doi:10.3969/j.issn.1000-436x.2013.z2.030

基于斜率密度聚类的相似文本标定

邹杜[1]，唐文军[1]，龙卫江[2]，张凌[3]
（1. 华南理工大学 信息网络工程研究中心，广东 广州 510640；
2. 华南理工大学 理学院，广东 广州 510640；3. 华南理工大学 计算机学院，广东 广州 510640）

摘 要：相似文本标定是抄袭检测的一个重要环节，现有标定方法大多采用直接对文本或指纹进行合并的方式，标定精度受干扰信息影响较大。针对这种局限性，分析了匹配指纹对的语义特征，提出基于斜率密度的相似文本聚类方法，将文本匹配合并问题转化成稠密样本点聚类问题，并在 PAN 公用语料库上对该方法进行了测试，得到的主要指标优于 PAN10 前 3 名。目前已将该方法用于华南理工大学特色专业教学平台的作业查抄，取得了较好的效果。

关键词：抄袭检测；相似文本标定；聚类；指纹
中图分类号：TP391 **文献标识码**：A **文章编号**：1000-436X(2013)Z2-0157-06

Similar text positioning method based on slope-density cluster

ZOU Du[1], TANG Wen-jun [1], LONG Wei-jiang[2], ZHANG Ling[3]
(1. Information Network Engineering and Research Center, South China University of Technology, Guangzhou 510640, China;
2. School of Science, South China University of Technology, Guangzhou 510640, China;
3. School of Computer Science & Engineering, South China University of Technology, Guangzhou 510640, China)

Abstract: Similar text positioning is an important part of plagiarism detection. The existing positioning method directly merges text or fingerprint to obtain similar text. Due to the disturb information in the similar text, the positioning accuracy is poor. The semantic features of the match fingerprints were analyzed, and a cluster method based on slope density for similar text positioning was proposed, which converts the text merge problem into dense sample points clustering problem, and improves the efficiency and accuracy of the positioning. Through the experiment on the PAN public corpus, the result shows it performs better than the PAN10 top three. This method has been used in the South China University of Technology 's feature professional teaching platform to detect the plagiarism of homework.

Key words: plagiarism detection; similar text positioning; cluster; fingerprint

1 引言

文本重用已成为人们诟病的问题，它与版权保护、文件管理、一稿多投等密切相关。人们处理文本重用问题的一个有效方法是将重用检测分为 2 个重要的任务：1) 预选，对指定的待判文档，从源文档集中找出相似程度最高的前 N 个，形成该待判文档的候选源文档集；2) 相似文本标定，对这 N 个候选源文档进行详细分析，定位重用片段，合并零散片段并标记重用内容。本文研究其中第二个任务。

2 常用的相似文本标定方法

相似文本检测的研究由来已久。MANBER[1]为减少文件系统冗余检测相同内容文件，BRIN 等[2]关注所编代码是否被他人抄袭。BRODER[3,4]提出了 Shingling 算法，STEIN 等[5]系统分析了 Hash 技术在信息检索方面的应用。CHARIKAR 等[6]提出了一种特殊的局部敏感散列（LSH）方法，将基于计数的相似度量指标用于文档查重，SHIVAKUMAR[7]等借鉴信息检索的向量空间模型（VSM），使用基于词频

收稿日期：2013-09-05
基金项目：国家自然科学基金资助项目 (61070092)
Foundation Item: The National Natural Science Foundation of China(61070092)

统计的方法来度量文档相似性。SCHLEIMER[8]等提出 Winnowing 方法，使用重叠 *k*-gram 方法和移动窗口取小的方式获取文档抽样。

随着检测算法研究的深入，相似文本标定成为研究的一个重点方向。SEDIYONO[9]提出了 LCCW 算法，以段落为单元，将段落拆分成连续单词的集合，通过逐词比较，生成最长连续相同单词集合。ZASLAVSKY 等人[10]提出了 MDR 系统，将预处理后的文档拆分成固定长度的 token，使用 matching statistics 算法构造后缀树，利用 LCS 算法得到最长公共子串。以上算法通过逐词比较的方法得到相似文本片段，属于精确匹配方法，无法处理包含干扰信息的相似文本。

KASPRZAK[11]采用 Word-5-grams 启发式匹配方法，2 个匹配 grams 间允许有最多 49 个不匹配的 grams。ZOU[12] 首次将聚类的概念用于相似文本标定，提出基于通道的聚类方法，合并通道内的相似文本片段。Muhr[13]以句子为单位，通过相似度判定句子是否相似，句间允许一定的不相似短句的存在。通过启发式匹配方法，可以将干扰信息造成的小片段合并成一个连续的大片段，并且可以通过调整阈值来减轻不同程度的干扰对相似文本标定造成的影响。

针对现有标定算法存在的问题，笔者首次将基于密度聚类方法用于相似文本标定中，结合相似文本的语义特征，提出基于斜率密度聚类的相似文本标定算法，从理论上证明该算法对相似文本的标定是可行的，并在标准数据集上检验该算法的有效性。

3 基于斜率密度聚类的相似文本标定方法

3.1 系统框架

相似文本检测的基本流程如图 1 所示。

文本重用检测分为预处理、快速预选和相似文本标定 3 个阶段。其中相似文本标定要解决 3 个问题：1）处理匹配指纹对，找出其语义规律；2）合并属于同一相似文本片段的匹配指纹对；3）处理合并结果并标定相似文本，得到较高精度的定位结果。

3.2 处理匹配指纹

相似文本标定要求能通过任一指纹定位到其代表的文本在原始文档中的相应位置。文中利用倒排索引记录指纹对应原始文档中的位置，使用 Hamming 方法快速比较 2 个指纹向量 F_x 和 F_y，得到相同指纹对应的下标，形成匹配指纹对(x,y)。

以 PAN 语料集中 2 个文档为例说明匹配指纹对的语义特征，将匹配指纹对投影到二维坐标的效果如图 2 所示。

图中坐标分别表示 2 个指纹向量中的指纹序号，黑点表示匹配指纹对。通过对比文档内容可知，不同分布的匹配指纹对代表不同的语义。

1）图中沿 45°方向分布的线段代表一段连续的匹配指纹对，对应文档中一段相同的文本。

2）图中类似正方形的阴影块代表一定数量、不连续但分布相对集中的匹配指纹对，对应文档中一段加入了干扰信息的相似文本。

3）图中零散出现的黑点表示文档相应位置只有少量相同的短语，不存在符合语义的相似文本段。

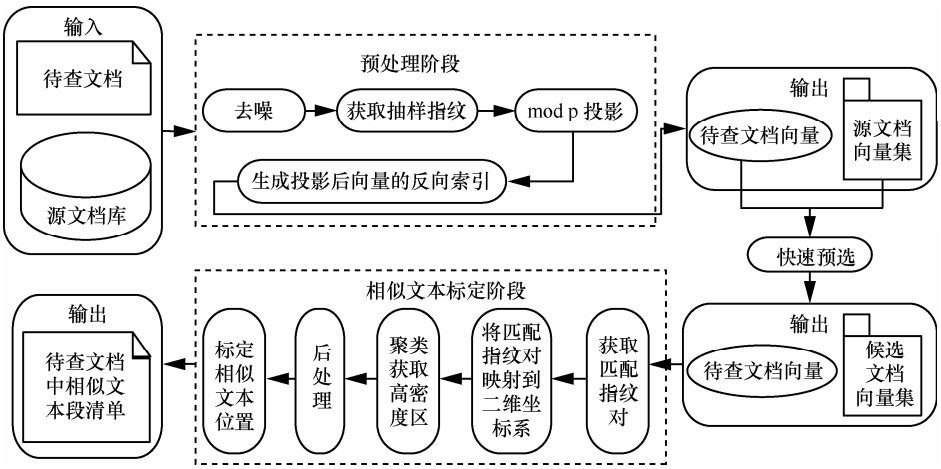

图 1 相似文本检测流程

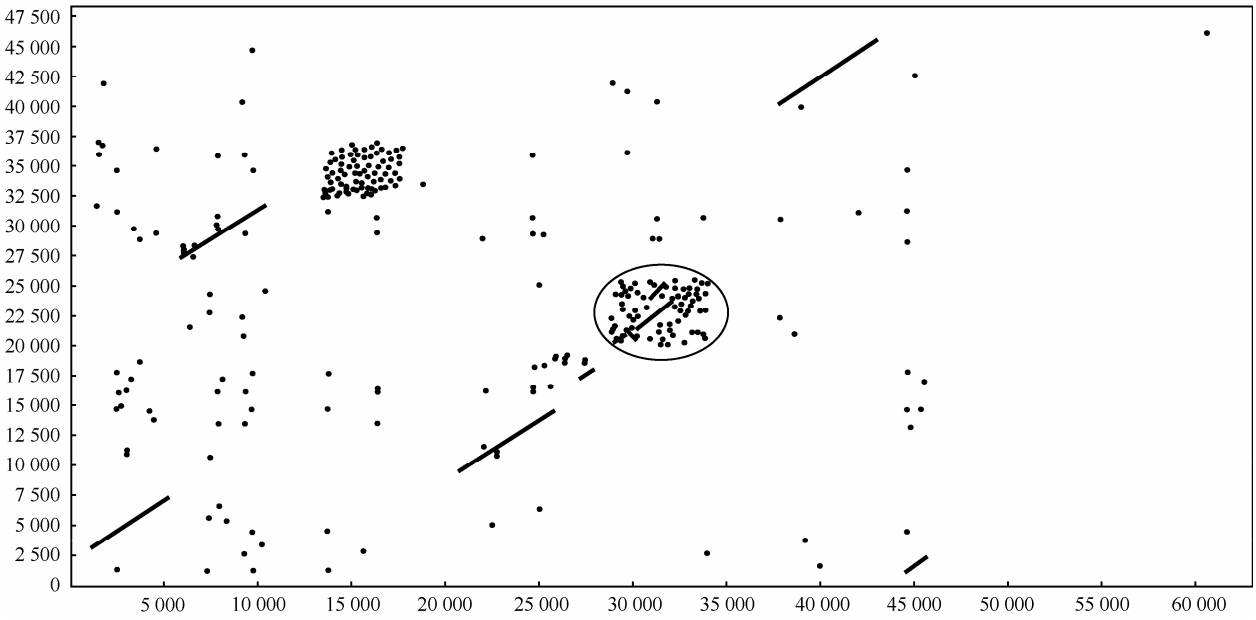

图 2　匹配指纹对分布

4）图中空白区域表示在文档的相应位置不存在相同或近似的文本片段。

相似文本标定的任务就是通过合适的算法找出图 2 中所有长线段和阴影块。由于干扰的存在，本该连续的匹配指纹对序列被拆分成零散的许多小线段，现有的精确式和启发式标定算法都很难将其标定为同一个相似文本。

根据以上分析，提出了基于斜率密度的聚类算法，减少干扰信息对标定的影响。

3.3　基于斜率密度聚类的相似文本标定

3.3.1　基于斜率密度聚类相关概念

基于密度的聚类算法是一种基于高密度联通区域的聚类算法。在整个样本空间中，各目标类簇由一群被低密度区域分割的稠密样本点组成，算法的目的是过滤低密度区域，发现稠密样本点。

与普通的高密度联通区域中稠密样本点的无规律分布不同，在文本重用中，匹配指纹对的分布必须满足一定的语义特征。借鉴基于密度的聚类算法原理，结合匹配指纹对的语义特征，给出基于斜率密度聚类相关的定义具体如下。

匹配点：一个匹配指纹对在二维平面上的投影，表示为 $p(x,y)$，其中 x 和 y 分别代表在各自抽样指纹向量中的序号。

ε 邻域：给定匹配点 $p(x,y)$ 沿坐标轴方向距离为 ε 内的区域称为该匹配点的 ε 邻域。

类簇区间：包含某个类簇所有匹配点的最小区间。表示为 $Ca_i(x,y,l_x,l_y)$ ，其中 x 和 y 是类簇 C_i 中匹配点在坐标轴上最小投影值，l_x 和 l_y 是类簇区间的边长。

斜率密度：在相似文本标定中，匹配指纹只有沿类簇区间的对角线方向分布才有实际的语义。因此定义类簇区间 Ca_i 的斜率密度为

$$\rho_i = \max\left(\frac{\sum x_i}{l_x}, \frac{\sum y_i}{l_y}\right)$$

其中 $\sum x_i$ 和 $\sum y_i$ 分别表示簇类中匹配点在相应坐标轴上的投影之和（不计重复投影），由上式可知，$0 \leqslant \rho_i \leqslant 1$。

类簇区间包含：如果类簇区间 Ca_i 和 Ca_j 满足条件

$$\forall p_j \in Ca_j : p_j \in Ca_i$$

则称类簇区间 Ca_i 包含 Ca_j，记为 $Ca_j \subseteq Ca_i$，简称类簇包含。

类簇区间相交：如果类簇区间 Ca_i 和 Ca_j 满足条件

$$\exists p_j \in Ca_j : p_j \in Ca_i$$

则称类簇区间 Ca_i 和 Ca_j 相交，记为 $Ca_i \cap Ca_j$，简称类簇相交。

斜率密度可达：类簇区间 Ca_i 和 Ca_j 中左下角和右上角 2 组顶点，如果存在一个顶点在另一类簇顶点的 ε 邻域内，则称 2 个类簇区间斜率密度可达。

形式化表示为

$$(|x_i-x_j|\leqslant\varepsilon\wedge|y_i-y_j|\leqslant\varepsilon)\vee(|x_i-x_j-l_{x_j}|$$
$$\leqslant\varepsilon\wedge|y_i-y_j-l_{y_j}|\leqslant\varepsilon)\vee(|x_i+l_{x_i}-x_j|$$
$$\leqslant\varepsilon\wedge|y_i+l_{y_i}-y_j|\leqslant\varepsilon)\vee(|x_i+l_{x_i}-x_j-l_{x_j}|$$
$$\leqslant\varepsilon\wedge|y_i+l_{y_i}-y_j-l_{y_j}|\leqslant\varepsilon)$$
$$\Rightarrow Ca_i\text{和}Ca_j\text{斜率密度可达}$$

斜率偏差 δ：一个类簇区间 Ca_i 沿坐标轴正向对角线的斜率的允许偏差程度，即

$$1-\delta\leqslant\frac{l_{y_i}}{l_{x_i}}\leqslant1+\delta$$

3.3.2 基于斜率密度聚类算法的实现

聚类前将每一个匹配点初始化成一个类簇，设包含所有类簇的集合为 D，有 $C_i\in D, 0\leqslant i<n$，n 为 D 中类簇的数量。根据基于斜率密度聚类相关定义，设定聚类规则如下。

规则 1 如果 2 个类簇斜率密度可达，则合并 2 个类簇，重新计算新类簇的类簇区间，将原类簇从 D 中删除。

规则 2 如果 2 个类簇满足类簇包含条件，删除被包含的类簇。

规则 3 退出条件，遍历 D，没有可合并的类簇。

根据规则 3，如果出现类簇合并，新类簇必须与现有的类簇重新进行比较，直至没有可合并类簇。在 n 较大时，算法迭代的效率较低。因此，算法分两步实现：1）顺序聚类，根据相似文本的语义特征，邻近类簇合并的可能性较大，因此首先对所有的初始类簇区间 Ca 按 x 值排序，然后沿 x 轴方向依次根据规则 1 进行聚类；2）迭代合并，在顺序聚类的基础上，再根据规则 1 和规则 2 进行迭代合并，直至没有新的合并产生。算法描述如图 3 所示。

3.3.3 后处理

后处理是根据一定的舍弃条件对聚类结果进行处理，舍弃不符合语义的类簇。可以针对合并后的类簇进行处理，也可以针对还原后的文本片段进行处理。实验表明，针对还原后的文本片段进行处理效果较好。根据相似文本的语义特征，定义类簇舍弃条件如下。

条件 1 类簇斜率超过了斜率偏差 δ 设定的范围。

条件 2 类簇的斜率密度小于预设的阈值。

上述 2 个条件必须综合考虑，如果类簇斜率超过斜率偏差，但其斜率密度高于某个阈值，则仍可判定为相似文本，反之不成立。

4 实验

4.1 实验评价指标

传统的信息检索评价指标无法衡量检测算法对具体抄袭片段的定位精度和效率。因此 PAN 提出一种新的评价指标[13]，以解决相似文本标定算法的效率评价问题。

设 d 是一个抄袭文档。定义 S 为 d 中所有的抄袭片段集合，定义 R 为通过标定算法得到的 d 中所有抄袭片段集合。定义查准率为

```
1) 第一层循环:
2)   从initCList中顺序取出一个初始类簇 initCA
3)   如果 initCA不存在于mergeCList
4)     新建一个合并类簇mergeCA并初始化为initCA
5)   否则
6)     取出包含initCA的合并类簇mergeCA
7)   第二层循环:
8)     从initCList中的initCA之后开始，顺序取出一个初始类簇 initCB
9)     如果initCB属于mergeCA 则跳回第8)步
10)    如果initCB与initCA满足规则1
11)      如果initCB属于合并类簇mergeCB
12)        将mergeCB中所有初始类簇标记为属于mergeCA
13)        将mergeCB从mergeCList中删除
14)      如果initCB不属于任何合并类簇
15)        将initCB标记为属于mergeCA
16)    如果initCB是initCList中最后一个初始类簇
17)      跳出第二层循环
18)    否则跳回第8)步
19)  如果initCA是initCList中最后一个初始类簇
20)    跳出第一层循环
21)  否则跳回第2)步
22)输出mergeCList
```

(a)顺序聚类阶段算法描述

```
1) 布尔变量 hasMerge=true
2) while循环:  当hasMerge为false时退出
3)   设定hasMerge为false
4)   第一层循环:
5)     从mergeCList中顺序取出一个类簇mergeCA
6)   第二层循环:
7)     从mergeCList中mergeCA之后开始，顺序取出一个类簇mergeCB
8)     如果mergeCB和mergeCA满足规则1或者规则2
9)       将mergeCB合并到mergeCA
10)      从mergeCList中删除mergeCB
11)      设定hasMerge为true
12)    如果mergeCB是mergeCList中最后一个类簇
13)      跳出第二层循环
14)    否则跳回第7)步
15)  如果mergeCA是mergeCList中最后一个类簇
16)    跳出第一层循环 到第2)步
17)  否则跳回第5)步
18) 输出mergeCList
```

(b)迭代合并阶段算法描述

图 3 聚类算法描述

$$prec_{\mathrm{PDA}}(S,R)=\frac{1}{|R|}\sum_{r\in R}\frac{|r\cap S|}{|r|}$$

查全率为

$$rec_{\mathrm{PDA}}(S,R)=\frac{1}{|S|}\sum_{s\in S}\frac{|s\cap R|}{|s|}$$

其中，$|S|$、$|R|$ 分别表示 S 和 R 中抄袭片段的个数，$|s|$、$|r|$分别表示 S 和 R 中一个抄袭片段的文本单元数，$|r\cap S|$、$|s\cap R|$表示文本片段 r 和 s 分别与 S 和 R 中重叠的文本单元数。

为了反映干扰信息对集合 R 的影响程度，定义粒度参数为

$$gran_{\mathrm{PDA}}(S,R)=\frac{1}{|S_R|}\sum_{s\in S_R}|C_s|$$

其中，

$$S_R=\{s\mid s\in S\wedge\exists r\in R:s\cap r\neq\phi\}$$

$$C_S=\{r\mid r\in R\wedge s\cap r\neq\phi\}$$

$gran_{\mathrm{PDA}}(S,R)$ 的取值范围为 $[1,|R|]$，$gran_{\mathrm{PDA}}(S,R)$ 值越大说明算法受干扰信息影响越大。

4.2　实验结果及分析

实验基于 PAN 抄袭检测竞赛英文语料集。语料集分为待查文档集和源文档集 2 个部分。相似文本根据干扰程度的不同，分为无干扰（none）、低干扰（low）和高干扰（high）3 种类型，其长度分布在 150~200 000 个字符间。

本次实验以 Character-15-gram 生成指纹，设定移动步长为 1 个字符，抽样窗口为 20，ε 邻域为 50，斜率偏差 δ 为 0.25，斜率密度 ρ 为 0.3。实验结果与 PAN10 竞赛中的前 3 名（KASPRZAK，ZOU，MUHR）进行对比分析。

实验在一个 6 节点 HPC 集群上进行，单个节点是双路 Intel Xeon 4 核 5500 系列处理器，内存 4 GB。语料集共 15 925 个待查文档，每个待查文档约 50 个候选源文档，共进行约 80 万次文档比对。实验启动了 24 个线程，运行时长约 18 min，平均每次比对耗时约 30 ms。不同干扰类型下的实验结果对比如图 4 所示。

实验结果采用 PAN 的查准（precise）、查全

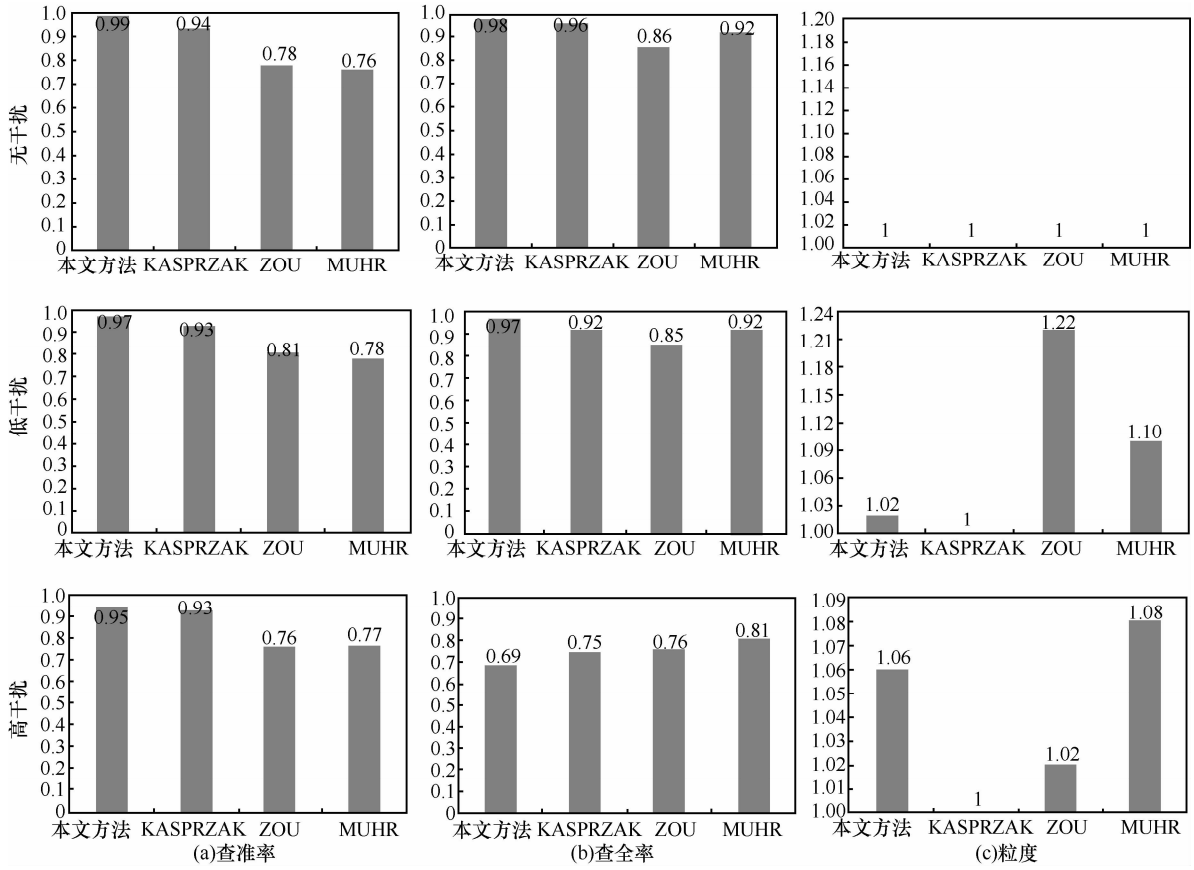

图 4　算法结果比较

（recall）和粒度（granularity）作为评价指标。由图可知，针对无干扰和低干扰相似文本，文中提出的算法的各项指标均较其他方法有优势，针对高干扰相似文本，查全率略低，且粒度较高。

经分析，造成查全率较低的原因有：1）部分仅包含高干扰相似文本的文档未通过预选；2）高干扰相似文本的匹配指纹对分布稀疏，实验采用的聚类参数难以将其识别为同一类簇。粒度较高的原因是匹配指纹对分布稀疏，本应合并为同一片段的文本被拆分成多个片段。

5 结束语

实验证明，文中提出的基于斜率密度的聚类方法在相似文本标定上的运行效果优于其他启发式合并方法。该方法已成功应用于华南理工大学特色专业教学平台，进行作业抄袭检查和标定。在下一步的工作中，将针对不同的聚类参数进行实验和调试，以期得到更好的结果。

参考文献：

[1] MANBER U. Finding similar files in a large file system[A]. Proceedings of the Winter USENIX Conference[C]. San Francisco, CA, 1994.1-10.

[2] BRIN S,DAVIS J, GARCIA-MOLINA H. Copy detection mechanisms for digital documents[A]. Proceedings of the ACM SIGMOD Conference on Management of Data[C]. ACM, NY, USA, 1995. 126-141.

[3] BRODER A Z. On the resemblance and containment of documents[A]. Proceedings of Compression and Complexity of Sequences 1997[C]. Salemo, Italy, 1997. 21-29.

[4] BRODER A Z. Identifying and filtering near-duplicate documents[A]. Proceedings of Combinatorial Pattern Matching[C]. Springer Berlin Heidelberg, Germany, 2000. 1-10.

[5] STEIN B, POTTHAST M. Applying Hash-based indexing in text-based information retrieval[A]. Proceedings of The 7th Dutch-Belgian Information Retrieval Workshop (DIR 07)[C]. Leuven, Belgium, 2007. 29-35.

[6] CHARIKAR M S. Similarity estimation techniques from rounding algorithms[A]. Proceedings of The Thiry-Fourth Annual ACM Symposium on Theory of Computing[C]. ACM, NY, USA, 2002. 380-388.

[7] SHIVAKUMAR N, GARCIA-MOLINA H. Building a scalable and accurate copy detection mechanism[A]. Proceedings of The First ACM International Conference on Digital Libraries[C]. ACM, NY, USA, 1996. 160-168.

[8] SCHLEIMER S, WILKERSON D S, AIKEN A. Winnowing: local algorithms for document fingerprinting[A]. Proceedings of The 2003 ACM SIGMOD International Conference on Management of Data[C]. ACM, New York, 2003. 76-85.

[9] SEDIYONO A, MAHAMUD K R K. Algorithm of the Longest Commonly Consecutive Word for Plagiarism Detection in Text Based Document[A]. Proceedings of The Third International Conference on Digital Information Management[C]. London, UK, 2008.253-259.

[10] ZASLAVSKY A, BIA A, MONOSTORI K. Using copy-detection and text comparison algorithms for cross-referencing multiple editions of literary works[C]. Research and Advanced Technology for Digital Libraries. Springer Berlin Heidelberg, Germany, 2001.103-114.

[11] KASPRZAK J, BRANDEJS M, KRIPAC M. Finding plagiarism by evaluating document similarities[A]. 3rd PAN Workshop Uncovering Plagiarism, Authorship and Social Software Misuse[C]. 2009. 32-36.

[12] ZOU D, LONG WJ, ZHANG L. A two-phase plagiarism detection method[A]. 2011 International Conference on Internet Technology and Applications[C]. Wuhan, China, 2011. 1-4.

[13] POTTHAST M. Overview of the 1st International Competition on Plagiarism Detection[A]. 3rd PAN Workshop Uncovering Plagiarism, Authorship and Social Software Misuse[C]. 2009. 1-9.

作者简介：

邹杜（1973-），男，湖南湘潭人，硕士，华南理工大学高级工程师，主要研究方向为计算机网络应用、信息检索。

唐文军（1971-），男，河南开封人，硕士，华南理工大学讲师，主要研究方向为计算机网络应用、网络安全。

龙卫江（1962-），男，湖南常德人，博士，华南理工大学副教授，主要研究方向为信息检索、模式识别。

张凌（1962-），男，江西宜春人，华南理工大学教授、博士生导师，主要研究方向为计算机网络系统管理与网络安全、下一代高性能计算机网络、计算机海量信息处理与电子商务技术等。