

一种图文组合相似度算法的设计与优化

鲜翠琼, 秦 学, 朱道恒, 操淑敏

(贵州大学大数据与信息工程学院, 贵州 贵阳 550025)

✉1243139443@qq.com; 16702755@qq.com; dhzhu911@163.com; 982913044@qq.com



摘 要: 包含文字和图片的文档作为信息的一种载体, 能够极大地丰富信息的表现形式。针对传统计算图文相似度的算法效率不高的问题, 提出一种图文组合相似度算法。将Jaccard相似系数引入余弦相似度, 通过加权计算两文本的相似度, 然后用感知哈希算法计算文档中图片相似度并找出最大值, 再计算单个文档中所有图片相似度均值, 与文本相似度加权求得文档的图文相似度。最后通过一个文档相似度查重系统验证了该算法能准确高效地完成文档之间相似度的量化, 且优化后的相似度算法能够极大提高该系统的运行效率。

关键词: 余弦相似度算法; Jaccard相似系数; 感知哈希算法; 文本相似度

中图分类号: TP391.1 **文献标识码:** A

Design and Optimization of a Similarity Algorithm for Combination of Graphics and Text

XIAN Cuiqiong, QIN Xue, ZHU Daoheng, CAO Shumin

(College of Big Data and Information Engineering, Guizhou University, Guiyang 550025, China)

✉1243139443@qq.com; 16702755@qq.com; dhzhu911@163.com; 982913044@qq.com

Abstract: As a carrier of information, documents containing both text and graphics can greatly enrich information. In view of the inconsistency of traditional algorithms for calculating similarity between graphics and text, this paper proposes a similarity algorithm for combining graphics and text. Jaccard similarity coefficient is introduced into cosine similarity, and the similarity of two kinds of text are calculated by weighting. Then, the graphic similarity in the document is calculated through PHash algorithm and the maximum value is derived. After that, the average similarity of all the graphics is calculated in a single document, and weighted with the text similarity, thus to obtain the similarity of both graphics and text of document. Finally, a document similarity check system is used to verify that the algorithm accurately and efficiently quantifies the similarity between documents, and the optimized similarity algorithm greatly improves the efficiency of the system.

Keywords: cosine similarity algorithm; Jaccard similarity coefficient; PHash similarity; text similarity

1 引言(Introduction)

文本相似度分析是中文信息处理领域的一种基础技术, 凡是在处理中文信息处理的领域都有其身影。机器翻译领域, 文本相似度可作为翻译准确度的衡量原则^[1]; 搜索引擎领域, 文本相似度算法是其核心; 自动问答领域, 文本相似度更是匹配回答与提问的重要技术^[2]; 在论文查重领域, 文本相似度研究更是系统检测学术不端行为的唯一手段; 在文本聚类领域, 文本相似度也是聚类表中的重要参考^[3]; 在新闻推荐、关联词推荐等智能精准推荐领域, 文本相似度研究更是

其基础工作^[4]。而在计算机视觉领域, 图像相似度度量是一个基础问题, 它在图像分类、图像检索、图像匹配、模式识别和目标检测等多方面都有十分广泛的应用。因此, 对图像文本相似度度量的研究在诸多领域都有非常重要的意义。

2 相关算法(Correlation algorithm)

2.1 余弦相似度

余弦相似度又称为余弦相似性, 是通过测量两个向量的夹角的余弦值来度量它们之间的相似性。利用两个向量之间的角度的余弦值确定两个向量是否大致指向相同的方向。

将一个字符串通过分词工具分成一个个词语项,将这些词语,映射到向量空间,形成字符串和向量数据的映射关系。利用词语词频映射字符串向量。通过计算对应向量的余弦值来表示两个文本字符串在统计学方法中的相似度。

给定两个向量 A 和 B ,其余弦相似性 θ 由点积和向量长度给出,如公式(1)所示^[5]:

$$\text{similarity} = \cos \theta = \frac{\vec{a} \cdot \vec{b}}{|\vec{a}| |\vec{b}|} = \frac{\sum_{i=1}^n a_i \times b_i}{\sqrt{\sum_{i=1}^n (a_i)^2} \times \sqrt{\sum_{i=1}^n (b_i)^2}} \quad (1)$$

2.2 Jaccard相似系数

通过计算文本中两个集合交集的大小除以并集的大小来评估它们的相似度即Jaccard相似系数(Jaccard Similarity Coefficient)^[6],用 $J(A, B)$ 表示。与Jaccard系数相关的指标叫Jaccard距离,用于描述集合之间的不相似度。Jaccard距离越大,样本相似度越低^[7]。定义如式(2):

$$d_j(A, B) = 1 - J(A, B) = \frac{|A \cup B| - |A \cap B|}{|A \cup B|} \quad (2)$$

2.3 图片相似度算法

目前对于图片相似度的计算,一般采用Hash算法^[8]、SIFT(Scale-Invariant Feature Transform, 尺度不变特征变换)算法^[9]、MD5值比较算法^[10]、像素对比算法^[11]、直方图比较算法^[12]。表1对比了这几种算法计算图片相似度的特点。

表1 几种算法特点对比

Tab.1 Comparison of several algorithms

算法	优缺点
Hash算法	正向快速、逆向困难、输入敏感、冲突避免
MD5比较算法	容易计算、压缩性、不可恢复、抗碰撞性
SIFT算法	准确率高、原理复杂、计算烦琐、实时性不高
像素对比算法	一般用于原图查找、性能不佳、效率不高
直方图比较算法	信息损失量大、误判率高、效率不高
PHash算法	健壮性好、简单快速、不受图片大小限制

对比表中几种算法的特点后选用PHash(感知哈希)算法作为本文图片相似度计算方法。PHash算法是基于DCT(离散余弦变换)来得到图片的Hash值,该算法分为以下几个步骤:

(1)缩放图片:统一将图片尺寸缩放为32*32,一共得到了1024个像素点。

(2)转灰度图:统一下一步输入标准,将非单通道图都转为单通道灰度图。

(3)计算DCT(离散余弦变换):计算32*32数据矩阵的离散余弦变换后对应的32*32数据矩阵。

(4)缩小DCT:取上一步得到32*32数据矩阵左上角8*8子区域。

(5)计算平均值:通过上一步可得一个8*8的整数矩阵 G ,计算这个矩阵中所有元素的平均值,记为 a 。

(6)计算Hash值:根据8*8的DCT矩阵,设置0或1的64位

的hash值,大于等于DCT均值的设为1,小于DCT均值的设为0。组合在一起构成一个64位的整数,即为这张图片的指纹^[13]。

得到图片的PHash值之后比较两图PHash值的汉明距离,一般认为只要汉明距离超过10,两张图片就基本不相似了。

3 图文组合相似度算法设计(Design of similarity algorithm for combination of graphics and text)

3.1 加权相似度

一般文档由文本和图片组成,由于文本中专业词汇容易反复出现,而Jaccard相似度只做集合运算,运算效率比余弦相似度高,故在余弦相似度算法中引入Jaccard相似系数不仅能有效平衡余弦相似度算法的不足,还可节省系统开销。

将余弦相似度算法求的结果与Jaccard相似系数算法所求结果的加权值作为文本相似度,再由文本相似度与PHash算法求得的所有图片相似度均值加权求出两份文档的相似度值,作为最终的文档相似度。

3.2 图文组合算法

上述算法仅针对两个字符串之间或两张图片之间的相似算法,如果要实现对存储在本地某一路径下的所有Word文档或者TXT文本进行相似度比较,需遍历所有文档之后再调用上述算法计算。先设计算法将Word文档中图片提取到本地新建文件夹中存储,然后设计算法遍历指定路径下所有文档和图片,最后设计算法来计算文档图文相似度如图1所示。

```

功能: 实现遍历比较文档图文相似度
输入: 待比较的文档
输出: 文本相似度和图片相似度
1  for 0 to 总文档数 N-2 {
2   for 1 to 总文档数 N-1 {
3    读取文件
4    分词
5    计算文档1与文档2的文本相似度  $Sim_T$ 
6    if 需要比较图片相似度 {
7      遍历两份文档各自图片的路径
8      for 0 to 第一份文档图片数-1 {
9        for 0 to 第二份文档图片数-1 {
10         计算图片相似度  $Sim_p$ 
11         for 循环 求单张图片相似度最
            大值  $Sim_{p_2}$ 
12         计算单个文档图片相似度均值  $\overline{Sim_p}$ 
13       }
14     }

```

图1 文档图文相似度比较算法

Fig.1 Document similarity comparison algorithm

根据式(1)和式(2),将Jaccard相似系数引入余弦相似度,定义一种两文本间的相似度公式(3):

$$Sim_T = \frac{w_1 \cos \theta + w_2 J(A, B)}{w_1 + w_2} \quad (3)$$

式中 $\cos \theta$ 为向量化的文本字符串间相似度, $J(A, B)$ 为Jaccard相似系数, w_1 、 w_2 分别为余弦相似度和Jaccard相似系数的权重系数,且 $0 < w_1$ 、 $w_2 < 1$, $w_1 + w_2 = 1$ 。

根据前文的图片相似度计算方法,将PHash算法中得到的两图片间明文距离定为 $d(x_A, y_B)$,由于计算图片的Hash值为一个64位的整数,则图片相似度可用式(4)计算:

$$Sim_p = 1 - \frac{d(x_A, y_B)}{64} \quad (4)$$

$$d(x_A, y_B) = \sum_{i=0}^{n-1} x[i] \oplus y[i] \quad (5)$$

其中, $d(x_A, y_B)$ 表示两图片PHash值的汉明距离, x 、 y 是具有相同位数的编码。

计算单个文档中所有图片相似度的均值如式(6):

$$\overline{Sim_p} = \frac{\sum_{i=1}^n Sim_{p_{im}}}{n} \quad (6)$$

式中, $Sim_{p_{im}}$ 表示单张图片相似度的最大值。将式(3)和式(6)中得到的相似度加权, 定义图文相似度计算公式(7)如下:

$$Sim_{PT} = \frac{w_3 Sim_T + w_4 \overline{Sim_p}}{w_3 + w_4} \quad (7)$$

式中, w_3 、 w_4 分别为文本和图片相似度的权重系数, 并且 $0 < w_3$ 、 $w_4 < 1$, $w_3 + w_4 = 1$ 。

3.3 算法优化

上述算法是将所有图片依次与其他图片对比, 对比完成后才处理结果, 这样在图片重复度较高时会增加很多额外开销。故在图1所示的比较算法的10和11行间增加一个判断语句, 其余部分不变, 如图2所示。

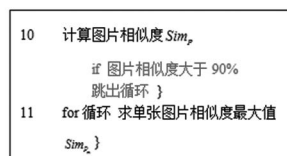


图2 减少图片计算次数的优化算法

Fig.2 An optimized algorithm for reducing the number of image calculations

改进算法后系统运行效率大幅提升, 但仍受限于文档中相似图片的顺序及数量。分析模块调用算法, 发现每组数据中只有几十到几百张图片, 但图片计算次数从几百到几十万次。比较算法每计算一次就要对图片进行一次PHash处理, 单次PHash处理很快, 但几十万次PHash处理就会大大加程序运行时间。故考虑在图片相似度计算发生前完成所有图片的PHash处理, 并将处理好的Hash指纹存储起来。计算图片相似度时不再调用Phash算法对图片进行DCT等计算, 直接读取已经计算好的图片Hash指纹。进一步优化比较算法如图3所示。

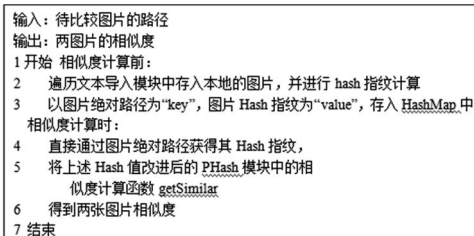


图3 减少图片处理次数的优化算法

Fig.3 An optimization algorithm for reducing the number of image processing

4 实验验证与分析(Experimental verification and analysis)

4.1 系统设计

设计一种文档相似度查询系统, 应用到判定作业是否抄袭的场景。系统主要包含六个模块: 文件导入模块、文本相似度模块、图片相似度模块、比较模块、输出模块、可视化界面模块, 如图4所示。系统可自动识别txt文档编码格式, 自动分离Word中图片和文字信息, 然后按前面介绍的方法对图片和文字分别计算相似度, 加权求出文档整体相似度, 最后将全部比较数据输入本地csv文档中。此外可视化界面让用户使用时更加方便, 只需把待比较文件放入同一文件夹下, 将路径传入程序, 便能自动计算文件夹内所有文档的相似度, 最后将结果导入到csv文件。

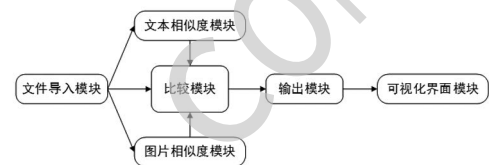


图4 系统模块框图

Fig.4 System module block diagram

4.2 测试结果分析

选四组数据(一组人工制造的有特殊相似度分布的数据加三组真实作业数据)进行测试。第一组为两份不同的文档分别去除图片和保留图片放入待查重目录进行查重测试, 产生特殊相似度分布的结果, 方便检查程序能否正常工作。另三组为实验报告合集, 导入系统得到测试结果如表2—表5所示。

表2 第一组测试数据性能测试结果

Tab.2 Performance test results of the first group of test data

测试数据	打开智能分词	打开图片查重	打开排序输出	耗时/秒	文档数	图片比较次数	文字处理数
第1组	✓			1			
		✓		7			
			✓	1			
	✓	✓		7	3	48	15261
	✓		✓	1			
		✓	✓	7			
	✓	✓	✓	8			

表3 第二组测试数据性能测试结果

Tab.3 Performance test results of the second group of test data

测试数据	打开智能分词	打开图片查重	打开排序输出	耗时/秒	文档数	图片比较次数	文字处理数
第2组	✓			25			
		✓		248			
			✓	24			
	✓	✓		263	48	5700	1458047
	✓		✓	24			
		✓	✓	246			
	✓	✓	✓	250			

表4 第三组测试数据性能测试结果

Tab.4 Performance test results of the third group of test data

测试数据	打开智能分词	打开图片查重	打开排序输出	耗时/秒	文档数	图片比较次数	文字处理数
第3组	✓			52			
		✓		1597			
			✓	51			
	✓	✓		1476	50	4489	93326
	✓		✓	55			
		✓	✓	1512			
	✓	✓	✓	1523			

表5 第四组测试数据性能测试结果

Tab.5 Performance test results of the fourth group of test data

测试数据	打开智能分词	打开图片查重	打开排序输出	耗时/秒	文档数	图片比较次数	文字处理数
第4组	✓			45			
		✓		3939			
			✓	42			
	✓	✓		3847	50	276150	4595751
	✓		✓	52			
		✓	✓	3975			
	✓	✓	✓	3925			

取第二组数据的测试结果作准确性测试如图5所示，得对比详情如图6所示。

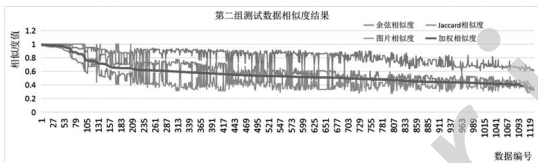


图5 第二组测试数据完整结果曲线图

Fig.5 Complete result curve of the second group of test data

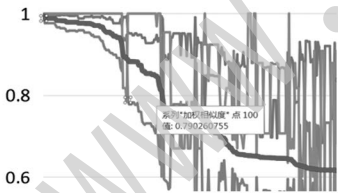


图6 第二组测试结果局部截图

Fig.6 Partial screenshot of the second group of test results

从图中看出，经组合算法加权后的值更加光滑地表现了相似度分布，且处于各算法的中间值，趋于稳定。从结果判断具有极高相似度的数据量较小，随数据量的增大，相似度在小范围内迅速下降，最后趋于平缓，符合实际情况。经过人工比对加权相似度超过80%的99条比对结果，发现大部分内容几乎完全相同。在总加权相似度低于0.8但图片相似度等其他三条相似度中两条值偏高的文档中，部分内容也存在删减性雷同。故实际使用时，判定抄袭的具体阈值还需实际作业类型来确定。一个字数量庞大的作业，经系统测试，当加权相似度大于80%，或其他几项文本相似度大于95%或图片相

似度大于90%才判定为抄袭的原则下，判定结果99.9%是准确的。

4.3 算法优化后的测试结果

算法优化后再次测试上述四组数据，优化前后运行时间对比如图7—图10所示。

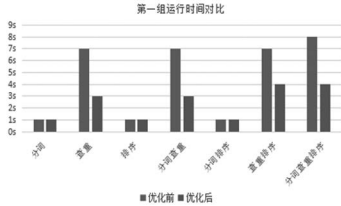


图7 第一组数据运行时间对比

Fig.7 Running time comparison of the first group of data

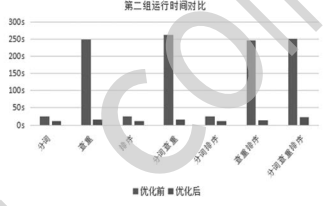


图8 第二组数据运行时间对比

Fig.8 Running time comparison of the second group of data

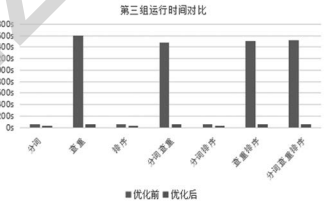


图9 第三组数据运行时间对比

Fig.9 Running time comparison of the third group of data

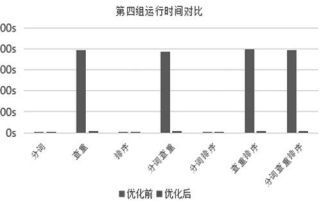


图10 第四组数据运行时间对比

Fig.10 Running time comparison of the fourth group of data

从图7至图10可知，系统经过优化后，图片处理次数从几十万次降至几百次，极大地缩短了系统的运算耗时。虽然会增加几十万次的读取操作，但这个代价完全不值一提。测试数据整体运行效率明显提高，验证了改进算法的有效性。

5 结论(Conclusion)

针对传统计算图文相似度的算法效率低的问题，本文提出一种图文组合相似度算法。通过将Jaccard相似系数引入余弦相似度通过加权计算文本间的相似度，再与PHash算法求得的图片相似度均值加权后得到全文相似度，并对组合相似度算法作出优化。最后在一个文档相似度查重系统中验证了该算法的准确性和可行性，并且运行相似度优化算法后，系统整体运行效率得到极大提升。未来可以考虑引入词频权重

(下转第4页)