重复模式识别算法及在Web信息抽取和聚类分析中的应用

木妮娜·玉素甫[1] 古丽娜·玉素甫[2,3]
(新疆师范大学计算机科学技术学院 乌鲁木齐 830054)[1]
(西北师范大学教育技术学院 兰州 730070)[2] (新疆师范大学教育科学学院 乌鲁木齐 830054)[3]

摘 要 序列中的重复模式识别算法及应用研究是数据挖掘领域的重要问题,是提取序列中有用信息的主要手段之一。近年来,针对各种重复模式定义、有效的识别算法设计以及重复模式识别算法在有关领域中的应用有了很多研究成果。文中对序列中重复模式的类型与特点作了描述,讨论了识别算法中常用的数据结构,以分类的方式重点回顾并总结了近年来重复模式在一些相关领域中的应用及相关算法的设计思路与技巧,并从加入的领域知识及约束、识别结果与算法扩充性、存在的主要问题等方面进行了讨论,其中包括在网络信息抽取、Web文档特征提取与聚类算法及相关的维文信息处理等领域中的应用。最后,讨论了关于序列重复模式识别算法在各个相关领域中的应用研究所面临的挑战,并探讨了未来的研究方向。

关键词 重复模式,Web文档特征,网络信息抽取,聚类算法,维文信息处理
中图法分类号 TP391 **文献标识码** A

Repetitive Pattern Recognition Algorithms and Applications in Web Information Extraction and Clustering Analysis

Munina YUSUFU[1] Gulina YUSUFU[2,3]
(School of Computer Science and Technology, Xinjiang Normal University, Urumqi 830054, China)[1]
(School of Education Technology, Northwest Normal University, Lanzhou 730070, China)[2]
(School of Education Science, Xinjiang Normal University, Urumqi 830054, China)[3]

Abstract Detection of repetitive patterns in sequences and their applications have become fundamental research areas in data mining. It is a very important means for extracting useful information from sequences. In recent years, many works have been conducted focusing on the definitions of repetitive patterns, efficient recognition algorithm designs, and applications in the relevant areas. In this paper, the classification and characteristics of repetitive patterns in sequence were briefly described, and the commonly used data structures in the algorithms were discussed. Recent studies on the applications of detection algorithms in relevant fields and their main design ideas were reviewed by discussing and evaluating certain aspects. These aspects include the field knowledge and constraints, identifying results, scalabilities of algorithms and existing main problems. Application areas include Web information extraction, Web document feature extraction and clustering algorithms, and related Uyghur language information processing. Finally, some challenges on detection algorithms of repetitive patterns in sequences and applications in various fields were discussed and future research trends were also explored.

Keywords Repetitive patterns, Web document features, Web information extraction, Clustering algorithms, Uyghur information processing

1 引言

序列中重复模式的识别与提取算法的研究涉及到很多相关的计算机学科领域知识,如组合数学、图论、自动机理论及机器学习等,具有重要的理论研究价值,并在数据挖掘、模式识别、数据压缩、生物信息学、自然语言处理等领域中具有极其广泛的实际应用。

Internet的迅速发展、Web的全球普及,以及生物信息的剧烈膨胀,序列数据库的规模急剧增大,这对算法的空间与时间复杂度及识别精度都提出了更高的要求;人们对基于Web的服务需求也在不断增加,不同应用领域可能对序列重复模式有不同的定义,需要加入领域知识或约束,从而使得识别结果更有意义。这些都给高效的重复模式的识别与提取算法的研究及应用带来了新的挑战。

本文首先对序列中重复模式的类型与特点作了简要描述,并讨论了识别算法中的常用数据结构;接着以分类的方式重点回顾并总结了近年来重复模式在一些相关领域中的应用及相关算法的设计思路与技巧,其中包括网络信息抽取、Web

本文受国家自然科学基金(61263044),新疆维吾尔自治区2015年双语教育研究项目(SY20153136)资助。
木妮娜·玉素甫(1963—),女,博士,副教授,主要研究方向为数据挖掘、模式识别、自然语言处理;古丽娜·玉素甫(1974—),女,博士生,副教授,主要研究方向为现代教育技术原理、现代远程教育、民族教育信息化,E-mail:gulina31@126.com(通信作者)。

文档特征提取与聚类算法及相关的维文信息处理等领域中的应用，主要从不同应用领域中序列重复模式的不同定义、加入的领域知识及约束、采用的主要数据结构及设计方法、识别结果与算法扩充性及存在的主要问题等方面进行了讨论与评价（由于篇幅所限，没有对识别算法本身进行深入讨论）；最后，讨论了关于序列重复模式识别算法在各个相关领域中的应用研究所面临的挑战，并探讨了未来的研究方向。

2 序列中重复模式的类型与特点及识别算法中的常用数据结构

2.1 序列中重复模式的类型与基本特点

序列数据是一种重要的数据类型，在许多应用领域普遍存在。根据应用领域的不同，序列数据可以分为不同的种类。因为事件序列、时间序列和数据流属于比较独立的研究领域，本文只对字符序列，如文本、生物序列及 Web 文档（清除音频及图形图像后的文档）等序列中，重复模式识别算法应用的各个主要方面进行探讨。

在字符序列中重复出现的字符串或基因片段等，称为重复模式（Repeat，Repetitive Pattern）。序列中的重复模式又称为重复片段、重复子串、重复序列、频繁模式（Frequent Pattern，重复模式＋设定阈值）。在蕴含着重大意义的生物序列（如 DNA 序列、蛋白质序列等）中存在着大量的重复片段，例如，在人类基因组的约 3.2×109 个碱基对中超过 50％已被识别为各种重复元素[1]，包括简单碱基重复和大规模的序列复制在内的相似部分。随着越来越多的生物基因组被破译，科学家已充分证明重复序列在进化过程中可以用于帮助形成新基因[2]。很多人类疾病都是由重复序列的突变所引起[3]。从海量基因数据中发现频繁出现的基因片段对于生物序列的特征分析与研究非常重要，对研究生物进化、物种相关性等有重要意义[1,4-5]。

在计算语言学统计文献中的高频字串（频繁模式）有很重要的意义，可应用于中文处理的分词、未登录词识别、新词检测、术语提取等研究领域，而且现在网络技术和电子商务的不断发展，对这方面的需求正变得日益强烈。基于重复模式识别的算法与技术，可广泛地应用于 Web 内容信息及论坛信息抽取、重复网页检测、文档聚类与分类、社会计算的热点发现等多种应用研究中。

重复模式的自动检测并非易事。查找序列中的重复模式，首先要形式化地定义重复模式，而后是查找满足定义的重复模式。重复模式定义的不同将导致查询结果不同，查询的时间与空间复杂度也会随之变化。序列中的重复模式可大致分为精确重复模式与相似重复模式等。根据定义的不同，精确重复模式（repeat）可以分为很多类型。文献[6]对精确重复模式的类型及识别算法中的常用数据结构作了综述。

不失一般性，假设∑是一个有穷字符集，S 是一个由∑中的字符连接而成的长度为 n 的字符串（文本），设 u 为 S 的一个子串，如果至少存在两个位置 i,j，使得 $S[i..i+r]=S[j..j+r]=u$，其中 $0\leqslant r\leqslant n-1$，则称 u 为 S 的一个重复子串。精确重复模式可定义为一个多元组：

$$R_{S,u}=(p;i_1,i_2,\cdots,i_e)$$

其中，p 为 $R_{S,u}$ 的长度；e 为 $R_{S,u}$ 的指数；u 为 $R_{S,u}$ 的生成元，也称为 S 的重复子串，$i_1,i_2,\cdots,i_e$ 为重复子串的起始位置。对于事先设定的阈值 λ，如果 u 的出现频次大于 λ，则称 u 为 S 的频繁模式；$R_{S,u}$ 根据上下文有时可简记为 R。

重复模式具有反单调特性，即如果 R 是重复模式，则其任何子模式也都是重复模式；反之，如果模式 P 的任何一个子模式都不是重复模式，则 P 就不是重复模式。此性质与频繁项集的 Apriori 性质相似，即：频繁项目集的子集是频繁项目集；非频繁项目集的超集是非频繁项目集。此性质在进行重复模式挖掘过程中发挥着重要作用。由于重复序列的反单调特性，在求重复模式时可以只求出最大的重复模式，这样可以用最少的重复模式表示整个重复模式集。

由此引出最大重复模式（Maximal repeat）的定义：设 $R_{S,u}=(p;i_1,i_2,\cdots,i_e)$ 是一个 repeat，若至少存在一对 s,t（$1\leqslant s<t\leqslant e$），使得 S 的第（i_s-1）个字符和第（i_t-1）个字符不相同，此时称 $R_{S,u}$ 为左最大化的（Left Maximal）；若至少存在一对 s,t（$1\leqslant s<t\leqslant e$），使得 S 的第（i_s+p）个字符和第（i_t+p）个字符不相同，此时称 $R_{S,u}$ 为右最大化的（Right Maximal）；若 $R_{S,u}$ 既是左最大化的，又是右最大化的，则称 $R_{S,u}$ 为最大重复模式。图 1 给出了一些精确重复模式实例。

	1	2	3	4	5	6	7	8
$S=$	a	b	a	a	b	a	a	b

图 1 序列 S=abaabaab

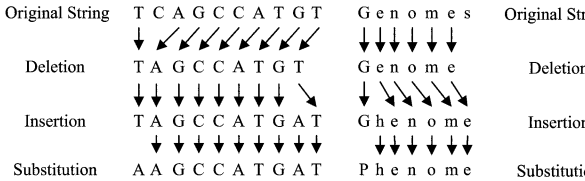
在图 1 中，序列 S 中有十多个各种重复模式。其中，$R_{S,aab}=(3;3,6)$ 为串联重复模式；$R_{S,baab}=(4;2,5)$ 为重叠重复模式；$R_{S,ab}=(2;1,4,7)$ 为散在重复模式，各重复子串出现的间距均为 1。$R_{S,a}$，$R_{S,ab}$，$R_{S,abaab}$ 为 S 中的最大重复模式。

具有间隙约束的频繁模式更具灵活性和有效性，在很多领域都有广泛应用。在图 2 中，u=ACG 是多序列 S_1,S_2,S_3 中的重复子串，其中各子串间间隙 g_i 满足条件：$dmin_i\leqslant g_i\leqslant dmax_i$。如果设定 $dmax_i=5$，则 S_3 中的重复子串 u=ACG 不符合查询条件。

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
$S_1=$	A	C	G	T	A	C	G	A	C	G	T	G	C	A	C	G	A	C	T	A	A
$S_2=$	A	C	T	A	C	G	T	G	A	C	G	C	C	T	C	A	A	C	G	T	G
$S_3=$	G	A	C	C	G	A	C	G	G	C	T	C	G	T	A	C	G	C	C	T	A

图 2 在多序列 S_1,S_2,S_3 中查找重复模式

序列中的相似重复模式，可视为由各种模式的精确重复片段通过一定数量的字符插入（insertion）、删减（deletion）和替换（substitution）操作而获得。近似片段转换为精确重复所需的（最少）符号操作被称为序列间的编辑距离（edit distance），也被称为 Levenshtein 距离，通常用于确定两个序列间的相似度。图 3 中给出了编辑距离的一个实例，相似重复序列 TCAGCCATGT 与 AAGCCATGAT 的编辑距离为 3，相似重复模式 Genomes 与 Phenome 的编辑距离为 3。



注：左边的一个 DNA 序列 TCAGCCATGT 在 3 个编辑操作后转换为一个相似的 DNA 序列 AAGCCATGAT；右边的一个文本序列（单词）Genomes 在 3 个编辑操作后转换为一个相似的文本序列（单词）Phenome

图 3 编辑距离操作的实例

此外,还有一些常见的序列距离函数应用于相似重复模式识别的不同领域中,如海明距离(Hamming distance)、频率距离(frequency distance)、频率变换距离(frequency transformation distance)、q-gram 距离等。

2.2　序列中重复模式识别算法中的常用数据结构

一般来说,序列中频繁模式的查找问题包括查找已知的特定模式、内容未知但长度给定的模式以及所有不定长的模式。对于已知的特定模式,可以采用经典的串匹配算法如 KMP 算法查找其出现的所有位置;对于内容未知但长度给定的模式串,则可以采用指纹(fingerprint)等哈希方法统计所有指定长度的子串的频度,从而得到满足条件的重复串;第三种是事先并不知道其内容和长度的无先验信息的模式,对于此类问题,文献中有许多种解决方案,其中基于后缀索引的方法是最常用、效率较高的频繁模式发现方法。后缀索引的有效实现方式有后缀树(Suffix Tree)及其变种。

后缀树算法支持有效的字符串匹配和各种重复模式的查询,例如发现重复子串、相似字符串匹配、文本压缩和文档聚类等,处理速度很快。图 4 给出了序列 abaabaab 的后缀树示例,黑色粗体标示一个后缀子串在后缀树中的表示,即对于任何一个叶子 i,从根到该叶子的整个路径上的边标签串联起来的内容就是 S 从位置 i 起的后缀子串,即 $S[i\cdots n]$;在此后缀树中黑色粗体标示了发生位置 5 和位置 2 的长度为 4 的重复模式 baab 及路径,即后缀 $S[5\cdots8]$和后缀 $S[2\cdots5]$的最长公共前缀为 baab。Ukkonen[7] 提出了改进的线性时间建树算法,它易于理解,而且具有在线特性(on-line),即按照从左到右顺序处理字符串,因此在应用中被广泛接受。

一旦构建后缀树,即可利用它来高效地解决很多串处理方面的问题。但当 n 很大时,构建后缀树对空间的消耗成为瓶颈。因此在对构建后缀树算法进行优化的同时,研究者也在寻找比后缀树占用更小空间的后缀结构,其中 Abouelhoda 等人[8] 提出的 Enhanced Suffix Arrays(增强后缀数组)、Ferragina 等人[9] 提出的 String B-Tree、Manber 等人[10] 提出的 Suffix Array(后缀数组)都是较有影响力的数据结构。

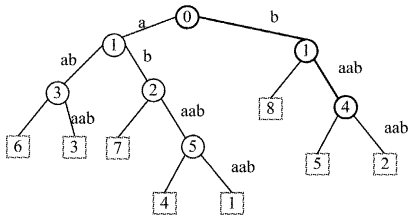


图 4

后缀数组是一种全文索引数据结构,相对于后缀树具有几个数量级的空间效率优势,并且时空复杂度与字符集大小无关,适合于中文文档的处理。研究者对后缀数组算法的研究热情从未停止,一些不同的线性算法相继出现[11-12],而应用其作为主要数据结构的识别重复模式的算法[13-16] 也在时空效率上得到了提高。

近年来,一些有效的索引数据结构相继研究成功,并被应用于序列中重复模式的识别与挖掘中,其中包括压缩后缀数组(Compressed Suffix Array)、LCP(Longest Common Prefix)[17] 数组、LPF(Longest Previous Factor)数组、后继数组(SUA)[18] 等。此外,后缀 trie 树、倒排索引散列表、Pat 数组、QSA(Quasi Suffix Array)数组[19]、BWT(Burrows-Wheeler Transform)数组[20] 等也被广泛应用于重复模式的识别算法中。图 5 给出了用链接符号 #1,#2,#3 连接 3 个文本 bbabab,abacac,bbaaa 后的新文本 t 的 LCP 数组和 SA 数组值,并形象地展示了 LCP 数组与 3 个文本中的重复模式的关系。

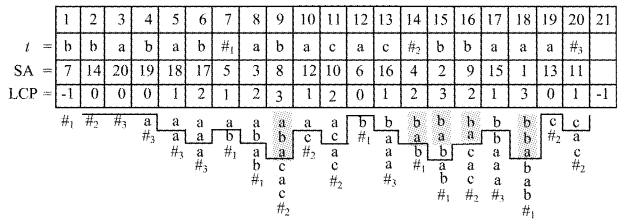


图 5

3　重复模式识别算法与技术在多个相关领域中的应用

3.1　在 Web 信息抽取中的应用

近年来,研究者利用重复模式挖掘与定位算法进行了网络信息抽取技术和 Web 搜索引擎建立方面的研究。传统的 Web 信息抽取通常采用机器学习的方式,构建专门的包装器(wrapper)来识别数据并将其转化为适当的格式,如 XML 和关联式表格等。人工创建的包装器有很多众所周知的缺点,主要是耗时而繁琐,难以维护。一些自动化的 Web 信息抽取方法越来越得到重视。网页中的有用信息往往位于具有特定排列方式和次序的结构中,特别是由搜索引擎产生的搜索结果通常是有规律的重复模式。因此,抽取重复模式可以发现对包装器有用的抽取规则。根据这一特征,高强等[21] 给出了一种基于重复模式自动抽取 Web 内容信息的方法,即以目标网页的 HTML tag 序列构造对应的后缀树,借助后缀树分析挖掘页面结构中所包含的重复模式,但其中有不符合要求的,因此建立了启发式规则(包括实例数量、关键 tag、模式长度、消除重叠、覆盖率等)对候选模式进行过滤和精化,选取出合理有效的重复模式,并从其实例中抽取网页数据记录的信息。

Chia-Hui Chang 等[22] 提出的 IEPAD 是较早的从未标记的网页中提取模式的系统,这种方法利用了以下事实:如果一个 Web 页面中包含多个(均一的)需要提取的数据记录,则为了良好的可视化效果,它们通常都是使用同一模板表示的。因此如果页面设计良好,则重复模式就较易被发现。在实现时,IEPAD 使用 PAT 树结构对网页中的最大(散在和串联)重复模式进行挖掘。图 6 给出文中抽取规则生成器(Extraction Rule Generator)的描述。因为这样的数据结构中只记录与后缀的精确匹配,IEPAD 进一步应用星比对算法(center star algorithm)对重复模式进行扩展,以更好地理解所有的记录项。

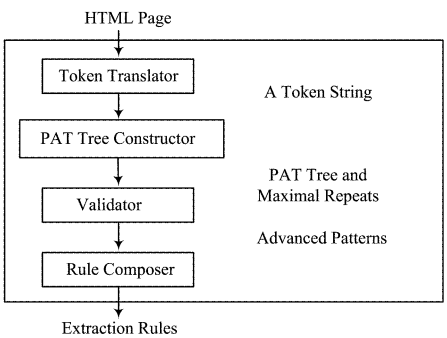


图 6　抽取规则生成器(Extraction Rule Generator)[22]

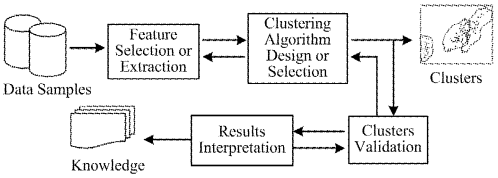
图 6 中的 Validator(验证器)用于将 PAT Tree 中发现的 Maximal Repeats 进行过滤,以找出有用的模式。除了重复模式的长度与频率信息,还使用了以下几个规则:1)一个最大重复模式 α 的规则性(regularity)通过计算两个相邻出现的重复模式之间的间隔即($p_{i+1}-p_i$)的标准偏差来测量;2)紧致性(compactness)是最大重复模式的密度量值,它可以用于消除远离给定阈值的重复模式,密度被定义为$(k-1*|\alpha|/\{p_k-p_1\}$,其中$|\alpha|$为 α 的长度;3)覆盖率(coverage)测量最大重复模式中的内容量,假设函数 $p(i)$以字节数返回输入网页中第 i 个子串的位置,则覆盖率被定义为$[p(p_k+|<|)-p(p_1)]/|$网页$|$,其中$|$网页$|$是输入网页的字节数。通过这些规则,有效地过滤并选取出了有效的重复模式。DELTA[23]也在记录级别上工作以检测重复模式,但是仅解析 HTML 标签字符串,从而验证重复记录标签在 HTML 标签树中具有相同的父标签。

网络论坛的信息抽取不同于一般网页的信息抽取。在网络论坛中,对于同一类型的页面,论坛系统采用同一模板生成,相同类型的页面结构非常相似。针对现有网络论坛信息抽取的不足,韩普等[24]提出了一种基于重复模式发现算法的论坛信息抽取方法。该方法通过自动定位重复模式进行论坛信息的抽取,较好地解决了在论坛信息抽取过程中需要人工查找、定位重复模式或者通过人工分析论坛页面代码定制抽取规则的问题。其对于清华大学网络学堂课程论坛的抽取准确率都达到 100%,而其他两个社会性质的论坛召回率和准确率稍低。高克宁等[25]通过对传统重复模式表示法的分析,提出了基于重复模式的 Web 信息语义表示法。该方法在形式化描述重复模式的基础上,抽取 Web 信息中的重复模式来建立表达 Web 信息语义特征的相关矩阵,并通过 γ 相似匹配算法计算重复模式的权重,继而进行 Web 信息分类。对 70 个网站共 130 个页面进行实验后的查准率和查全率分别为 85%和 84%(取最少实例数量 $MinIns=3$,$MinLen=7$ 时)。

3.2 在 Web 文档特征提取与聚类算法中的应用

随着 Internet 的迅猛发展,Web 已经成为全球传播与共享科研、教育、商业和社会信息等最重要和最具潜力的巨大信息源。如何有效地组织 Web 文档,并对互联网的信息进行有效分析,便捷、准确地挖掘出需要的信息知识,是目前数据挖掘领域研究的一个重要问题。聚类分析是一种组织文档的有效方法,它把一组无标签的文档按照内容的相似性归成若干类别。文档聚类分析被广泛研究并应用于信息检索、Web 挖掘[26]等相关领域。近年来,随着话题发现与跟踪(Topic Detection and Tracking,TDT)技术[27-28]的发展,借助聚类分析,可以从大规模文档中的大量无标签的信息中发现未知话题,并以可读的形式向用户报告。由此,文本聚类也成为网络热点话题发现的重要技术[29]。

文本聚类也常常被称为无指导的机器学习,其目标是将未知的文本集自动形成聚簇(clusters),使得簇内文本之间的相似程度很高,而不同簇间的文本之间的相似程度尽可能地低。Rui Xu 和 Donald Wunsch II[30]提出的聚类分析过程主要包括:1)特征选择或提取;2)聚类算法设计或选择;3)聚类验证;4)结果解释。4 个步骤有一个反馈途径,它们彼此密切相关,并影响着计算的簇,如图 7 所示。从图中可以看出,聚类分析时,文档标引和特征选择或特征提取是基础和关键性步骤。文本聚类的首要工作是文档标引,即将文本从无结构或者半结构化的原始形式转化为计算机能够理解的表示模型。向量空间模型(Vector Space Model,VSM)仍是文本特征表示的主要方法,采用的是 BOW(Bag of Words)方法,将文档表示为词及其在该文档中出现频度的一个向量。BOW 虽然具有直观且易于实现等优点,但是忽略了词出现的词法、语法及实际上下文语景,还导致了文本聚类的"高维诅咒"问题。随着文本数量的急剧增长和数据维度的不断增大,这个问题变得更加严重。最基本且最有效的改进应该是选择文本表达能力较强的特征单元作为文本特征项,以提高对文本的表达能力。特征单元不同,则特征空间不同,其中文本向量的分布也会完全不同,可以说特征单元的选择从根本上影响着整个文本聚类的效果。



注:典型的聚类分析包括具有反馈路径的 4 个步骤,这些步骤彼此密切相关,并影响着计算得到的聚簇[30]

图 7 聚类过程

文档集合中谈论同一个话题的文档往往会包含许多共同的或相似的词语,因此同一个主题的文本中会出现大量相似的重复短语,这些重复短语正是构成文本之间相似性的重要因素。它可以描述属于这些文档的共同属性,并可以作为文档特征。与其他聚类算法相比,基于重复模式的聚类分析可显著降低文档数据的维度,同时可以为聚类结果提供简洁明了的类别标签[31]。同时,根据 Web 页面中出现的重复信息对 Web 页所体现的语义进行表示,可以提高 Web 页分类正确的精度。对维吾尔语的聚类分析而言,重复短语的识别就更加重要。与英文类似,维吾尔语是一种拼音文字,词与词之间也是以空格隔开。因而在处理维吾尔文时,包括在文本聚类及分类研究中,通常以空格作为自然分隔符来获取词的集合,而不考虑分词的问题。但是,英文中的一个单词在维语中经常对应一个短语,例如,单词 Software(软件)在维语中是"يۇمشاق دىتال",它是由两个词组成的词组(短语)。如果采用 BOW 方法用单词作为特征项进行聚类分析,则会出现يۇمشاق(软的)和دىتال(组件)的特征项,进而影响聚类效果。如果运用最大重复模式识别算法,则会提取到较为完整的短语项,结合其他规则与方法,如互信息计算等,可得到并利用类别区分能力较强的完整的短语作为特征项进行聚类分析。

较早的基于抽取重复模式的聚类分析算法是利用后缀树实现的。Oren Zamir 等[32]提出了后缀树聚类算法(Suffix Tree Clustering,STC)对 Web 文档进行聚类,主要用于解决 Web 搜索引擎一次查询返回文档太多的问题,是一种增量式的线性算法。与基于向量空间模型的 KMeans[33],Single-Pass[34]和 SOM[35]不同,STC 算法利用后缀树来有效发现文本所共同含有的短语信息(即公共子串),并进而利用这些信息来构建基类。为了避免出现大量重复的或非常相似的类别,STC 合并那些高度重叠的基类。Oren Zamir 等[36]对此进行了对比实验,结果证明与基于向量空间模型的算法相比,基

于短语的 STC 算法有效地考虑了词与词的关系,产生了比较好的聚类效果。但是,STC 算法依然存在不足,这些不足主要表现在时空复杂度以及信息的抽取和利用等方面。由于 STC 不是一种基于主题的聚类方法,因此它无法保证一个类别中包含共同短语信息的文本都是关于同一主题的。近年来,针对 STC 算法的不足,学者们提出了许多 STC 的改进方法。针对后缀树所占空间较大,Janruang 等[37]在聚类过程中利用语义相似度提出了 SSTC(Semantic Suffix Tree Clustering)算法,采取了有效的启发式方法进行结果聚类,通过语义相似度减少了后缀节点和分支,改善了 STC 方法的性能。

由于后缀树的性能(时间和空间复杂度)是与语言字符集的大小相关的[7,32,36],而汉语比英语有大得多的字符集,因此后缀树并不适合用于中文文本的短语发现。同时,后缀树作为一种有效的英文文档特征抽取方法,并不太适合东方语言(例如中文),因为它要求被聚类的文档有明确的词界限,而中文不像英文那样有明确的分隔符(如空格或定界符等)。林建敏等[38]阐述了基于后缀树的文本聚类 STC 算法,对其所存在的缺陷进行了分析,并在此基础上提出了采用 PAT-array 和模糊聚类相结合的方法对其进行改进,以提高聚类的质量。翟献民等[39]针对后缀树聚类选取基类时基类短语出现信息不规范、重复和冗余的问题,提出了一种基于改进后缀树的维吾尔语文档的聚类分析算法。以短语互信息算法选出遵守维吾尔语语法规则的基类短语,并利用短语归并算法对选取的重复基类短语进行归并,最后利用短语去冗余算法对冗余的基类短语进行了处理。与传统 STC 相比,改进的后缀树聚类算法的全面率和准确率都得到了提高。

由 Zhang 和 Dong[40]于 2004 年提出的基于 SHOC 算法的 WICE 聚类系统,是用来处理中文查询的搜索结果的聚类系统。该系统使用 suffix Array 发现关键短语,对中文等字符集规模很大的语言也具有很高的效率。同时,SHOC 算法使用 SVD 方法来发现短语之间的语义,进而获得层次性分类结果。胡吉祥等[31]提出了一种基于重复串的特征提取方法,其可以从文本中提取有意义的特征,且对于中文无需分词。实验表明,该方法可以降低特征空间的维度,同时能有效改善传统以词为特征的聚类算法的性能。为了高效地获取所有子串的频度,引入了 Yamamoto 等[41]提出的子串类概念,即将全部子串归成至多$(2n-1)$个子串类(Classes of Substrings),只需计算这些子串类的频度就可得到全部$n(n-1)/2$个子串的频度信息。对于长度为 n 的文档,提取最大重复模式的时间复杂度为 $O(n\log n)$,空间复杂度为 $O(n)$。

文献[40]与文献[31]在识别最大重复模式时,使用了与文献[13]相似的算法,即,首先查找 $S=S[1]\cdots S[n-1]S[n]$ 的右最大化的重复串 R;再令 $\sim S$ 表示 S 的逆串,即 $\sim S=S[n]S[n-1]\cdots S[1]$,当 $\sim R$ 在 $\sim S$ 中是右最大化的时,则可推出 R 在 S 是左最大化的。因此,R 在 S 是最大化的。这些算法虽然是线性算法,但是在预处理阶段需要计算两次 suffix Array 和 LCP 数组的值,计算量是识别重复模式算法的数倍(取决于采用的 suffix Array 和 LCP 数组算法),因而实际运行效率会受到影响。文献[38]和文献[31]提出了关键重复串(有意义串)的概念,并用完整性、稳定性和独立性[31]来加以定义。其中,使用了最大重复模式定义完整性,使用互信息 MI(Mutual Information)定义稳定性,而独立性是以上文(left context)与下文(right context)的频率指标定义的。

Johannes Fischer 等[15]提出基于约束的字符串挖掘算法,可通过发现部分序列集中频繁出现而在另一部分序列集中极少出现的模式来实现聚类分析。由于它使用了后缀数组,因此克服了同类算法[42]由于使用后缀 trie 树而空间复杂度太高、实际使用率低的缺点。

一些聚类算法充分考虑了文本数据的具体特点,把频繁词语集合的概念用于文本聚类,用频繁项集取代距离函数作为聚类的标准。文献[43]中的 FTC 算法利用文档之间共同包含的频繁词语集合来衡量文档之间的聚合程度。Benjamin 等人[44]提出了基于频繁项的层次文本聚类算法,认为同一个类别中的文档应该比不同类别间的文档包含更多的相同频繁词语集合,应直接利用频繁词语集合来衡量类别之间的聚合程度。Guralink 等人[45]把序列投影到一个向量空间,其中向量属性集就是找到的频繁子序列模式,然后应用 k-means 的方法对这些向量进行聚类。由于此方法找到的频繁子模式一般很多,因此表征每条序列的向量是高维且稀疏的,这会影响聚类的质量。为了克服这些缺点,文献[46]也使用频繁子序列的基本方法,但只选用一条最能区分不同序列能力的频繁子序列来作为这条序列的特征,使得聚类效率和质量有所提高。

4　主要挑战及未来的研究方向

目前,各种新应用领域的序列重复模式识别算法与技术研究方兴未艾,研究成果与实际应用的需求之间还有一定差距,许多问题有待进一步解决,我们认为主要的挑战及未来的研究方向包含如下几个方面。

1)经过近几十年的发展,索引结构的时空效率研究取得了很多实质性的成果,并在序列重复模式识别算法中得到了广泛的应用,但仍然存在一些不足。比如,后缀数组不支持排序查询、近似匹配、压缩匹配等功能[47],其扩展性需要进一步完善;而后缀树的性能与语言字符集的大小相关,因此后缀树并不适合于中文文本的短语发现。后缀数组的动态更新也是一个问题,当有文本更新时,只能采用索引完全重建的方法实现索引的更新,既费时又影响检索的准确性[48]。如何实现后缀数组的动态更新与维护,也成为未来研究值得关注的一个方向。

2)各种应用领域序列数据的快速增长和高负载的 Web 数据挖掘与信息检索等要求,对序列中重复模式识别算法的时间和空间复杂度均提出了更高要求。因此,除了继续开发与研究快速高效的频繁模式识别算法来提高模式发现的准确性和时效性外,在线技术[49]、多重复模式匹配等方法与技术也值得进一步研究,并利用并行技术[50]、网格计算等技术来拓展算法的可实现性。

3)随着互联网的发展,互联网语料的规模也以指数规律增大,其规模已经远远超出一般服务器内存的大小;基因数据的碱基对数量每 12～16 个月就增加一倍,其规模也超出服务器内存的规模[51]。因而,针对大规模语料如何采取合适的策略将语料划分并获得原始语料的频繁模式,开发相对快速的识别算法[52-53],也显得十分迫切,其算法时间和空间复杂度的权衡也尤为重要,需要根据实际需求作出抉择。

4)如第 3 节所述,如何对 Web 信息进行有效分析,并有效地挖掘出需要的信息知识,是目前数据挖掘领域研究的一

个重要问题。Web 文本分类与聚类面临的一些特殊问题，如很高的维度、大量的文本、如何有效地进行文档标引和特征提取等，在维吾尔语 Web 文本处理中同样存在，加之维吾尔语是一种黏着性语言，其特点是有比较复杂的时态变化和非常丰富的形态结构，在进行 Web 文本分类与聚类时，其原始特征空间的维数更大，文本表示更加稀疏，词形变化更加灵活[39,54]。因此，如何利用重复模式算法进行有效的文本特征选择及空间降维表示，结合 Web 信息语义表示法进行维吾尔语 Web 文本的分类与聚类，并在热点话题发现等领域得到应用，是一个需要不断探索与创新的研究方向。

5)特定的问题往往对应着特定的领域知识。不同应用领域可能对序列的频繁模式识别加进领域知识或约束，从而使得模型的性能得到提高，并使得识别结果更有意义。比如，在检测中文术语时，提取语料中的重复模式后，针对模式的组成结构采用了词性规则及单字词概率等进行候选术语的检测和过滤[55]；在不同的应用领域，应用了左右熵、似然比、相关频率、上下文邻接分析[56]、位置成词概率等统计特征。在基于重复模式的 Web 文档聚类分类算法中，使用了文档反文档频率(TF-IDF)、信息增益(Information Gain，IG)、互信息等特征提取方法。但是，巨大的计算量、复杂的噪声模式、海量的时变数据给传统的统计分析也带来了巨大挑战，需要不断发展与研究更加灵活的数据分析与统计技术，从而使得现有的方法能适应这种面向应用领域的频繁模式识别与查询。因此，能有效适应领域约束的序列频繁模式识别算法也有待进一步研究。

结束语 序列中重复模式识别算法的研究及应用是数据挖掘领域的重要问题，它可以从海量数据中得到正常的和异常的行为模式，具有非常重要的理论意义和应用价值。从早期的精确重复模式识别，到现在的在自然语言处理中的各种应用，以及在网络信息抽取、Web 文档特征提取与聚类算法中的应用，无论是算法设计技术还是应用领域，都经历了重大的变化。本文对序列中重复模式的类型与基本特点及识别算法中的常用数据结构进行了讨论，以分类的方式重点回顾并总结了近年来重复模式在一些相关领域中的应用及相关算法设计思路与技巧及存在的问题，并进行了相应的讨论，对面临的主要挑战及未来的研究方向进行了探讨。

参 考 文 献

[1] LANDER E S, LINTON L M, BIRREN B, et al. Initial Sequencing and Analysis of the Human Genome[J]. Nature, 2001, 409(6822): 860-921.

[2] MAKALOWSKI W. Not junk after all[J]. Science, 2003, 300(5623): 1246-1247.

[3] Int'l Human Genome Sequencing Consortium. Finishing the euchromatic sequence of the human genome[J]. Nature, 2004, 431(7011): 931-945.

[4] SHAPIRO J A, STERNBERG R V. Why repetitive DNA is essential to genome function[J]. Biological Reviews, 2005, 80(2): 227.

[5] TREANGEN T J, SALZBERG S L. Repetitive DNA and next-generation sequencing: Computational challenges and solutions [J]. Nature Reviews Genetics, 2011, 13(1): 36-46.

[6] SMYTH W F, YUSUFU M. Computing Regularities in Strings [C]// Proceedings of the 2nd IEEE International Conference on Computer Science and Information Technology. 2009: 298-302.

[7] UKKONEN E. On-line Construction of Suffix Trees[J]. Algorithmica, 1995, 14(3): 249-260.

[8] ABOUELHODA M I, KURTZ S, OHLEBUSCH E. Replacing Suffix Trees with Enhanced Suffix Arrays[J]. Journal of Discrete Algorithms, 2004, 2(1): 53-86.

[9] FERRAGINA P, GROSSI R, MUTHUKRISHNAN S. The String B-Tree: a New Data Structure for String Search in External Memeory and Its Applications[J]. Journal of the ACM, 1999, 46(2): 236-280.

[10] MANBER U, MYERS G. Suffix Arrays: a New Method for On-line String Searches[J]. SIAM Journal on Computing, 1993, 22(5): 935-948.

[11] PUGLISI S J, SMYTH W F, TURPIN A H. A taxonomy of suffix array construction algorithms[J]. ACM Computing Surveys, 2007, 39(2): 1-35.

[12] NONG G, ZHANG S, CHAN W H. Two Efficient Algorithms for Linear Time Suffix Array Construction[J]. IEEE Transactions on Computers, 2011, 60(10): 1471-1484.

[13] FRANEK F, SMYTH W F, TANG Y D. Computing all repeats using suffix arrays[J]. Automata, Languages and Combinatorics, 2003, 8(4): 579-591.

[14] PUGLISI S J, SMYTH W F, YUSUFU M. Fast Optimal Algorithms for Computing all the Repeats in a String[J]. Mathematics in Computer Science, 2010, 3(4): 373-389.

[15] FISCHER J, HEUN V, KRAMER S. Fast Frequent String Mining Using Suffix Arrays[C]// Proceedings of the Fifth IEEE International Conference on Data Mining. 2005: 609-612.

[16] ILIE L, SMYTH W F. Minimum Unique Substrings and Maximum Repeats[J]. Fundamenta Informaticae, 2011, 110(1-4): 183-195.

[17] PUGLISH S J, TURPIN A. Space-time tradeoffs for longest-common prefix array computation[C]// Proceedings of 19th International Symposium on Algorithms and Computation. 2008: 124-135.

[18] 王镝，王国仁，吴青泉，等. DNA 序列中基于后继数组索引的 LPR 查找算法[J]. 计算机研究与发展，2006，43(z3)：195-199.

[19] 木妮娜·玉素甫，古丽娜·玉素甫，张海军. 基于 QSA 数组计算序列中所有 NE 重复模式的算法[J]. 计算机科学，2014，41(3)：249-252，262.

[20] KULEKCI M O, VITTER J S, XU B J. Efficient maximal repeat finding using the burrows-wheeler transform and wavelet tree [J]. IEEE/ACM Transactions on Computational Biology and Bioinformatics, 2012, 9(2): 421-429.

[21] 高强，张敬之，耿桦，等. 基于重复模式的 Web 信息抽取[J]. 计算机科学，2007，34(4)：210-212，221.

[22] CHANG C H, LUI S C. IEPAD: Information extraction based on pattern discovery[C]// Proceedings of the 10th International Conference on World Wide Web. 2001: 223-231.

[23] MUROLO A, NORRIE M C. Revisiting Web Data Extraction Using In-Browser Structural Analysis and Visual Cues in Modern Web Designs[C]// Web Engineering, Lecture Notes in Computer Science. 2016: 114-131.

[24] 韩普，王泽. 基于重复模式的论坛信息抽取研究[J]. 南京师范大学学报(工程技术版)，2010，10(3)：74-77.

[25] 高克宁,马安香,张斌,等. 基于重复模式的 Web 信息语义表示方法的研究[J]. 小型微型计算机系统,2009,30(1):26-30.

[26] ZAMIR O, ETZIONI O, MADANI O, et al. Fast and intuitive clustering of Web documents[C]// Proceedings of the KDD'97. 1997:287-290.

[27] ALLAN J, et al. Topic Detection and Tracking: Event-Based Information Organization[D]. Dordrecht: Kluwer Academic Publishers, 2002.

[28] HASHIMOTOA K, KONTONATSIOSB G, MIWAC M, et al. Topic detection using paragraph vectors to support active learning in systematic reviews[J]. Journal of Biomedical Informatics, 2016, 62:59-65.

[29] FRANZ M, MC CARLEY J S. Unsupervised and supervised clustering for topic tracking[C]// Proceedings of the 24th Annual International ACM SIGIR. New Orleans, Louisiana, USA: ACM, 2001:310-317.

[30] XU R, WUNSCH D. Survey of Clustering Algorithms[J]. IEEE Transactions on Neural Networks, 2005, 16(3):645-678.

[31] 胡吉祥,许洪波,刘悦,等. 重复串特征提取算法及其在文本聚类中的应用[J]. 计算机工程,2007,33(2):65-67.

[32] ZAMIR O, ETZIONI O, MADANI O, et al. Fast and intuitive clustering of Web documents[C]// Proceedings of the 3rd International Conference on Knowledge Discovery and Data Mining. New York: AAAI Press, 1997:287-290.

[33] HONG Y, SAM K. Learning assignment order of instances for the constrained K-means clustering algorithm[J]. IEEE Transactions on Systems Man and Cybernetics Part B-Cybernetics, 2009, 39(2):568-574.

[34] HALL L O, GOLDGOF D B. On convergence properties of the single pass and online fuzzy c-means algorithm[C]// 2010 IEEE International Conference on Fuzzy Systems. Washington, DC: IEEE, 2010:1-3.

[35] AIOLLI F, SAN-MARTINO G, HAGENBUCHNER M, et al. Learning nonsparse kernels by self organizing maps for structured data[J]. IEEE Transactions on Neural Networks, 2009, 20(12):1938-1949.

[36] ZAMIR O, ETZIONI O. Web Document Clustering: A Feasibility Demonstration[C]// Proceedings of the 21st International ACM SIGIR Conference on Research and Development in Information Retrieval. Melbourne, Australia, 1998:46-54.

[37] JANRUANG J, GUHA S. Semantic Suffix Tree Clustering[C]// Proceedings of International Conference on Data Engineering and Internet Technology. 2011:35-40.

[38] 林建敏,谢康林. 基于 PAT-array 和模糊聚类的文本聚类方法[J]. 计算机工程,2004,30(12):126-127.

[39] 翟献民,田生伟,禹龙,等. 面向维吾尔语文本的改进后缀树聚类[J]. 计算机应用,2012,32(4):1078-1081.

[40] ZHANG D, DONG Y S. Semantic, hierarchical, online clustering of web search results[C]// Advanced Web Technologies and Applications, Lecture Notes in Computer Science. 2004:69-78.

[41] YAMAMOTO M, CHURCH K W. Using Suffix Arrays to Compute Term Frequency and Document Frequency for All Substrings in a Corpus[J]. Computational Linguistics, 2001, 27(1):1-30.

[42] LEE S D, DE RAEDT L. An efficient algorithm for mining string databases under constraints[C]// Knowledge Discovery in Inductive Databases, Lecture Notes in Computer Science. 2005:108-129.

[43] BEIL F, ESTER M, XU X. Frequent term-based text clustering[C]// Proceedings of the 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. Edmonton, Alberta, Canada, 2002:436-422.

[44] FUNG B C M, WANG K, ESTER M. Hierarchical Document Clustering Using Frequent Itemsets[C]// Proceedings of the SIAM International Conference on Data Mining. 2003.

[45] GURALNIK V, KARYPIS G. A scalable algorithm for clustering sequential data[C]// Proceedings of the IEEE International Conference on Data Mining. 2001:179-186.

[46] WANG J Y, ZHANG Y Z, ZHOU L Z, et al. Discriminating subsequence discovery for sequence clustering[C]// Proceedings of the 7th International Conference on Data Mining. 2007:605-610.

[47] GROSSI R, VITTER J S. Compressed suffix arrays and suffix trees with applications to text indexing and string matching[J]. SIAM Journal on Computing, 2005, 35(2):378-407.

[48] 刘小珠,彭智勇. 全文索引技术时空效率分析[J]. 软件学报,2009,20(7):1768-1784.

[49] OKANOHARA D, SADAKANE K. An online space-efficient algorithm for searching the longest match string[M]// Lecture Notes in Computer Science. 2008:696-707.

[50] VU L, ALAGHBAND G. Novel parallel method for mining frequent patterns on multi-core shared memory systems[C]// Proceedings of the International Workshop on Data-Intensive Scalable Computing Systems. 2013:49-54.

[51] BEDATHUR S, HARITSA J R. Search-optimized persistent suffix tree storage for biological applications[C]// Proceedings of 12th IEEE International Conference on High Performance Computing. 2005.

[52] MOEN S, AKSEHIRLI E, GOETHALS B. Frequent Itemset Mining for Big Data[C]// Proceedings of IEEE International Conference on Big Data. 2013.

[53] MOENS S, AKSEHIRLI E, GOETHALS B. Frequent itemset mining for big data[C]// IEEE International Conference on Big Data. 2013:111-118.

[54] 阿力木江·艾沙,库尔班·吾布力,吐尔根·依布拉音. 维吾尔文文本分类中若干问题的研究[M]. Scientific Research Publishing, Inc. USA, 2015.

[55] 崔世起,刘群,孟遥,等. 基于大规模语料库的新词检测[J]. 计算机研究与发展,2006,43(5):927-932.

[56] LUO Z Y, SONG R. An Integrated Method for Chinese Unknown Word Extraction[C]// Proceedings of the Third SIGHAN Workshop on Chinese Language Learning. Barcelona, Spain, 2004:148-155.