摘　要

随着互联网的发展，大量近似重复的文本广泛存在于现实世界中，如何检测这些近似重复的文本成为了一个研究的热点问题，这一技术在不同领域存在着很多应用：数字图书馆中相似内容的自动链接、数字知识产权保护（剽窃检测）、近似重复网页检测（搜索引擎优化）、数据去重、垃圾邮件检测等。采用传统的哈希算法（SHA1、MD5 等）只能鉴别完全相同的文档，不适用于存在细微修改的近似文档。当前，近似重复检测的主要方法是生成文本指纹，通过计算文本间文本指纹的距离，衡量文本的相近程度。

本文在研究该领域的三种代表性算法（shingling、I-Match、simhash）的基础上，提出了融合这些算法优点的改进算法并进行了系统实现和验证，主要工作包括三个方面：

（1）.　提出了基于 shingle 特征的 simhash 算法。Shingling 算法以连续词串作为特征，有利于提高检测的准确率，但生成指纹集合、计算集合基于 Jaccard 相似度的距离，计算量大。Simhash 算法以指纹间的汉明距离度量相似性，计算量小，且指纹占用空间小。但 simhash 算法以单词为特征，不能很好的表征文档的语义。本文将 shingles 作为 simhash 算法的输入特征，以提高 simhash 算法的准确率。

（2）.　提出了基于随机词典的多指纹 simhash 算法。I-Match 算法完全依赖单词的 IDF 值去除近似重复文本间的不同单词，检测的召回率很低。基于随机词典的 I-Match 算法提出利用原始文档集的词典随机生成多个子词典，子词典分别过滤文档，生成多个 I-Match 指纹，以提高 I-Match 方法的稳定性。对于同样是生成单指纹比对的 simhash 算法，本文引入基于随机词典的 I-Match 算法的提高召回率的方法，以提高 simhash 算法的召回率。

（3）.　以“中美百万册数字图书馆”中的图书数据构建了一个近似重复文本检测数据集，对上述两种改进算法在该数据集上进行了的实验验证。在最优参数、F-measure 的度量上，基于 shingle 特征的 simhash 算法的 0.7469 比原 simhash 算法的 0.6117 提高了 22%；融合算法的 0.8805 比基于 shingle 特征的 simhash 算法的 0.7469 提高了 18%，比原始的 simhash 算法提高了 43%。实验表明两点改进思路对相应性能的提升都得到了验证，最终的融合算法比原始 simhash 算法在 F-值度量上有较大提升。

本文认为，取得如此性能提升的主要原因是，依据三种经典算法的特点，进行了有针对性的融合，改进了 simhash 算法的特征选择策略和指纹生成策略，分别有利于 simhash 算法准确率和召回率的提升。

关键词：近似重复文本检测、网页去重、simhash 算法

Document fingerprint and its application in near duplicate document detection

Jun Fan Microelectronics
Directed by Tie Junhuang

With the rapid development of the World Wide Web, dissemination reproduced or plagiarism other's literature with or without modification has become very easy. There are a huge number of these kinds of duplicated documents in the real world. How to detect these near duplicate documents has become a hot research topic. There is a wide range of applications. Such as: Automatically link of duplicate document in the digital library, protection of intellectual property (or called plagiarism detection), near duplicate web page detection (one kind of search engine optimization technique), data deduplication, spam detection. Traditional Hash algorithms like SHA1, MD5 can only detect documents exactly the same or not. They can't handle documents with minor modifications. The main method in near duplicate document detection is generating document fingerprints, measure the similarity of documents through the distance of the corresponding document fingerprints.

In this article, we described the three "state of art" algorithm (shingling, I-Match, simhash) in detail. We did some fusion based on the characters of each class of algorithms mentioned above, implemented a system and some experiments. Our works are:

1. Shingling based simhash algorithm: the input feature of shingling algorithm is k-shingles (word sequences of length k), it is benefit for the precision of detection. But the measure of distance of fingerprints is Jaccard similarity of set, have a high computational complexity. The distance of fingerprints in simhash algorithm is hamming distance; it is low in computational complexity, and small in space. But the input feature of the simhash algorithm is words of the document; it can't represent the document well. In this article, we use the k-shingles (word sequences of length k) as the features of the simhash algorithm to improve

precision of simhash algorithm.

2. Multiple random lexicons based simhash algorithm: the effectiveness of the I-Match algorithm is based on filtering different words in near duplicate documents by IDF values of the words totally. It has a low recall. The multiple random lexicons based I-Match algorithm filter documents by randomly created lexicons and generate multiple fingerprints to improve the stability of the I-Match algorithm. This method is applicable to other single-signature based algorithm, such as simhash. We filter documents by randomly created lexicons and generate multiple simhash fingerprints to improve recall.
3. We construct a near duplicate document detect dataset based on the books in the "China-US Million Book Digital Library Project". We tested our algorithms in this synthetic dataset. With the best parameters' set and in the F-measure's view, from the shingling based simhash algorithm to the simhash algorithm, we get a 22% improvement from 0.7469 to 0.6117. From the fusion algorithm to the shingling based simhash algorithm, we get an 18% improvement from 0.8805 to 0.7469. Our fusion algorithm gets a 43% improvement compared with the simhash algorithm in total. The experiment result proves the efficiency of the above two algorithms. The fused integrated algorithm performs much better than the original simhash algorithm in the F-measure's view.

With such an improvement, credit to the targeted fusion based on the characters of each algorithms. We improved the feature selection strategy and the fingerprint generation strategy of the simhash algorithm, which help to improve precision and recall correspondingly.

Keywords: near duplicate document detection、 near duplicate web page detection、 simhash algorithm