基于关联规则的 Web 文档聚类算法*

宋擒豹, 沈钧毅

(西安交通大学 计算机科学与技术系,陕西 西安 710049)
E-mail: qbsong@mail.xjtu.edu.cn; jyshen@mail.xjtu.edu.cn
http://www.xjtu.edu.cn

摘要: Web 文档聚类可以有效地压缩搜索空间,加快检索速度,提高查询精度.提出了一种 Web 文档的聚类算法.该算法首先采用向量空间模型 VSM(vector space model)表示主题,根据主题表示文档;再以文档为事务,以主题为事务项,将文档和主题间的关系看作事务的形式,采用关联规则挖掘算法发现主题频集,相应的文档集即为初步文档类;然后依据类间距离和类内连接强度阈值合并、拆分类,最终实现文档聚类.实验结果表明,该算法是有效的,能处理文档类间固有的重叠情况,具有一定的实用价值.

关 键 词: 文档聚类;关联规则;Web 挖掘;WWW
中图法分类号: TP311 文献标识码: A

随着 WWW 的飞速发展,Internet 上的资源和服务均呈现出爆炸性增长的趋势.为了帮助人们有效地使用这些资源和服务,陆续有一些功能强大的搜索引擎问世了.这些搜索引擎在给人们带来很大便利的同时也暴露出搜索结果不能很好地满足用户需求的问题.Web 文档聚类技术可以缩减搜索空间,加快检索速度,提高查询精度,因而受到了人们的广泛关注[1~6].

Web 文档聚类主要有基于概率和基于距离的两类方法.基于概率的方法[5,6]以贝叶斯概率为理论基础,用概率的分布方式描述聚类结果,可以处理类间相互重叠的情况;缺点是当特征空间维数较高或特征值间呈现出较强的相关性时,聚类精度和效率均不能令人满意.基于距离的方法[4],如 K-均值和最近邻等,都以传统的特征向量表示文档,再将文档看作是向量空间中的一个点,通过计算点之间的距离进行聚类,比较形象直观;缺点是特征向量必须经过规范化处理以避免由于文档长度不同或各个文档间关键词出现的频度各异而产生的畸变,特别是当数据维数较高时,聚类的质量和算法的性能都明显下降.

我们用主题表示文档,将文档和主题间的关系描述成事务的形式,根据臻于成熟的关联规则挖掘算法[7]初步划分文档类,然后依照类间耦合度和类的内聚性进行聚类确认,有效地解决了上述方法中普遍存在的扩展性问题.

1 Web 文档的结构化表示

Web 文档是一种半结构化数据,为便于检索和查询,需要进行结构化处理.Web 文档表示就是抽取和描述其特征,并在此基础上建立特征的结构化描述的过程.

在对 Web 文档进行结构化表示的时候,我们首先用向量空间模型 VSM(vector space model)表示每一个主题,并根据建立的主题特征向量和文档内容形成文档的主题向量,再依此分别计算给定文档与这些主题间的关联度,然后根据关联度创建文档-主题事务矩阵.最后,对文档-主题事务矩阵中的行向量(即事务)进行规范化处

* 收稿日期: 2000-04-04; 修改日期: 2000-08-28
基金项目: 国家自然科学基金资助项目(60173058);国家 863 青年基金资助项目(863-306-QN2000-5)
作者简介: 宋擒豹(1966－),男,陕西华县人,博士,副教授,主要研究领域为数据挖掘,知识工程,计算机网络安全;沈钧毅(1939－),男,江苏扬州人,教授,博士生导师,主要研究领域为数据库理论,数据挖掘,数据仓库.

理,将它转换成单位向量,以使关联度之间具有可比性.

下面我们依次对上述 Web 文档结构化过程中用到的概念进行具体定义和阐述.

定义 **1.** 主题特征向量.设 T 是主题的集合,对于其中的每一个主题 $T_i \in T$,我们用特征向量

$$\vec{T_i} = \left[(k_{i,1}, w_{i,1}), (k_{i,2}, w_{i,2}), ..., (k_{i,j}, w_{i,j}), ..., (k_{i,l}, w_{i,l})\right]^T$$

表示.其中,$k_{i,j}$ 代表主题 T_i 中的第 j 个关键字/短语;$w_{i,j}$ 为第 j 个关键字/短语 $k_{i,j}$ 对应的权值,表示该关键字/短语在该主题中的重要程度,且 $\sum w_{i,j} = 1, 1 \le j \le l$; $l = \left\|\vec{T_i}\right\|$,为主题 T_i 中关键字/短语的个数,各个主题的 l 依实际情况而定,可以不同.

用 VSM 定义主题特征向量,代表主题的关键字/短语及其重要性可以根据具体情况来设定,各个主题的关键字个数也可以不同,这就充分兼顾了不同主题各自的具体情况,具有广泛的适用性和较强的可维护性.

定义 **2.** 文档的主题向量.设 D 是文档的集合,其中每一个文档 $D_j \in D$ 关于主题 T_i 的向量 $\overrightarrow{ToD_j}(T_i)$ 表示文档 D_j 对主题 T_i 的贡献,定义为

$$\left.\begin{aligned} \overrightarrow{ToD_j}(T_i) &= \left[\mu_{i,1}^j, \mu_{i,2}^j, ..., \mu_{i,k}^j, ..., \mu_{i,l}^j\right]^T, \\ \mu_{i,k}^j &= \left\|\cup K_{i,k}\right\| \Big/ \left\|\cup D_j\right\|^{\times w_{i,k}}, \end{aligned}\right\} \tag{2}$$

其中 $l = \left\|\vec{T_i}\right\|$,为向量 $\vec{T_i}$ 的长度;$\left\|\bigcup \kappa_{i,k}\right\|$ 是主题 T_i 的第 $k(1 \le k \le l)$ 个关键字/短语 $k_{i,k}$ 在 D_j 中出现的频度;$\left\|\bigcup D_j\right\|$ 为 D_j 中有效词的个数;$w_{i,k}$ 为第 k 个关键字/短语 $k_{i,k}$ 在主题特征向量 $\vec{T_i}$ 中的权值.

由定义可知,文档的主题向量反映的是文档与某一主题的特征向量之间的联系.由于主题特征向量的长度可以不同;即使长度相同,如果直接用它来表示文档,数据维数太高,故需做降维处理.为此,还需要分别计算文档与不同主题间的关联度.

定义 **3.** 文档与主题的关联度.关联度表示文档和某一主题之间的关联程度.文档 D_j 和主题 T_i 之间的关联度 $\lambda_{i,j}$ 可按下式计算:

$$\lambda_{i,j} = \sum_{k=1}^{l} \mu_{i,k}^j \ . \tag{3}$$

其中, $l = \left\|\vec{T_i}\right\|$,为向量 $\vec{T_i}$ 的长度; $\mu_{i,k}^j$ 表示文档 D_j 对主题 T_i 中第 k 个关键词/短语的贡献,由公式(2)产生.

关联度将文档与某一主题间的联系用一个数据项表示,将数据维数由原来的 $\|D\| \times \sum_{k=1}^{\|T\|} l_k$ 维降到了现在的 $\|D\|$ 维,大大压缩了数据量,为提高处理效率奠定了基础.

此时,我们就可以将文档视为事务,而将主题看作事务项(若关联度不为零,则对应的事务项出现,否则不出现),并建立式(4)所示的文档-主题事务矩阵:

$$\omega_{n\times m} = \begin{bmatrix} \lambda_{1,1} & \lambda_{1,2} & \cdots & \lambda_{1,j} & \cdots & \lambda_{1,m} \\ \lambda_{2,1} & \lambda_{2,2} & \cdots & \lambda_{2,j} & \cdots & \lambda_{2,m} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ \lambda_{i,1} & \lambda_{i,2} & \cdots & \lambda_{i,j} & \cdots & \lambda_{i,m} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ \lambda_{n,1} & \lambda_{n,2} & \cdots & \lambda_{n,j} & \cdots & \lambda_{n,m} \end{bmatrix} . \tag{4}$$

其中,$\lambda_{i,j}$ 为文档 D_j 和主题 T_i 之间的关联度,由式(3)计算得到;每个行向量代表一个事务;$1 \le i \le n, 1 \le j \le m$,$m$ 和 n 分别为主题和文档的数量.

建立了事务矩阵 $\omega_{n\times m}$ 之后,还要对每一行向量进行规范化处理,将其转换成单位向量,以使关联度之间具有可比性;同时,为区别起见,将由此产生的新矩阵称为规范化事务矩阵 $\omega'_{n\times m}$.

根据上述过程,我们给出下列建立规范化文档-主题事务矩阵的算法.

算法 **1.** 建立规范化文档-主题事务矩阵算法

Input: D: The set of Web documents;

T: The set of topic feature vectors;

Output: document-topic transaction matrix [λ];

Function: Set up document-topic transactions

(1) for (j:=1; $j \le \|D\|$; j++) do begin //set up document-topic transaction matrix [λ]

(2) for (i:= 1; $i \le \|T\|$; i++) do begin

(3) for each $k \le l$,computing $\mu_{i,k}^{j}$ according to formula (2);

(4) end for

(5) computing $\lambda_{i,j}$ according to formula (3);

(6) end for

(7) for (i:=1; $i \le \|D\|$; i++) do begin //normalizing matrix [λ]

(8) for (j:=1; $j \le \|T\|$; j++) do begin

(9) $\lambda_{i,j} = \lambda_{i,j} \Big/ \sum_{j=1}^{\|T\|} \lambda_{i,j}$;

(10) end for

本算法可分为两部分:第 1 部分计算文档和主题间的关联度,建立文档-主题事务矩阵,若用 NoK_i 表示主题 T_i 中关键词/短语的个数,其算法复杂度为 $O(\|D\| \times \|T\| \times \sum_i NoK_i)$;第 2 部分对文档-主题事务矩阵中的行向量进行单位化处理,形成规范化文档-主题事务矩阵,算法复杂度为 $O(\|D\| \times \|T\|)$.因此,建立文档-主题事务矩阵算法的复杂度为 $O(\|D\| \times \|T\| \times (\sum_i NoK_i + 1))$.

为了后文叙述方便,我们引入文档特征向量的概念.

定义 **4.** 文档特征向量.在规范化事务矩阵 $\omega'_{n\times m}$ 中,行向量描述了某一文档和所有 $\|T\|$ 个主题之间的关联关系,称为文档特征向量,并用 $\overrightarrow{vd}$ 表示.

2 基于关联规则的文档聚类

建立了文档-主题的规范化事务矩阵 $\omega'_{n\times m}$ 后,就可以采用关联规则挖掘算法发现主题频集,并将对应的文档集看作初步文档类,然后采用类验证技术对其进行确认.

2.1 根据关联规则发现文档类

在关联规则中,项集 $I = \{i_1, i_2, ..., i_n\}$ 表示 n 个不同数据项组成的集合,事务集 T 的每一个事务是项集 I 的一个子集.设 C 为 I 的一个子集,将 T 中 C 的支持数定义为包含 C 的事务数 $\sigma(C) = \|\{t \mid t \in T, C \subseteq t\}\|$.关联规则是一个形如 $X \xrightarrow{s,\ \alpha} Y$ 的表达式,其中 $X \subseteq I, Y \subseteq I$,$\alpha$为其置信度,定义为

$$\alpha(X \to Y) = \sigma(X \cup Y) \Big/ \sigma(X) \quad . \tag{5}$$

s 为支持度,定义如下:

$$s(X \to Y) = \frac{1}{\|T\|} \sigma(X \cup Y) . \tag{6}$$

关联规则挖掘的任务就是找出所有的 $X \xrightarrow{s,\ \alpha} Y$ 规则,使得α大于等于给定的置信度阈值,而 s 大于给定的支持度阈值.

由公式(6)可知,支持度 s 反映了蕴含关系 $X \to Y$ 在事务集 T 中出现的频度,频度超过指定阈值的蕴含关系所对应的项集则为频集.由此我们认为:之所以有频集产生,是因为对应的事务之间存在着某种程度的相似性.

在文档-主题事务矩阵中,我们将文档看作事务,将主题视为事务项.因此,如果某些事务项(即主题)经常一起出现在某些事务(即文档)中,那么对应的事务(即文档)自然也是相似的.换句话说,根据关联规则挖掘算法得到的频集,我们可以找到对应的事务集(即文档集),并能将它作为文档初步分类的结果.

但如前所述,支持度计算公式(6)反映的是某一项集在整个事务集中出现的频度,频度低于指定阈值的项集不能成为频集.我们将文档看作事务,将主题视为事务项,尽管有些事务项在整个事务集中出现的频度较低,但其对应的事务仍然是相似的.因此,支持度计算公式(6)对我们的问题来说已不再适用,需要对其进行重新定义.

定义 **5.** 支持度.设 $\Gamma=\bigcup_k \overrightarrow{vd}_k$, $\forall \psi \subset \Gamma$, 其支持度定义为

$$\delta(\psi)=\frac{1}{\|\psi\|}\sum_{i=1}^{\|\psi\|}\sum_{j=1}^{r}\lambda_{i,j}. \tag{7}$$

其中, $\|\psi\|>1$,是 Ψ中文档的个数;r 是 Ψ中各文档特征向量对应元素都不为 0 的元素个数.

显然,该定义将 Ψ的支持度$\delta(\Psi)$定义为 Ψ中所有文档与主题间完全非零关联度的平均值,有效地解决了若干个涉及相同主题的文档由于一般意义上的支持度不够而不能归为一类的情况.同时,也引入了模糊论的基本思想,不再用“1”或“0”来简单地描述一个文档和主题的关联情况,而是以区间[0,1]内的连续值表示,更符合客观实际.

利用关联规则发现文档类时,首先采用 Agrawal 等人[7]提出的频集快速发现算法,并根据本文定义的支持度计算公式获得主题频集;然后扫描数据库即可得到对应的文档集,此即初步文档类.频集快速发现算法在文献[7]中已得到了详尽的阐述,在此不再赘述.

2.2 对文档类进行确认

对关联规则发现的文档类进行确认分两步进行:(1) 计算不同文档类之间的耦合度,对耦合度大于指定阈值的类进行合并;(2) 度量每一文档类的内聚性,对内聚性小于指定阈值的类进行拆分.

我们用文档类间的距离表示它们之间的耦合度.

定义 **6(**类间距离**).** 为了增强类的内聚性,减弱类间耦合度,将文档类 p 和 q 间的距离定义为离差平方和的形式:

$$D_{p,q}^2=\frac{n_p\times n_q}{n_p+n_q}\left(\bar{x}_p-\bar{x}_q\right)^{\mathrm{T}}\left(\bar{x}_p-\bar{x}_q\right). \tag{8}$$

其中,n_p 和 n_q 分别为这两个类中文档的数目,$\bar{x}_p$ 和 $\bar{x}_q$ 依次表示它们各自重心的坐标.文档特征向量的值即为该文档在多维向量空间中的坐标值.

我们用某个文档和它所归属的文档类之间的连接强度表示类的内聚性.

定义 **7.** 连接强度.类内连接强度表示类内文档之间的相似程度.设 C 是文档类,文档 $d\in C$,d 和 C 之间的连接强度定义为

$$\vartheta(d,C)=\sum_{j=1}^{\|r\|}\lambda_{d,j}\Big/\sum_{i=1}^{\|C\|}\sum_{j=1}^{\|r\|}\lambda_{i,j}. \tag{9}$$

由上述定义可知,连接强度实际上反映了某文档对其所属类贡献的大小.贡献小于指定阈值的文档自然应该从所属类中除去.

算法 **2.** 文档类确认算法

Input: $\Re$: The set of initial Web document clusters;

ϕ: Threshold of coupling;

φ: Threshold of cohesion.

Output: $\Re$, the validated Web document clusters.

Function: Validating the initial Web document clusters $\Re$.

(1) for (i:=1; $i \le \|\Re\|-1$; i++) do begin // merge relevant clusters
(2) for (j:=2; $j \le \|\Re\|$; j++) do begin
(3) $D^2_{i,j}$:=compu-dist($C_i \in \Re, C_j \in \Re$);
(4) if $D^2_{i,j} \le \phi$ then $C'_i := \cup C_j$;
(5) end for
(6) delete C_i and all $C_j \subset C'_i$ from $\Re$;
(7) $C'_i := C'_i \cup C_i$
(8) end for
(9) append $\bigcup_i C'_i$ to $\Re$;
(10) forall cluster $C_i \subset \Re$ do begin // eliminate documents from relevant
(11) forall document $d \in C_i$ do begin // clusters and form new cluster
(12) if $\upsilon(d, C_i) \le \varphi$ then do begin
(13) $C'_i := \cup d$;
(14) delete d from C_i;
(15) end if
(16) end for
(17) end for
(18) append $\bigcup_i C'_i$ to $\Re$;

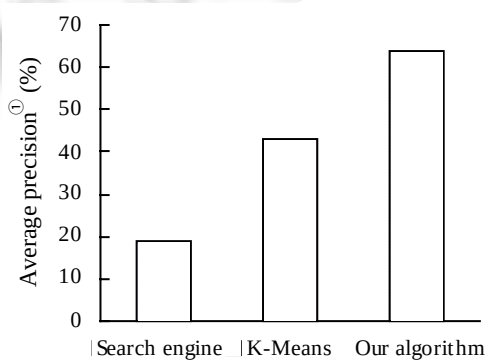
文档类确认算法分为合并和剔除两个过程.合并过程由一个双重循环组成,其时间复杂度为 $O(\|\Re\| \times (\|\Re\|-1))$,其中$\Re$是关联规则挖掘算法产生的初始文档类的个数.剔除过程同样是一个双重循环,其时间复杂度为 $O(\|\Re\| \times \sum_i NoD_i)$,其中 NoD_i 是第 i 个文档类 C_i 中文档的数量.由于 $\|\Re\| \times \sum_i NoD_i$ 和 $\|\Re\|^2$ 都远大于 $\|\Re\|$,因此,本算法的时间复杂度为 $O(\|\Re\| \times (\sum_i NoD_i + \|\Re\|))$.

3 实验结果

为了对本文提出的算法进行评价,我们将它和搜索引擎 Yahoo 以及 K-均值聚类算法进行了比较.实验中采用的 Web 文档集是用搜索引擎 Yahoo 从 Internet 上搜索得到的.整个实验在 P-II 450 计算机的 Windows 98 平台上进行.

首先测试算法的准确性.我们用 Yahoo 根据不同的主题进行了 50 次搜索,下载每次搜索到的前 20 个文档构成由 Yahoo 产生的 50 个文档类,每个类中包含有 1 000 个文档;再用人工方法剔除无关文档,同样将它们分成 50 个类,并依此作为分类准确性的基准;然后分别采用本文算法和 K-均值算法对这个文档集进行聚类.

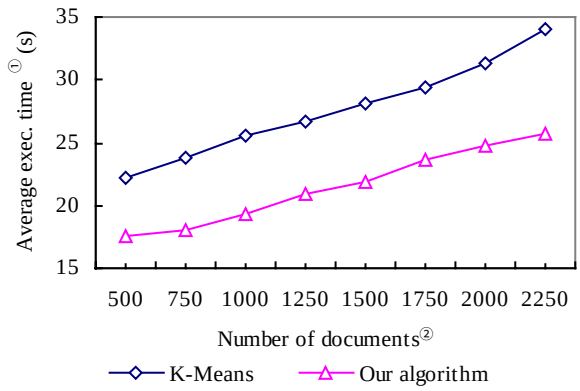
由于不同的聚类算法产生的类的数目很可能不同,为了使比较更趋公平起见,我们选用各自质量最好的 40 个类进行比较.图 1 即为由不同算法产生的 40 个类的平均精度的对比情况.由于本文的算法允许类间重叠,并采用了类确认技术,因此,平均聚类精度最高.



①平均精度,②算法.

Fig.1 Precision comparison of different clustering algorithms
图 1 不同聚类算法的精度对比

然后测试算法的扩展性.我们同样用 Yahoo 在 Internet 上搜索前面 50 个主题的相关文档,但是下载的是每个主题的前 10 个文档,并以 5 的增幅逐步递增到前 45 个文档,形成文档数量依次为 500、750、1 000、1 250、1 500、1 750、2 000 和 2 250 的 8 个文档集;然后分别采用本文算法和 K-均值算法对这 8 个文档集进行聚类,并计算它们的平均聚类时间,得到如图 2 所示的结果.图 2 表明,随着文档数的增多,两种算法的平均执行时间都在增加,但是本文的算法平均执行时间的增幅较小,增长趋势较为缓慢,说明本文的算法的扩展性较好.其原因在于,本文的算法用主题表示文档,降低了文档特征向量的维数;同时,又以主题为事务项,减少了数据处理的工作量.



①平均执行时间,②文档数.

Fig.2 Scalability comparison of different clustering algorithms

图 2 不同聚类算法的扩展性对比

综上所述,本文的算法优于 K-均值算法.

4 结 语

Web 文档聚类在信息检索、自然语言处理和电子商务等领域有着广泛的应用.

基于关联规则中的频集发现方法,本文提出了一种 Web 文档的聚类算法.该算法以文档为事务,以主题为事务项,适合处理高维数据,具有较好的扩展性;同时,也能适应文档类间固有的相互重叠情况.实验结果表明,该算法是有效的,具有一定的实用价值.

References:

[1] Broder, A.Z., Glassman, S.C., Manasse, M.S. Syntactic clustering of the Web. Technical Report, 1997-015, Palo Alto, CA: Digital Systems Research Center (Digital), 1997.

[2] Chang, C.H., Hsu, C.C. Customizable multi-engine search tool with clustering. Computer Network and ISDN Systems, 1997, 29(8-13):1217~1224.

[3] Chen, L., Katya, S. Webmate: a personal agent browsing and searching. In: Sycara, K.P., Wooldridge, M., eds. Proceedings of the 2nd International Conference on Autonomous Agents. New York: ACM Press, 1998. 132~139.

[4] Ron, W., Bienvenido, V., Mark, A.S., *et al*. Hypursuit: a hierarchical network search engine that exploits content-link hypertext clustering. In: ACM, ed. Proceedings of the 7th ACM Conference on Hypertext. New York: ACM Press, 1996. 180~193.

[5] Ackerman, M., Billsus, D., Gaffney, S., *et al*. Learning probabilistic user profiles. AI Magazine, 1997,18(2):47~56.

[6] Cheeseman, P., Stutz, J. Bayesian classification (autoclass): theory and results. In: Fayyad, U.M., Piatetsky-Shapiro, G., Smyth, P., *et al*., eds. Advances in Knowledge Discovery and Data Mining. Menlo Park, CA: AAAI/MIT Press, 1996. 153~180.

[7] Agrawal, R., Srikant, R. Fast algorithm for mining association rules. In: Jorge, B.B, Matthias, J., Carlo, Z., eds. Proceedings of the 20th International Conference on Very Large Databases. Santiago: Morgan Kaufmann Publishers, Inc., 1994. 487~499.

A Web Document Clustering Algorithm Based on Association Rule*

SONG Qin-bao, SHEN Jun-yi

(*Department of Computer Science and Technology, Xi'an Jiaotong University, Xi'an* 710049, *China*)

E-mail: qbsong@mail.xjtu.edu.cn; jyshen@mail.xjtu.edu.cn

http://www.xjtu.edu.cn

Abstract: By grouping similar Web documents into clusters, the search space can be reduced, the search accelerated, and its precision improved. In this paper, a new clustering algorithm is introduced. In the clustering technique, topics are represented according to VSM (vector space model), documents are represented according to topics, and the relation between documents and topics is viewed in a transactional form, each document corresponds to a transaction and each topic corresponds to an item. A frequent item sets can be found by using the association rules discovery algorithm, corresponding documents can be seen as initial clusters. These clusters are merged according to the distance between clusters, or divided according to the strength of connection among documents of a cluster. By real Web documents, experimental results show the algorithm's effectiveness and suitability for tackling the overlapping clusters inhered by documents.

Key words: document clustering; association rule; Web mining; WWW

* Received April 4, 2000; accepted August 28, 2000

Supported by the National Natural Science Foundation of China under Grant No.60173058; the Youth Foundation of the National High Technology Development 863 Program of China under Grant No.863-306-QN2000-5