单一文本流中,从而形成局部热点;同时从横向来看,在相同的时间区间内,该字母也在其他文本流中密集出现,从而形成全局热点.这时:(1) 每个单一文本流自身因为具有局部热点而存在较多的冗余信息,从而具有较大的可压缩性,这种可压缩性可以通过 Kolmogorov 复杂度来度量.(2) 多个文本流之间如果存在全局热点,那么,它们必然因为存在共享信息而彼此相似.这种相似性可以通过从一个文本流构造出另一个文本流的难易程度来评价,而后者可以由条件 Kolmogorov 复杂度来衡量.(3) 热点之间根据共享信息的多少,组织成了一个树形的层次结构,热点离根节点越远,共享的信息越多.

在上述观察的启发下,本文做了以下工作:(1) 基于 Kolmogorov 复杂度定义了文本流冗余信息量的概念,并证明了它与文本流局部热点的关系;(2) 提出了一种文本流相似性度量指标:流信息距离 SID(stream information distance),以衡量文本流之间的信息相似程度;(3) 借鉴计算生物学的种系发生树(phylogeny)的概念,提出了一种基于 SID 多文本流层次聚类算法,以挖掘多文本流的热点及其层次结构.

1 相关工作

文献[4–6]从新闻文本流中发现和跟踪主题和事件,这方面的工作称作主题发现和跟踪(topic detection and tracking,简称 TDT).现有关于 TDT 的工作的主要缺点是只考虑了单一的文本流,没有考虑多个文本流之间的相关性,更没有考虑主题的层次结构.

文献[7]提出了使用无穷自动机来识别数据流突发特征及其内在结构的方法.基于文献[7]的开创性工作,文

献[2,8−10]先后提出了一系列挖掘数据流突发模式的方法.但是,这些工作主要局限在具有突发性的特征的识别上,没有从这些特征中去发现有意义的主题或者热点.文献[1,3,11−13]主要从分析数据的统计特征出发挖掘数据流的突发事件.其主要缺点是,只能判断概率分布的相似性,无法准确识别信息的实际含义是否相似,而且缺乏对冗余信息的判断,所以有可能发现无意义的、平凡的事件.文献[3,11]还将聚类技术引入突发模式挖掘,但是没有考虑聚类的层次关系,从而不能发现主题事件之间内在的层次结构.文献[13]将小波分析和滑动窗口技术应用到了数据流的突发事件挖掘,但是它主要局限于数值型数据的分析,且依赖于窗口尺寸的先验设置.

在评价信息的距离方面,文献[14,15]提出了基于本体计算文本语义距离的方法.但是,该方法需要首先从语料库中构建本体树,在本质上仍然是基于统计分析的方法,其缺点与前面所述类似.文献[16,17]提出了基于Kolmogorov 复杂度计算字符串信息距离的思想.由于 Kolmogorov 复杂度本身不依赖于任何概率分布,而是从字符串相互变换的角度衡量其相似性,因此成为本文工作的基础之一.

2 Kolmogorov 复杂度

首先简要介绍与本文工作密切相关的几个基本概念和性质.Kolmogorov 复杂度衡量了非随机变量的信息含量,称为算法信息量[18],以下简称为信息量.设 x 是一个有限长度的字符串,U 是一个通用图灵机,$l(x)$表示 x 的长度,$U(p)$表示程序 p 在计算机 U 上的输出.则 Kolmogorov 复杂度定义如下:

定义 1(Kolmogorov 复杂度)[18]. 字符串 x 的 Komogorov 复杂度 $K_U(x)$定义为在 U 上产生 x 的程序的最小长度,即 $K_U(x)=\min\{l(p)|U(p)=x\}$,与 $K_U(x)$对应的程序 p 记为 x^*.字符串 x 关于 y 的条件 Kolmogorov 复杂度 $K_U(x|y)$定义为 x 在已知 y^*时的最小描述长度,即 $K_U(x|y)=\min\{l(p)|U(p,y^*)=x\}$.

文献[18]证明,对于不同的 U,$K_U(x)$之间仅相差一个与 x 无关的常数,因此 $K_U(x)$可简记为 $K(x)$.为了描述 Kolmogorov 复杂度的性质,下面先引入“近似相等”的概念:

定义 2(近似相等). 如果变量 A 和变量 B 最多相差一个常数,即存在常数 c,使$|A-B|\leqslant c$,则称 A 和 B 是近似相等的,记为 $A\backsimeq B$.

关于近似相等关系有下述引理:

引理 1. 近似相等关系满足等价关系.

证明:(1) 自反性:对于任意变量 A,$|A-A|\leqslant 0$,所以 $A\backsimeq A$;(2) 对称性:对于变量 A 和 B,如果 $A\backsimeq B$,则$|A-B|=|B-A|\leqslant c$,所以 $B\backsimeq A$;(3) 传递性:对于变量 A,B,C,如果 $A\backsimeq B,B\backsimeq C$,即有$|A-B|\leqslant c_1,|B-C|\leqslant c_2$,则由三角不等式可知,$|A-C|\leqslant c_1+c_2$,即 $A\backsimeq C$. □

本文后面将不区别相等和近似相等,简单地把“$\backsimeq$”记为“=”.关于 Kolmogorov 复杂度有如下结论成立:

引理 2[19]. 设 x,y 和 z 都是任意的字符串,则有 $0\leqslant K(x)\leqslant l(x),K(xy)=K(yx)=K(x)+K(y|x)=K(y)+K(x|y),K(xy)\geqslant K(x),K(xy)\geqslant K(y),K(xx)=K(x),K(x|z)\leqslant K(x|y)+K(y|z)$成立.

引理 3[19]. 任意给定字符串 x_1,x_2 和 y,如果 $K(x_1)=K(x_2)$,则 $K(x_1y)=K(x_2y)$和 $K(yx_1)=K(yx_2)$成立.

根据上述定义和引理,可以导出如下 3 点启发性知识:

Idea 1. 一个字符序列包含的信息量的大小取决于通用图灵机为产生这一序列所需要的最短程序的长度.显然,冗余信息越多(越有规律)的字符序列所需要的程序越短;反之,所需要的程序越长,极端情况下将与序列一样长,成为不可压缩的字符串.

Idea 2. 条件 Kolmogorov 复杂度 $K(x|y)$和 $K(y|x)$实际上给出了从一个字符序列产生另一个字符序列所需要的最小信息量,这个值越小,说明 x 和 y 共享的信息越多.

Idea 3. 从通信的角度看,如果把定义 1 中的 x 看作是信源的输出序列,则 x^*可以看作是 x 的压缩编码,而 Kolmogorov 复杂度 $K(x)$正是 x 的压缩编码长度的理论极限值.

需要说明的是,Kolmogorov 复杂度是不可计算的[18],在应用中常常采用启发式的近似算法来求解.本文采用文献[17]提出的方法,使用标准压缩算法 bzip2 来作为 Kolmogorov 复杂度的近似计算方法.

3 局部热点

在具体展开以前首先声明,本文所研究的文本流中的数据都是分类型数据.因此,一个文本流可以用某个字母表来编码成字符串,一个字母可以代表一个事件、类别等等分类型数据.在此意义上,本文研究的数据对象可抽象为字符串.

定义 3(*M*-文本流集合). *M*-文本流集合定义为集合 $\boldsymbol{X}(M)=\{X_1,\ldots,X_M\}$,其中,$M$ 是文本流的个数,$X_i=\langle x_{i1}x_{i2}\ldots x_{in}\rangle(1\leqslant i\leqslant M)$是第 i 个的文本流,$\langle x_{i1}x_{i2}\ldots x_{in}\rangle$是 X_i 在时间区间$[1,n]$内的字符序列.

定义 4(局部热点). 给定 *M*-文本流集合 $\boldsymbol{X}(M)$,文本流 $X_i\in\boldsymbol{X}(M),1\leqslant i\leqslant M$,在某个时间区间 $T=[s,e]$内的序列片段为 $X_{iT}=\langle x_{is}x_{i(s+1)}\ldots x_{ie}\rangle$,其概率分布为 P.令 $W_{iT}=x_{is}\cup x_{i(s+1)}\cup\ldots\cup x_{ie}$ 为其字母表,给定熵的上限阈值$\theta<\ln|W_{iT}|$,如果 X_i 在时间区间 T 内的熵 $H(X_{iT})<\theta$,则称此时具有最大概率值的字符 w 为 X_i 在时间区间 T 内的θ-局部热点,记为 $LH(X_i,T,\theta)=\{w|H(X_{iT})<\theta$,且 $w=\arg\max_{w\in W_{iT}} P(w)\}$.其中, $H(X_{iT})=-\sum_{j=1}^{|W_{iT}|}P(w_j)\ln P(w_j), w_j\in W_{iT}$.当没有歧义时,$\theta$-局部热点简称为局部热点.

注意:(1) 局部热点是局部于单一文本流的,是本文所考察的文本流热点中层次最低的,它是多文本流存在全局热点的前提;(2) 局部热点是在文本流的熵值低于某个上限阈值的前提下定义的.这里对于熵值的限定,实质上是对文本流可压缩性的限定.文本流的冗余信息量定义如下:

定义 5(文本流的冗余信息量). 文本流 X_i 在时间区间 $T=[s,e]$内的冗余信息量 $RI(X_i,T)$定义为序列 $X_{iT}=\langle x_{is}x_{i(s+1)}\ldots x_{ie}\rangle$的 Kolmogorov 复杂度的倒数,即 $RI(X_{iT})=1/K(X_{iT})=1/K(x_{is}x_{i(s+1)}\ldots x_{ie})$.

根据 Idea 1,文本流的可压缩性取决于它的冗余信息量,命题 1 揭示了文本流的局部热点与其冗余信息量之间的关系.

命题 1. 已知文本流 X_i 在某个时间区间 $T=[s,e]$内的序列片段为 $X_{iT}=\langle x_{is}x_{i(s+1)}\ldots x_{ie}\rangle,|X_{iT}|=n$,字母表为 $W_{iT}=x_{is}\cup x_{i(s+1)}\cup\ldots\cup x_{ie},|W_{iT}|=m$,则 $LH(X_i,T,\theta)\neq\varnothing$的必要条件是 $RI(X_{iT})>1/(n\theta+(m+1)\log n)$.

证明:可以分两步描述序列 X_{iT}:首先描述 X_{iT} 中各个字符的概率分布.由于有 m 个不同的字符,所以需要 $(m+1)\log n$ 比特;其次,需要区分与 X_{iT} 具有相同概率分布的其他序列,由于这样的序列的个数(包括 X_{iT} 在内)不超过 $2^{nH(X_{iT})}$ 个,所以需要 $nH(X_{iT})$比特.因此,$K(X_{iT})\leqslant nH(X_{iT})+(m+1)\log n$.又根据定义 4,$LH(X_i,T,\theta)\neq\varnothing$时有 $H(X_{iT})<\theta$,所以,$K(X_{iT})<n\theta+(m+1)\log n$.所以,由定义 5,$RI(X_{iT})>1/(n\theta+(m+1)\log n)$. □

注意,命题 1 表明,一个文本流只有当其冗余信息多到某种程度时才可能存在局部热点.这与我们在观察 1 中看到的现象是吻合的.

4 流信息距离 SID(stream information distance)

本节描述两个文本流之间的信息相似度指标:流信息距离(stream information distance,简称 SID).根据观察 1 和 Idea 2,文本流之间存在全局热点时必然会因为共享冗余信息而相似.这种相似性反映在不同文本流之间互相转换的难易程度上,而后者可由条件 Kolmogorov 复杂度来衡量.根据这个思路,下面给出 SID 的定义,并证明其满足度量的 3 个基本性质.

定义 6(流信息距离 SID). 两个文本流 X,Y 的流信息距离 $SID(X,Y)$定义为

$$SID(X,Y)=1-\{[K(Y)-K(Y|X)]/K(XY)\} \tag{1}$$

在公式(1)的分子 $K(Y)-K(Y|X)$中,$K(Y)$是单独描述 Y 需要的长度,$K(Y|X)$是在 X 的帮助下描述 Y 需要的长度,二者的差值就是 X 中包含的关于 Y 的信息量.显然,这个差值越大,X 和 Y 的信息越接近,$SID(X,Y)$也越小.但是,仅仅使用 $K(Y)-K(Y|X)$是不够的.考虑如下情形:假设 X 和 Y 共享了 100 比特的信息,Y 的长度为 200 比特,那么,当 X 的长度为 200 比特时,X 和 Y 之间的距离显然应该比 X 的长度为 2 000 比特时的距离要小.但是这两种情况下,$K(Y)-K(Y|X)$的值都是相等的.所以,为了避免由于数据长度不对称造成的偏差,公式(1)中采用 $K(XY)$作分母,以实现归一化.

命题 2. SID 满足:(1)非负性:$SID(X,Y)\geqslant 0$,且当且仅当 $X=Y$ 时,$SID(X,Y)=0$;(2) 对称性:$SID(X,Y)=SID(Y,X)$;

(3) 三角不等式:$SID(X,Z)\leqslant SID(X,Y)+SID(Y,Z)$.

证明:(1) 根据引理 2,$K(XY)\geqslant K(Y)\geqslant K(Y)-K(Y|X)$,所以$[K(Y)-K(Y|X)]/K(XY)\leqslant 1$,所以 $SID(X,Y)\geqslant 0$.当 $X=Y$ 时,$SID(X,Y)=SID(X,X)=1-\{[K(X)-K(X|X)]/K(XX)\}$.注意到 $K(X|X)=0$,$K(XX)=K(X)$,故 $SID(X,Y)=0$;反之,当 $SID(X,Y)=0$ 时,$K(Y)-K(Y|X)=K(XY)$,$K(XY)=K(Y)+K(X|Y)$,所以 $K(X|Y)+K(Y|X)=0$.由 Kolmogorov 复杂度的非负性(引理 2)可知,$K(X|Y)=K(Y|X)=0$,所以 $X=Y$.

(2) $SID(X,Y)=1-\{[K(Y)-K(Y|X)]/K(XY)\}=[K(XY)-K(Y)+K(Y|X)]/K(XY)$,又因为 $K(XY)=K(YX)=K(Y)+K(X|Y)$,所以 $SID(X,Y)=[K(X|Y)+K(Y|X)]/K(XY)=SID(Y,X)$.

(3) 根据步骤(2),即证 $SID(X,Z)=[K(X|Z)+K(Z|X)]/K(XZ)\leqslant[K(X|Y)+K(Y|X)]/K(XY)+[K(Y|Z)+K(Z|Y)]/K(YZ)$,即证下述两个不等式成立:

$$K(X|Z)/K(XZ)\leqslant K(X|Y)/K(XY)+K(Y|Z)/K(YZ) \quad (a)$$

$$K(Z|X)/K(XZ)\leqslant K(Y|X)/K(XY)+K(Z|Y)/K(YZ) \quad (b)$$

下面先证不等式(a).令 $w=K(X|Z)$,$u=K(X|Y)$,$v=K(Y|Z)$,根据引理 2,有 $w\leqslant u+v$ 成立,又令 $w=u+v-\Delta$,则有

$$\begin{aligned}K(X|Z)/K(XZ)&\leqslant w/[K(Z)+w]\\&\leqslant(u+v-\Delta)/[K(Z)+u+v-\Delta]\\&=1/[K(Z)/(u+v-\Delta)+1]\\&\leqslant 1/[K(Z)/(u+v)+1]\\&=(u+v)/[K(Z)+u+v]\\&=u/[K(Z)+u+v]+v/[K(Z)+u+v]\\&=u/[K(Z)+u+v]+v/[K(Z)+u+v].\end{aligned}$$

注意,根据引理 2,$K(Z)+u+v-K(XY)=K(Z)+K(X|Y)+K(Y|Z)-K(XY)=K(YZ)-K(Y)\geqslant 0$,所以 $K(Z)+u+v\geqslant K(XY)$.同理,$K(Z)+u+v\geqslant K(YZ)$,所以 $K(X|Z)/K(XZ)\leqslant u/K(XY)+v/K(YZ)=K(X|Y)/K(XY)+K(Y|Z)/K(YZ)$.

即不等式(a)成立,同理可证不等式(b)成立.所以,$SID(X,Z)\leqslant SID(X,Y)SID(Y,Z)$成立. □

命题 2 表明,SID 满足距离度量的 3 个重要性质,因此是有效的距离指标.本文后面将应用 SID 作为距离尺度挖掘挖掘文本流热点.

5 全局热点

基于 SID 可以对 M-文本流进行有效的聚类划分,每一个簇即意味着一个全局热点.同时,为了揭示热点之间内在的层次关系,采用层次聚类技术,使得位于最低层次的是每个文本流自身的局部热点.因此,基于 SID 的全局热点挖掘实际上是以 SID 为相似性度量的多文本流层次聚类.

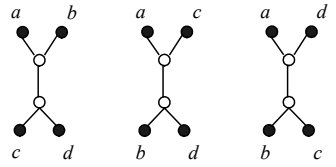
定义 7(热点树). M-文本流集合 $\boldsymbol{X}(M)$的全局热点树 HT 是一棵层次聚类树,其叶子节点表示一个具有局部热点的文本流,非叶子节点表示一个全局热点.

为了得到多文本流的热点树,必须实现对 M-文本流的层次聚类.为此,本文借鉴了计算生物学中种系发生树(phylogeny)的思想[20],设计了一种层次聚类算法 HCMS(hierarchical clustering over multiple streams).该算法无须验地指定聚簇数 k.下面首先引入若干基本概念:

定义 8(四元树、一致集合 $\boldsymbol{Q}_{HT}$). (1) 一个四元树定义为一棵具有 6 个节点的树,其中包括 4 个叶子节点和 2 个非叶子节点,且非叶子节点的度均为 3;(2) 给定一棵热点树 HT 和一个四元树 $ab|cd$,如果在 HT 中 a,b 间的路径与 c,d 间的路径之间没有公共节点,则称 HT 与四元树 $ab|cd$ 是一致的,与 HT 一致的四元树的集合记为 $\boldsymbol{Q}_{HT}$.

事实上,四元树中的非叶子节点将 4 个元素划分成两组,每组的两个叶子节点位于同一个父节点下.于是,一棵四元树可以记为 $ab|cd$.其中,叶子节点 ab 为一组,而 cd 是另一组.集合 $\boldsymbol{X}=\{a,b,c,d\}$的四元树如图 2 所示.

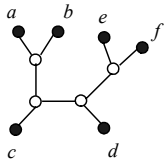
图 3 给出了一个 $\boldsymbol{Q}_{HT}$的例子.可以以文本流为叶子节点,首先从 $\boldsymbol{X}(M)$中生成一个四元树集合 $\boldsymbol{Q}_{HT}$,然后基于这个集合构建出 $\boldsymbol{X}(M)$的热点树 HT,从而完成层次聚类,实现 $\boldsymbol{X}(M)$的热点挖掘.



(a) *ab*|*cd* (b) *ac*|*bd* (c) *ad*|*bc*

Fig.2 An example of 4-tree

图 2 一个四元树的示例



$\boldsymbol{Q}_T$={*ab*|*cd*,*ab*|*ce*,*ab*|*cf*, *ab*|*de*,*ab*|*df*,*ab*|*ef*,*ac*|*de*, *ac*|*df*,*ac*|*ef*,*bc*|*de*,*bc*|*df*, *bc*|*ef*,*ad*|*ef*,*bd*|*ef*,*cd*|*ef*}

(a) *HT* (b) $\boldsymbol{Q}_{HT}$

Fig.3 An example of $\boldsymbol{Q}_{HT}$

图 3 $\boldsymbol{Q}_{HT}$的示例

接下来的问题是,已知 $\boldsymbol{Q}_{HT}$,如何求出 *HT*.我们的思路是,使四元树中相邻叶子节点所代表的文本流之间的 SID 越小越好.首先给出节点距离的定义.

定义 9(节点距离). 设 *a*,*b*,*c*,*d* 是 4 个文本流,则四元树 *ab*|*cd* 的节点距离 *D*(*ab*|*cd*)定义为

$$D(ab|cd)=SID(a,b)+SID(c,d) \tag{2}$$

热点树 *HT* 的节点距离 *D*(*HT*)定义为

$$D(HT)=\sum_{ab|cd\in \boldsymbol{Q}_{HT}} D(ab\,|\,cd) \tag{3}$$

设要参加聚类的文本流集合为 $\boldsymbol{X}$,$|\boldsymbol{X}|=n$,将 $\boldsymbol{X}$ 中的文本流分成 4 个一组,则共有 C_n^4 组,对其中任意一组{*a*,*b*,*c*,*d*},有 $C_4^2/2=3$ 个四元树,分别是 *ab*|*cd*,*ac*|*bd*,*ad*|*bc*.令

- $D_{\min}(\{a,b,c,d\})=\min\{D(ab|cd),D(ac|bd),D(ad|bc)\}$;
- $D_{\min}(\boldsymbol{X})=\sum_{\{a,b,c,d\}\subseteq \boldsymbol{X}} D_{\min}(\{a,b,c,d\})$;
- $D_{\max}(\{a,b,c,d\})=\max\{D(ab|cd),D(ac|bd),D(ad|bc)\}$;
- $D_{\max}(\boldsymbol{X})=\sum_{\{a,b,c,d\}\subseteq \boldsymbol{X}} D_{\max}(\{a,b,c,d\})$.

设 *HT* 是 $\boldsymbol{X}$ 的热点树,则显然有 $D_{\min}(\boldsymbol{X})\leqslant D(HT)\leqslant D_{\max}(\boldsymbol{X})$.我们的目标就是要寻找 $\boldsymbol{X}$ 的热点树 *HT*,使得 *D*(*HT*)尽可能地小.显然,最好的结果是 $D(HT)=D_{\min}(\boldsymbol{X})$,最坏的结果是 $D(HT)=D_{\max}(\boldsymbol{X})$.基于上述分析,提出下述优化目标:

$$\left.\begin{aligned}&\min S(HT)=\frac{D_{\max}(\boldsymbol{X})-D(HT)}{D_{\max}(\boldsymbol{X})-D_{\min}(\boldsymbol{X})}\\&\text{s.t. } D_{\min}(\boldsymbol{X})\leqslant D(HT)\leqslant D_{\max}(\boldsymbol{X})\end{aligned}\right\} \tag{4}$$

在公式(4)中,*S*(*HT*)是一个线性的归一化的记分函数,$0\leqslant S(HT)\leqslant 1$.且当 $D(HT)=D_{\min}(\boldsymbol{X})$时,*S*(*HT*)=1;当 $D(HT)=D_{\max}(\boldsymbol{X})$时,*S*(*HT*)=0.

下面给出一种随机的、启发式算法来求解公式(4).首先定义 3 个基本变换操作.

定义 10(基本变换操作). 在一棵热点树 *HT* 上可以进行下述 3 种基本变异操作:

(1) 叶子节点交换 LS(leaf swap):随机选择两个叶子节点并交换它们的位置;

(2) 子树交换 SS(subtree swap):随机选择两个非叶子节点,然后交换以它们为根的两棵子树;

(3) 子树转移 ST(subtree transfer):随机选择一棵子树,并随机地转移到其他位置,同时保持热点树的性质不变.

定义 10 中的 3 种基本变换可以构成 2^3=8 种变异组合(可用 3 位二进制数对组合进行编码,0 表示不执行,1 表示执行,每一位对应一个变异操作.例如,最高位为 LS,次高位为 SS,最低位为 ST,则编码为 000~111).下面的算法 1 给出了在多文本流上的层次聚类算法 HCMS 的主要步骤.

算法 1. *HCMS*(X,α).

输入:文本流集合 $\boldsymbol{X}$;记分阈值α.

输出:热点树 *HT*.

(1) n=|$\boldsymbol{X}$|

(2) 随机生成具有 2n–2 个节点的热点树 *HT*,其中,n 个叶子节点表示文本流的局部热点,n–2 个非叶子节点表示不同层次的全局热点.

(3) 根据公式(4)计算 S(*HT*);

(4) while (S(*HT*)<α={

(5) 从定义 10 给出的变换操作的 8 个组合中随机选择一个在 *HT* 上执行,得到结果集 $\boldsymbol{MT}$;

(6) 从 $\boldsymbol{MT}$ 中选择 S(*T*)最大的树 *T*;

(7) if S(*T*)>S(*HT*), *HT*=*T*;

(8) }

(9) return *HT*;

算法 1 首先随机地生成一棵热点树 *HT*(n 个叶子节点,n–2 个非叶子节点),然后以此为基础,随机地选择一个变异操作组合,从结果集中找出具有最大记分的树.如果其记分大于 *HT*,则取而代之.重复这一过程,直到 S(*HT*)小于一个事先指定的阈值α为止.算法 1 的收敛性在后面的实验中得到了验证.注意,由于以公式(4)作为优化目标,因此无须事先指定聚簇数 k.

算法 2 给出了全局热点挖掘的主要步骤.

算法 2. *GlobalHotspots*(X(M),T=[s,e],θ,α).

输入:M-文本流 $\boldsymbol{X}$(M)={X_1,…,X_M};时间区间 *T*;局部热点阈值θ;记分阈值α.

输出:热点树 *HT*.

(1) $\boldsymbol{X}'$={};

(2) foreach X_i∈$\boldsymbol{X}$(M) {

(3) n=|X_{iT}|, where X_{iT}=〈$x_{is}x_{i(s+1)}$…x_{ie}〉;

(4) compute RI(X_{iT});

(5) if RI(X_i,T)>1/($n\theta$+(m+1)log$n$)

(6) $\boldsymbol{X}'$=$\boldsymbol{X}'$∪{X_{iT}};

(7) }

(8) *HT*=*HCMS*($\boldsymbol{X}'$,α);

(9) return *HT*;

算法 2 首先根据命题 1 对文本流进行筛选,找出那些满足局部压缩性的文本流,组成集合 $\boldsymbol{X}'$,然后调用 HCMS(算法 1)得到代表全局热点层次结构的热点树 *HT*.

关于算法的收敛性.算法 2 的收敛性依赖于算法 1,所以这里着重讨论算法 1 的收敛性.算法 1 在本质上是一种遗传算法,其收敛性遵循遗传算法收敛性的有关理论[21].文献[21]证明了,如果一种遗传算法在每一次迭代过程中保留最优个体,则该遗传算法将收敛于全局最优解,即有如下定理:

定理 1[21]. 若一种遗传算法在选择后保留最优个体,则收敛于全局最优解.

算法 1 在每次迭代过程中都选取得分 S(*T*)最高的 *HT* 参与下次迭代(算法 1 第(6)行、第(7)行),符合定理 1

的条件.因此,算法 1 将收敛于全局最优.因此,算法 2 也将收敛于全局最优.

6 实验分析

实验分为 3 个部分:第 1 部分检验算法 1 的收敛性,第 2 部分在真实数据集上验证局部热点和冗余信息量的关系,第 3 部分应用算法 2 在真实数据集上挖掘全局热点.实验环境为:Intel Core 2 双核 CPU,主频 2.0GHz,2 级缓存 4MByte,内存 2GByte,Linux 操作系统,GCC 编译器,实验程序用 C 语言编制.

6.1 验证收敛性

首先,考察算法 1 在不同数据规模时的收敛情况.实验在分别由 M=50,100,200 个文本流构成的 3 个仿真文本流集合上进行.实验选取的记分阈值 α=0.8,同时去掉算法 1 第(4)行的条件判断,让循环(即算法 1 第(5)行~第(7)行)无限进行下去,以观察算法 1 是否收敛以及收敛的速度.

实验结果如图 4 所示.图中纵坐标是循环次数,横坐标是记分值 $S(HT)$.从图 4 可以得到以下结论:

(1) 对于不同的数据规模,算法 1 都可以稳定地收敛到 $S(HT)$=0.85 附近.

(2) 随着数据规模的增大,收敛速度变慢.当 M=50 时,算法 1 循环大约循环 400 次后进入收敛状态;当 M=100 时,需要循环 800 次左右;当 M=200 时,需要循环大约 1 100 次后才进入收敛状态.

其次,由于算法 1 是随机算法,其初始值是随机产生的(算法 1 第(2)行),所以还需要考察不同的随机初值对于算法收敛性的影响.实验在 M=50 的文本流集合上执行 3 次,结果如图 5 所示.从图 5 可以看出,当初始值不同时,算法 HCMS 仍然可以稳定地收敛,但是收敛速度不同,当初始值较大时,收敛速度较快.

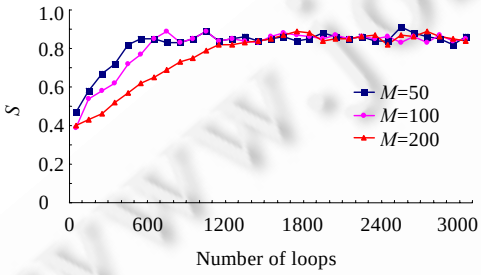


Fig.4 Convergence of HCMS on variant data scales

图 4 不同数据规模时 HCMS 的收敛性

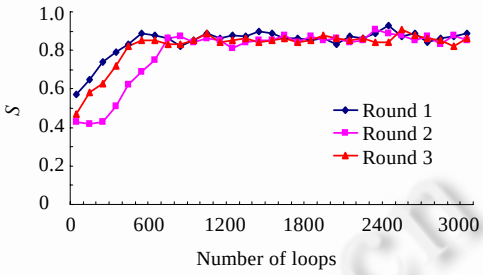


Fig.5 Convergence of HCMS on variant initializations

图 5 不同初始值时 HCMS 的收敛性

6.2 挖掘局部热点

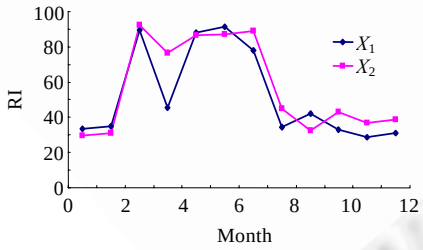


Fig.6 Distribution of RI

图 6 冗余信息量的分布

根据本文的理论,当文本流出现局部热点时,其冗余信息量应该比较大(第 2 节).为了验证这一点,我们选取人民日报和四川日报 2003 年全年头版报道作为实验数据,以月为时间单位,观察其冗余信息量的变化.实验结果如图 6 所示.其中,X_1 表示人民日报,X_2 表示四川日报.

从图 6 可以看出,X_1 和 X_2 在 2003 年的冗余信息量明显呈现两个峰值,分别是在 3 月和 5 月~7 月.事实上,3 月的舆论热点是我国正在召开的两会,而 5 月~7 月全国上下抗击非典成为舆论的焦点.X_1 和 X_2 在这两个时期的冗余信息量明显高于平常时期,这就验证了本文前面提出的观点,当文本流存在局部热点时,其可压缩性较大,冗余信息量较多.因此,图 6 中的冗余信息量随时间变化的曲线准确地反映了 X_1 和 X_2 两个文本流的局部热点的存在和变化.

6.3 挖掘全局热点

实验选取 6 家媒体作为样本,考察它们在 2009 年 8 月的头版新闻构成的文本流.这 6 家媒体分别是:X_1=人民日报,X_2=搜狐网,X_3=新华网,X_4=凤凰网,X_5=光明日报,X_6=成都日报.把算法 2 应用到 $\boldsymbol{X}(6)=\{X_1,\ldots,X_6\}$,参数 $\theta=0.6$,$\alpha=0.8$.实验得到的热点树如图 7 所示.

由图 7 可知,人民日报(X_1)、新华网(X_3)和光明日报(X_5)作为中央官方媒体,在 8 月较多地报道了经济发展方面的新闻.而成都作为国家城乡改革实验区,其官方媒体成都日报(X_6)更多地关注城乡统筹方面的消息.上述 4 家媒体的热点在热点树中的近似程度虽然有差别(X_1 和 X_3 最为接近),但都是“经济”节点的子节点,说明它们都属于经济这个大范畴.而搜狐网和凤凰网作为两家民间网络媒体,在 8 月更多地关注莫拉克台风给台湾带来的灾害,成为民生方面的热点.

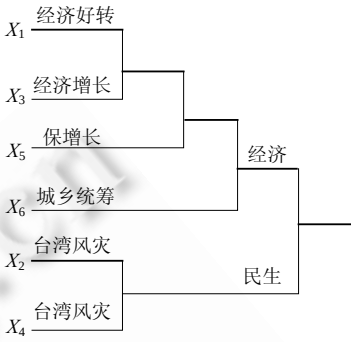


Fig.7 Hotspot tree of $\boldsymbol{X}(6)$

图 7 $\boldsymbol{X}(6)$的热点树

此外,我们将文献[4]提出的算法应用于本节的 6 个文本流,结果得到的热点是“经济”和“台湾”两个主题.与算法 2 得到的结果(如图 7 所示)比较,该结果没有揭示文本流局部热点和全局热点的层次关系.

上述结果表明,算法 2 由于考虑了单个文本流中的冗余信息,并采用流信息距离 SID 作为信息相似度的度量指标,能够较为准确地发现文本流的局部热点,以及多个文本流之间的全局热点及其内在层次关系.

7 结 论

本文将 Kolmogorov 复杂度应用到了多文本流热点挖掘中,定义并阐述冗余信息 *RI* 的概念,证明了它与局部热点的关系.并进一步提出了流信息距离 SID,作为文本流信息相似度的度量指标.最后,基于 SID 并借鉴计算生物学中种系发生树的概念,提出了一种利用层次聚类实现全局热点挖掘的算法.在仿真数据集上的实验验证了算法在不同数据规模下均具有较好的收敛性.同时,在真实数据集上的实验验证了本文关于 RI 与局部热点的关系的理论结论.同样地,在真实数据集上的实验表明,本文提出的热点挖掘算法能够对多个文本流中存在的局部热点进行层次聚类,从而较为准确地发现全局热点.

References:

[1] Parikh N, Sundaresan N. Scalable and near real-time burst detection from eCommerce queries. In: Proc.of the ACM KDD 2008. New York: ACM, 2008. 972−980. [doi: 10.1145/1401890.1402006]

[2] Lappas T, Arai B, Platakis M, Kotsakos D, Gunopulos D. On burstiness-aware search for document sequences. In: Proc.of the ACM KDD 2009. New York: ACM, 2009. 477−485.

[3] Fung GPC, Yu JX, Yu PS, Lu HJ. Parameter free bursty events detection in text streams. In: Proc. of the VLDB 2005. Trondheim: VLDB Endowment, 2005. 181−192.

[4] Allan J, Carbonell J, Doddington G, Yamron J, Yang YM. Topic detection and tracking pilot study: Final report. In: Proc. of the DARPA Broadcast News Transcription and Understanding Workshop. Arlington: NSF, 1998. 65−74.

[5] Brants T, Chen F, Farahat A. A system for new event detection. In: Proc. of the 26th ACM SIGIR Int'l Conf. on Research and Development in Information Retrieval (SIGIR 2003). Toronto: ACM, 2003. 102−112. [doi: 10.1145/860435.860495]

[6] Yang YM, Ault T, Pierce T, Lattimer CW. Improving text categorization methods for event tracking. In: Belkin NJ, Ingwersen P, Leong MK, eds. Proc. of the SIGIR 2000. Athens: ACM, 2000. 65−72. [doi: 10.1145/345508.345550]

[7] Kleinberg J. Bursty and hierarchical structure in streams. In: Proc. of KDD 2002. Edmonton: ACM, 2002. 91−101. [doi: 10.1145/775047.775061]

[8] Kumar R, Novak J, Raghavan P, Tomkins A. On the bursty evolution of blogspace. In: Proc. of the WWW 2003. Budapest: ACM, 2003. 568−576. [doi: 10.1145/775152.775233]

[9] He Q, Chang KY, Lim EP. Analyzing feature trajectories for event detection. In: Proc. of the SIGIR 2007. Amsterdam: ACM, 2007. 186−197. [doi: 10.1145/1277741.1277779]

[10] Vlachos M, Meek C, Vagena Z, Gunopulos D. Identifying similarities, periodicities and bursts for online search queries. In: Proc. of the SIGMOD 2004. New York: ACM, 2004. 131−142. [doi: 10.1145/1007568.1007586]

[11] Yuan ZJ, Du K, Jia Y, Miao JJ. Stream event detection: A unified framework for mining outlier, change and burst simultaneously over data stream. In: Proc. of the 7th IEEE Int'l Conf. on Data Mining Workshops. New York: ACM, 2007. 575−580. [doi: 10.1109/ICDMW.2007.55]

[12] Wang XH, Zhai CX, Hu X, Sproat R. Mining correlated bursty topic patterns from coordinated text streams. In: Proc of the 13th ACM SIGKDD Int'l Conf. on Knowledge Discovery and Data Mining. San Joes: ACM, 2007. 784−793. [doi: 10.1145/1281192.1281276]

[13] Zhu YY, Shasha D. Efficient elastic burst detection in data streams. In: Proc. of the KDD 2003. Washington: ACM, 2003. 542−562. [doi: 10.1145/956750.956789]

[14] Jiang JJ, Conrath DW. Semantic similarity based on corpus statistics and lexical taxonomy. In: Proc. of the Int'l Conf. on Research in Computational Linguistics. Berlin, Heidelberg: Springer-Verlag, 1997. 19−33.

[15] Budanitsky A, Hirst G. Semantic distance in WordNet: An experimental, application-oriented evaluation of five measures. In: Proc. of the Workshop on WordNet and other Lexical Resources. Pittsburgh: ACM, 2001. 95−100.

[16] Bennett CH, Gács P, Li M, Vitányi PMB, Zurek WH. Information distance. IEEE Trans. on Information Theory, 1998,44(4): 1407−1423. [doi: 10.1109/18.681318]

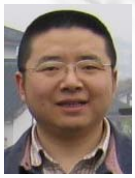
[17] Li M, Chen X, Li X, Ma B, Vitányi P. The similarity metric. In: Proc. of the 14th Annual ACM-SIAM Symp. on Discrete Algorithm. Baltimore: ACM, 2003. 863−872.

[18] Chaitin GJ. Algorithmic Information Theory. Cambridge: Cambridge University Press, 1987.

[19] Li M, Vitányi PMB. An Introduction to Kolmogorov Complexity and Its Applications. 2nd ed., New York: Springer-Verlag, 1997.

[20] Berry V, Gascuel O. Inferring evolutionary trees with strong combinatorial evidence. Theoretical Computer Science, 2000,240: 71−80. [doi: 10.1007/BFb0045078]

[21] Rudolph G. Convergence analysis of canonical genetic algorithms. IEEE Trans. on Neural Networks, 1994,5(1):96−101. [doi: 10.1109/72.265964]



杨宁(1974－),男,四川成都人,博士,讲师,CCF 会员,主要研究领域为机器学习,数据挖掘.

陈瑜(1974－),男,博士,讲师,主要研究领域为数据挖掘,计算智能.

唐常杰(1946－),男,教授,博士生导师,CCF 高级会员,主要研究领域为数据库系统,数据挖掘.

郑皎凌(1981－),女,博士,讲师,主要研究领域为数据库系统,数据挖掘.

王悦(1981－),男,博士,主要研究领域为数据库系统,数据挖掘.

李红军(1977－),男,博士,讲师,主要研究领域为数据库系统,数据挖掘.