改进的TextRank双层单文档摘要提取算法

何春辉[1]，李云翔[2]，王孟然[3]，王梦贤[4]
(1. 湘潭大学 数学与计算科学学院，湖南 湘潭 411105；2. 湖南城市学院 理学院，湖南 益阳 413000；
3. 长沙县印山学校，长沙 410135；4. 湖南城市学院 管理学院，湖南 益阳 413000)

摘 要：本文提出了基于句子重要度的累积贡献率摘要句筛选算法和改进的TextRank双层单文档摘要提取算法. 摘要提取算法采用了分层结构，在不同层上融合了基于句子重要度的累积贡献率摘要句筛选算法，同时使用了长句和短句两种不同分割方式相结合的策略来构建摘要提取算法. 用手工整理的中文单文档摘要数据集验证了算法的性能，结果表明：提取的摘要质量非常好.

关键词：TextRank；信息抽取；摘要算法；累计贡献率
中图分类号：TP391 **文献标识码**：A doi:10.3969/j.issn.1672-7304.2017.06.0012
文章编号：1672-7304(2017)06-0055-06

Improved TextRank Double Layers Single-document Summation Extracting Algorithm

HE Chunhui[1], *LI Yunxiang*[2], *WANG Mengran*[3], *WANG Mengxian*[4]
(1. School of Mathematics and Computational Sciences, Xiangtan University, Xiangtan, Hunan 411105, China; 2. School of Science, Hunan City University, Yiyang, Hunan 413000, China; 3. Yinshan School of Changsha County , Changsha, Hunan 410135,China; 4. School of Management, Hunan City University, Yiyang, Hunan 413000, China)

Abstract: A summation sentence selection algorithm based on accumulating contribution rate of sentence importance and an improved TextRank double layers single-document summation extraction algorithm are proposed in this paper. The summation extraction algorithm adopts the hierarchical structure, on the different layer, the summation sentence selection algorithm based on accumulating contribution rate of sentence importance is blended, at the same time, using long sentences and short sentences in two different ways to construct summation extraction algorithm. The manual finishing Chinese single-document summation data set is used to verify the performance of the algorithm, the results show that the quality of the extraction summation is very fine.

Key words: TextRank; information extraction; summation algorithm; accumulating contribution rate

随着信息技术的发展，文本数据出现了指数级的增长趋势. 面对如此丰富多样的信息，如何从大量的文本内容中快速筛选出自己所需信息就显得格外重要. 文本摘要提取算法起源于 1958 年，最初由 IBM 公司的 H. P. Luhn[1]提出. 国内在摘要提取算法方面的研究起步较晚，早期的文本摘要系统由王永成[2]教授在1988年研制成功. 通过对文献[3]的分析发现目前主流的文本自动摘要提取算法主要分为了 2 大类：一类是生成式摘要提取算法，它们对大规模数据集的依赖程度很大，而且算法计算复杂度较高，目前还处于理论研究阶段；另一类是抽取式摘要提取算法，该类算法计算复杂度低、易操作，因此应用较广泛，但算法通常是以单一句子为单元进行摘要抽取，生成的摘要会包含一些噪声数据.

对于摘要句的筛选方式，目前主流的做法就是根据句子的重要度进行降序排序，然后采用某种指标来选取重要度排名靠前的句子进行合并来生成文档的摘要. 目前这些主流的摘要句筛选方法还不是很成熟，不利于提取高质量的摘要.

为了克服现有算法的不足，在摘要句筛选方面，提出了一个基于累积贡献率的摘要句筛选算法；摘要提取算法方面，在经典 TextRank 算法基础上提出了改进的 TextRank 双层单文档摘要提

收稿日期：2017-10-23
基金项目：益阳市科技计划项目(2014JZ40)
第一作者简介：何春辉(1991-)，男，湖南永州人，工程师，硕士，主要从事数据挖掘与信息处理研究. E-mail: xtuhch@163.com

取算法．文章各节内容分别对 TextRank 算法进行了相关介绍，根据相关文献资料详细阐述了文档句子重要度计算方法，给出了基于句子重要度累积贡献率的摘要句筛选算法的步骤和改进的 TextRank 双层单文档摘要提取算法的流程图及核心步骤，最后用相应的数据集来验证了算法的性能并根据实验结果给出了相应的结论．

1 经典 TextRank 算法

TextRank[4]算法是 Mihalcea 和 Tarau 于 2004 年在自动摘要提取任务中得到的成果，原理与 PageRank[5]算法类似，但它未考虑文档的篇章结构等信息．为提升算法性能，有学者提出了改进的 iTextRank[6]摘要提取算法来提高摘要质量．

2 句子重要度计算和摘要句的筛选方法

2.1 基于 TF-IDF 和余弦距离计算句子相似度

文档经过预处理以后，得到段落、句子、词语等信息．本文采用词频-逆文档频率 TF-IDF[7]算法计算词语的权重．词频 TF 的计算公式为

$$\mathrm{TF}_i=\frac{n_i}{\sum_k n_k}. \tag{1}$$

式中 TF_i 为第 i 个词的词频；n_i 为文档中第 i 个词语出现的次数；右边的分母为文档中所有词语出现的次数．逆文档频率 IDF 的计算公式为

$$\mathrm{IDF}(t,D)=\log(\frac{N}{n_t}). \tag{2}$$

式中 t 表示被测试词语；D 表示总文档集；N 表示文档总个数；n_t 表示含有被测词语 t 的文档总数量．将 TF 和 IDF 得到的结果进行相乘，即用 W_t=TF*IDF 算出第 t 个词语的权重 W_t.

下一步结合空间向量模型并采用余弦距离计算句子间的相似度，余弦距离计算公式为

$$\mathrm{Sim}(s_i,s_j)=\frac{\sum_{k=1}^{N}(w_{ik}\cdot w_{jk})}{\sqrt{(\sum_{k=1}^{N}w_{ik}^2)\times(\sum_{k=1}^{N}w_{jk}^2)}}. \tag{3}$$

式中 s_i 和 s_j 分别为句子 i 和句子 j；w_{ik} 为句子 i 中的第 k 个特征词的权重；w_{jk} 为句子 j 中的第 k 个特征词的权重．利用式(3)算出所有句子之间的相似度，得到句子的相似度矩阵[6] $SS_{n\times n}$，权重转移矩阵的一般结构为

$$SS_{n\times n}=\begin{pmatrix} w_{11} & \cdots & w_{1n} \\ \vdots & \ddots & \vdots \\ w_{n1} & \cdots & w_{nn} \end{pmatrix}. \tag{4}$$

矩阵的所有元素是由各个句子间的相似度构成，它是一个对称矩阵，且对角线上元素均为 1．

2.2 句子重要度计算方法

根据 2.1 节的叙述，经过计算之后得到文档中句子间的相似度矩阵，然后构建无向图，采用 PageRank 算法对所有句子的重要度进行求解．句子重要度 SW 的计算和所有句子重要度权重矩阵 P_i 的求解算法分别如式(5)和式(6)[6]所示．

$$SW(V_i)=\frac{1-d}{n}+d*\sum_{V_j\in\mathrm{In}(V_i)}\frac{w_{ij}}{\sum_{V_k\in\mathrm{Out}(v_j)}w_{jk}}SW(V_j). \tag{5}$$

其中 $SW(V_i)$ 是第 i 个句子重要度；n 是节点个数；d 是阻尼系数，设置为 0.85；$\mathrm{In}(V_i)$代表链入 V_i 的节点集合；$\mathrm{Out}(V_j)$代表链出 V_j 的节点集合；w_{ij} 表示第 i 个句子和第 j 个句子间相似度．

$$P_i=SS_{n\times n}\cdot P_{i-1}. \tag{6}$$

由于计算节点的重要度需要用到节点自身的值，这样就需要进行迭代计算．PageRank 算法证明了式(6)的迭代计算方式满足一定条件时最终会收敛．因此当相连 2 次迭代的结果 P_i 和 P_{i-l} 的差别很小的时候，就停止迭代，输出由所有句子重要度构成的向量值，即每个句子的重要度．

2.3 摘要句筛选算法

经过了文本预处理步骤一篇文档会被分解成若干个句子，相似句去重后，通过 2.2 节的方法计算文档里面所有句子的重要度结果，根据结果来筛选摘要句．对于摘要句的筛选方式，本文提出了基于句子重要度的累积贡献率摘要句筛选算法，该算法可以根据设定的累积贡献率阈值自动给出文档中摘要句子的集合．其**核心步骤**如下：

算法输入：向量 P 和累积贡献率阈值 ACR_0

算法输出：文档的摘要句子总数

步骤(1)：输入非空向量 P=[r_1，r_2，r_3，…，r_n]，ACR_0；

步骤(2)：求句子重要度总和，即 S=sum(r_i)，(i=1，2，…，n)；

步骤(3)：取 sw=0，ACR=0，sent_num=0；//即初始化

步骤(4)：计算累积贡献率 ACR 得到摘要句子总数 sent_num 的值，再循环执行：(i=1，i<P.length)，{sw=sw+r_i；//对句子重要度求和，ACR=sw/S；//

计算累积贡献率 ACR

if(ACR>=ACR_0){//判断当前累积贡献率是否达到了设定的累计贡献率阈值 ACR_0

sent_num=i；//获取摘要句子总数

break；//跳出循环，得到 sent_num

}else{

i=i+1；

}

}

步骤(5)：return sent_num；//返回摘要句子的总数.

根据上述算法步骤可以自动获取文档的摘要句子总数 sent_num，结合句子重要度向量降序排名结果，可取出所有句子中重要度排名在前 sent_num 的句子作为摘要句，根据这些摘要句的重要度在未排序的句子重要度向量 P_i 中找到与之对应的值所在的下标，根据下标值找到对应的原始句子，按照出现先后顺序将找出句子的内容进行合并生成最终的摘要．这个算法考虑了句子总数和句子本身内容所带有的重要度信息来自动筛选摘要句，能较好的弥足现有算法的不足.

3　改进的 TextRank 双层单文档摘要提取算法

现有抽取式摘要提取算法大都以句子为单位进行抽取,这样会导致句子存在很高的相似度时，算法无法避免对相似句子的计算，从而影响摘要质量．为了较好的解决该问题，本文主要作如下改进：将抽取式的文本自动摘要算法当做一个过滤的过程，把摘要提取算法当作一个过滤器，可通过增加过滤器的层数或者改变过滤器的内部结构来减少过滤后留下的噪声．基于这种启发，提出了改进的 TextRank 双层单文档摘要提取算法，算法的具体流程图如图 1 所示.

根据图 1 所示的流程图，下面对算法实施方式及各参数的取值情况和两层之间存在差异性的步骤给出相关说明．算法在结构上分了两层，从表面上看第 1 层与第 2 层大致是相同的，但功能却不一样，因此一些关键步骤的处理方式存在较大差异．算法核心步骤及具体差异说明如下：

算法输入：单篇中文文档的内容

算法输出：对应文档提取的摘要内容

步骤(1)：文档分句及预处理步骤．主要是对文档执行分句、去除特殊字符及空格、舍弃长度

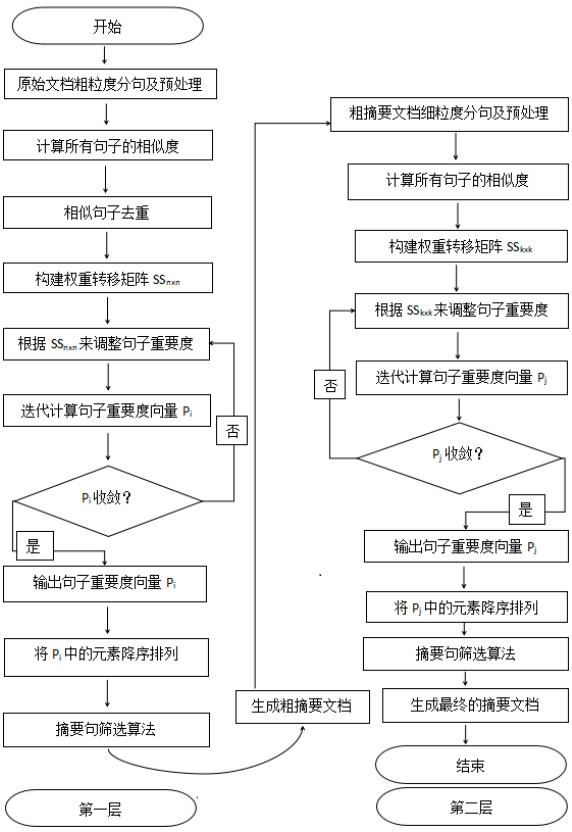

图 1　改进的 TextRank 双层单文档摘要提取算法

较短的句子、分词、计算词语的 TF-IDF 值等常规的预处理操作．第 1 层和第 2 层除了文档的分句标准与舍弃长度较短的句子上存在差异之外，其余地方无差异．第 1 层是精筛层，句子切分的细粒度较粗，分句的不同在于第 1 层中采用汉语中常用的句号、问号、分号、感叹号、冒号、省略号这 6 个符号作为分句标准，这样便于快速有效的筛选出重要的长句；第 2 层是一个去噪层，分句标准是在第 1 层的基础上加了逗号，这样就将原始长句变成细粒度更小的短句，方便剔除原始句子中的噪声．在舍弃较短的句子方面，第 1 层精筛层主要是舍弃长度为零的空句，第 2 层去噪层主要是舍弃长度小于 4 的句子，经测试可以有效降低噪声.

步骤(2)：计算所有句子间的相似度．这一步利用公式(3)来进行求解，对句子相似度高于 90% 的句子进行后向去重，并对重复句子做出特殊标记，这样可避免计算重复句子与其它句子之间的相似度，此步骤两层之间的处理方式无差异.

步骤(3)：相似句子去重．此步只在第 1 层中设置，第 2 层中未设置，它的功能是从原始文档的句子列表中剔除掉第 1 层在步骤(2)中做了特殊标记的句子.

步骤(4)：根据句子间相似度来构建权重转移矩阵 $SS_{n\times n}$，构建方式如公式(4)所示，此步两层之间无差异.

步骤(5)：根据权重转移矩阵来调节句子重要度，计算方式见式(5)，此步两层之间无差异.

步骤(6)：迭代计算句子重要度向量 P_i，计算方式见式(6)，此步骤两层之间处理方式无差异.

步骤(7)：判断句子重要度向量 P_i 是否收敛. 若否，返回步骤(5)；若是，输出句子重要度向量 P_i，此步两层之间无差异.

步骤(8)：将上一步得到的句子重要度向量的元素值按照降序方式进行排序，此步骤两层之间无差异.

步骤(9)：摘要句筛选算法. 其具体计算过程在2.2节的摘要句筛选算法部分给出了详细描述，这是本文的核心部分，在两层之间存在较大的差异，主要差异在于不同层之间算法设定的句子重要度累积贡献率阈值 ACR_0 是不同的. 第 1 层是精筛层，目标是在大量的句子中选出极少数重要的句子，根据实验测试发现，算法将第 1 层的累积贡献率阈值 ACR_0 设为 0.2 会取得较好效果；而第 2 层是去噪层，目标是剔除前面提出重要句子中的少数噪声数据，经试验测试，发现将第 2 层的累积贡献率阈值 ACR_0 设为 0.8 时，算法会取得较好效果，其余处理方式无差异.

步骤(10)：摘要文档的生成. 经过第 1 层处理后，得到精筛层提取出来的句子列表，根据这些摘要句子重要度的值在未排序的句子重要度向量 P_i 中找到与之对应的值所在的下标，根据下标值在原始文档中找出对应句子内容，再按照出现的先后顺序将找出句子的内容进行合并并生成最终的摘要，且将它作为第 2 层输入，经过第 2 层的去噪处理后，得到最终的句子列表，然后按照这些摘要句重要度的值在未排序的句子重要度向量 P_i' 中找到与之对应的值所在的下标，根据下标值找出第 1 层给出的粗摘要文档中对应句子的内容，将找出句子的内容按照先后顺序进行合并，以生成最终的摘要.

4 算法验证

4.1 摘要句筛选算法的验证

为了验证摘要句筛选算法的效果，通过随机抽取 1 篇关于文本挖掘的中文文档来进行自动摘要测试，根据摘要的质量来验证算法的效果. 本实验所用的样例文档内容如下：

文本挖掘是抽取有散布在文本文件中有价值的知识，并且利用这些知识更好地组织信息的过程. 国家重点研究发展规划首批实施项目中明确指出，文本挖掘是“自然语言理解与知识挖掘”中的重要内容. 文本挖掘是信息挖掘的一个研究分支，用于基于文本信息的知识发现. 文本挖掘利用智能算法，如神经网络、基于案例的推理、可能性推理等，并结合文字处理技术，分析大量的非结构化文本源如文档、电子表格、客户电子邮件、问题查询、网页等，抽取或标记关键字概念、文字间的关系，并按照内容对文档进行分类，获取有用的知识和信息. 文本挖掘是包括数据挖掘技术、信息抽取、信息检索的交叉学科.

本文算法结构有两层，实验中第 1 层累积贡献率阈值为 0.2，第 2 层累积贡献率阈值为 0.8. 根据算法结构，第 1 层得到了长句子重要度递减曲线图和各句重要度的值，结果如图 2 所示.

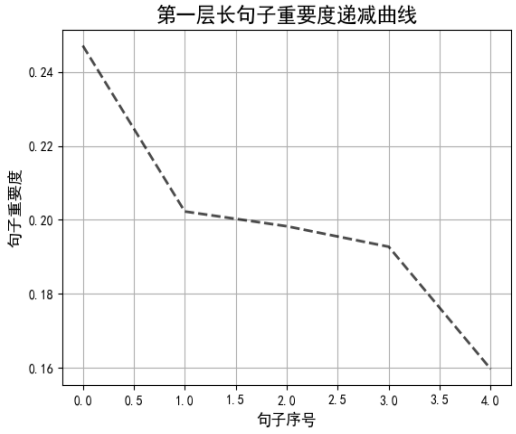

图 2 第 1 层长句子重要度递减曲线图

根据图 2 的结果，原始句子一共有 5 句，结合设定的累积贡献率阈值 0.2，算法自动给出了摘要句子数量为 1，然后利用句子重要度最高的句子的下标在原始文档的句子列表中找出相应的句子为第 4 句，对应的句子内容为“文本挖掘利用智能算法，如神经网络、基于案例的推理、可能性推理等，并结合文字处理技术，分析大量的非结构化文本源如文档、电子表格、客户电子邮件、问题查询、网页等，抽取或标记关键字概念、文字间的关系，并按照内容对文档进行分类，获取有用的知识和信息. ”，再将上面句子的内容作为粗摘要文档输入第 2 层，经过预处理步骤之后，

算法第 2 层得到了短句子重要度递减曲线图及各短句重要度的值，如图 3 所示.

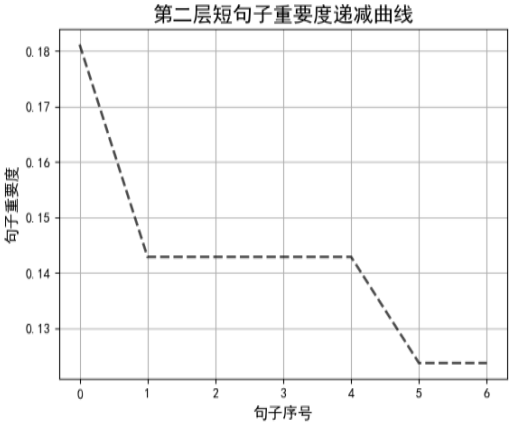

图 3　第 2 层短句子重要度递减曲线图

根据图 3 结果，原始短句子一共有 7 句，设定的累积贡献率阈值为 0.8,算法自动给出了摘要句数量为 5，然后利用句子重要度最高的前 5 个句子的下标依次在粗摘要文档的句子列表中找出相应的句子为第 1、3、4、5、7 句，各句对应的原始句子内容分别为(中括号里的内容)：[文本挖掘利用智能算法，]、[并结合文字处理技术，]、[分析大量的非结构化文本源如文档、电子表格、客户电子邮件、问题查询、网页等，]、[抽取或标记关键字概念、文字间的关系，]、[获取有用的知识和信息．]．根据上述两份摘要结果可以看出，第 2 份摘要的质量比第 1 份摘要的质量高很多，在保证摘要的简洁性、准确性和连贯性的同时，有效的剔除了噪声句子.

4.2　改进的 TextRank 双层单文档摘要提取算法在不同数据集上的试验对比

本文在公开的中文摘要数据集[8]LCSTS 上，通过人工选择了 6 个领域共 1 000 篇内容比较完善的文档，按照文档标题表达的意思，由 3 个不同的人从文本内容中抽取了相应的短句组成目标摘要，并且只允许从原文中抽取短句子，不允许根据原文的理解重新生成新句子，以便跟算法抽取的摘要结果形成对比．最后，在抽取的数据集上进行了试验，采用 Edmundson[9]方法验证算法性能．Edmundson 方法以句子为单位进行比较，用重复率 R 来评测，其具体公式为

$$\text{重复率}R=\frac{MS}{TS}\times 100\%\ . \tag{7}$$

$$\text{平均重复率}\overline{R}=\frac{\sum_{i=1}^{N}R_i}{N}\ . \tag{8}$$

式(7)中 MS 是指提取的摘要句子与目标句子内容相同的句子数；TS 是指文档中的句子总数．式(8)中 N 表示文档总数量；R_i 表示第 i 篇文档的重复率．数据集一共包含 1 000 篇文档，涵盖了 6 个领域，其中混合集是指将以上 1 000 篇不同的文档经过混合后随机抽取 200 篇文档组成的混合数据集，具体划分情况见表 1.

表 1　数据集在 6 个不同领域的划分

领域名	政治	经济	科技	金融	教育	房地产	混合集
文档数	200	100	200	200	100	200	200

算法在上述数据集上的测试结果见表 2.

表 2　改进 TextRank 双层单文档摘要提取算法在不同数据集上的试验结果

各领域数据集划分	试验文档数量/篇	平均重复率 $\overline{R}$/%
政治	200	77.3
经济	100	75.8
科技	200	79.8
金融	200	78.4
教育	100	74.7
房地产	200	79.4
混合集	200	78.1

数据集说明：因为 LCSTS 数据集在不同领域所包含的文档数量不同，根据实际情况，由人工整理了 1 个非平衡测试数据集，其中经济和教育领域各 100 篇文档，其余 4 个领域以及混合集各 200 篇文档．根据表 2 试验结果可看出，算法在不同数据集上的性能表现良好，在科技领域的数据集上表现最好，平均重复率达到了 79.8%；教育领域表现一般，平均重复率只有 74.7%；在混合集上的平均重复率为 78.1%．由实验结果可知该算法既可以处理特定领域的摘要提取任务，又能处理多个交差领域的摘要提取任务，在摘要提取的覆盖率方面，算法的性能有较大的提升.

5　结论

自动文本分类技术的研究已成为模式识别领域的一个热点[10]．文本摘要更是信息抽取领域的一个难点．本文在经典 TextRank 摘要提取算法的基础上提出了改进的 TextRank 双层单文档摘要提取算法，该算法很好的融合了新提出的基于句子重要度的累积贡献率摘要句筛选算法来构建智能的文本摘要提取算法．从实验结果来看，

算法提取的摘要质量较高，可适应多领域文本摘要提取的任务．而不足之处在于只使用了人工整理的 1 000 篇中文文档进行验证，算法没在其它大规模数据集上做相关测试．未来的工作将尝试结合句法依存分析进一步的提高自动摘要提取算法的性能．

参考文献：

[1]LUHN H P. The automatic creation of literature abstracts[J]. IBM Journal of Research and Development, 1958, 2(2): 159-165.

[2]WANG Y C. Automatic extraction of words from Chinese textual data[J]. Journal of Computer Science and Technology, 1987, 2(4): 287-291.

[3]曹洋．基于TextRank算法的单文档自动文摘研究[D]．江苏：南京大学, 2016.

[4]MIHALCEA R, TARAU P. Textrank: Bringing order into texts[C]. Conference on Empirical Methods in Natural Language Processing, EMNLP 2004, 2004: 404-411.

[5]PAGE L, BRIN S, MOTWANI R, et al. The PageRank citation ranking: Bringing order to the web[J]. Stanford Digital Libraries Working Paper, 1998, 9(1): 1-17.

[6]余珊珊, 苏锦钿, 李鹏飞．基于改进的TextRank的自动摘要提取方法[J]．计算机科学, 2016, 43(6): 240-247.

[7]AIZAWA A. An information-theoretic perspective of TF–IDF measures[J]. Information Processing & Management, 2003, 39(1): 45-65.

[8]HU B T, CHEN Q C, ZHU F Z. LCSTS: A large scale Chinese short text summarization dataset[J]. Computer Science, 2015, 1967-1972.

[9]EDMUNDSON H P. New methods in automatic extracting[J]. Journal of the Acm, 1969, 16(2): 264-285.

[10]秦玉平, 邱凤凤, 冷强奎．组合凸线器和Hadamard纠错码相结合的多类文本分类算法[J]．渤海大学学报：自然科学版, 2017, 38(1): 71-75.

(责任编校：龚伦峰)