A Component Histogram Map Based Text Similarity Detection Algorithm

Huajun Huang, Shuang Pang, Qiong Deng, and Jiaohua Qin
(Corresponding author: Huajun Huang)

College of Computer and Information Engineering, Central South University of Forestry and Technology
498 Shaoshan South Road, CHangsha, Hunan Province 410004, China
(Email: hhj0906@163.com)

(Received Apr. 10, 2015; revised and accepted May 16 & May 24, 2015)

Abstract

The conventional text similarity detection usually use word frequency vectors to represent texts. But it is high-dimensional and sparse. So in this research, a new text similarity detection algorithm using component histogram map (CHM-TSD) is proposed.This method is based on the mathematical expression of Chinese characters, with which Chinese characters can be split into components. Then each components occurrence frequency will be counted for building the component histogram map (CHM) in a text as text characteristic vector. Four distance formulas are used to find which the best distance formula in text similarity detection is. The experiment results indicate that CHM-TSD achieves a better precision, recall and F1 than cosine theorem and Jaccard coefficient.

Keywords: Component histogram map, distance calculation, text similarity detection

1 Introduction

As a branch of natural language processing, text similarity detection is more and more important for information security. It has been used in many fields such as information retrieval (IR), duplicated detection, Data clustering and classification [3]. In general, there are two ways for text similarity detection, one is that based on semantic similarity, and the other one is non-semantic. Semantic similarity detection usually based on dictionary computation like HowNet [13] and WordNet [4]. Huang has ever proposed a method that combined the external dictionary with TF-IDF to compute text similarity [5]. Some people also use a large-scale corpus for semantic similarity detection [7], but its uncommon because of its disadvantages. Non-semantic similarity detection mostly uses word frequency statistics and string comparison two methods. The most common used methods of word frequency statistics are VSM [11, 12] the text similarity can be computed through cosine [14] theorem or Jaccard coefficient [10]. In the other hand, Shingling [15] and maximum string matching algorithm [6] is often used for string comparison. All of the methods above performance well in certain situations, but there are also some shortcomings. For examples, the semantic method based on dictionary is too depending on person and the knowledge library to express the sense of a word exactly. Word frequency statistics is very high-dimensional and sparse [8].

From the above, a new Chinese text similarity detection method was proposed. This method used CHM (component histogram map) to avoid high-dimensional and sparse problem. Mathematical expression of Chinese characters [9], used to split Chinese characters into components was the basic theorem for this method. And the components were taken as research object. Components are correlated with each other to compose Chinese characters, so these components are correlative. CHM was built with each components occurrence frequency. Then the distance between text and duplicate text is calculated with Bhattacharyya formula. From the results, we can see that CHM-TSD performance better than cosine theorem and Jaccard coefficient.

2 Related Theories

In the process of text duplicate detection, text feature representation and similarity detection are two very important steps [12]. VSM is the most common method for text feature representation. Assuring d_i is the i-th text, W_{ij} is the weight of the j-th word of d_i, then the i-th text can be represented as $\vec{d_i} = (W_{i,1}, W_{i,2}, \cdots, W_{i,3})$, so all the texts in the experiment can compose a vector space $D = (\vec{d_1}, \vec{d_2}, \cdots, \vec{d_n})$. The similarity of each pair of text can be computed as two vectors distance through cosine

theorem. The formula is as follows:

$$\begin{aligned} sim(d_i, d_j) &= \cos(\theta) \\ &= \frac{\vec{d_i} \cdot \vec{d_j}}{\left\|\vec{d_i}\right\| \times \left\|\vec{d_j}\right\|} \\ &= \frac{\sum_{t=1}^{n} W_{i,t} \times W_{j,t}}{\sqrt{\sum_{t=1}^{n} W_{i,t}^2} \times \sqrt{\sum_{t=1}^{n} W_{j,t}^2}}, \end{aligned}$$

where $\left\|\vec{d_j}\right\|$ is norm of d_i. The value of cosine similarity between two vectors is between 0 and 1, 0 indicates the two texts are different and 1 indicates they are the same.

In the Chinese character library, there are 6763 common Chinese characters which encoded with Gb-2312. And all these Chinese characters can be combined to thousands of words and even more. For example, the word segmentation software of Chinese Academy of Science (ICTCLAS), has extracted 130,000 commonly used words from the corpus provided by Sogo lab [8]. So it is obvious that the number of Chinese word in a corpus needs to be counted is quite big. This leads to high-dimensional and sparse vectors space. Therefore, a new text representation method based on component relation map has been proposed.

A mathematical expression of Chinese characters of a Chinese character is a formula for splitting Chinese characters into components. It composes of operators and components. Component is a part of a Chinese character and composes of strokes. Every component has a corresponding number as its identifier. There are two kinds of components, one is the ordinary components, and the other called composed components consist of two or more components by certain structural rules. As shown in Figure 1. Chinese characters are formed with the components by different structural rules [9].

There are six operators of the mathematical expression of Chinese characters, $lr(left\ right)$, $ud(up\ down)$, $we(whole\ embody)$, $lu(left\ up)$, $ld(left\ down)$, $ru(right\ up)$. All these operators represent the combination mode of components. As shown in Figure 1, the rectangle A and B are components [9]. $A\ lr\ B$ means that A is on the left and B is on the right. It has two results, a composed component and a Chinese character.

As mentioned above, Chinese characters are compose of components and correlated rules. In this research, we have selected 505 components which can form all the common used Chinese characters as research objects. As Chinese characters increasing, the number of each component will increase clearly, but the number of kinds of components won't. Figure 2 gives some examples of mathematical expression of Chinese characters.

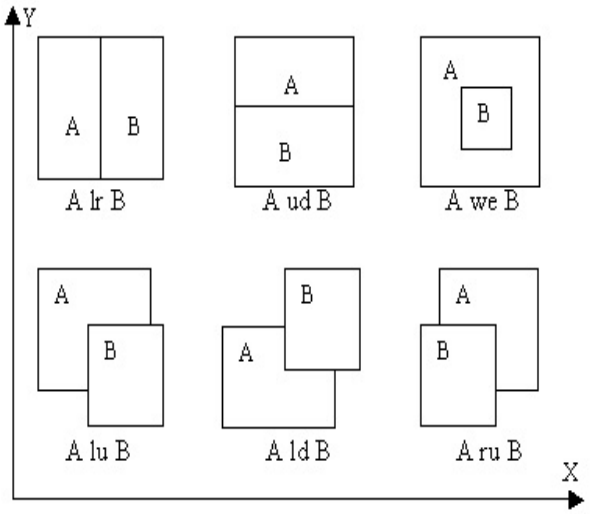

Figure 1: Intuitive description of the operators

Chinese Characters	Mathematical Expression
膘	124 lr(203 ud 142)
彬	86 lr 86 lr 435
勰	447 lr(5 ud 303 ud 67)lr 16
丙	1 ud 100
蚕	95 ud 209

Figure 2: Mathematical expression of Chinese characters

3 Text Duplicate Detection Model

3.1 Detection Model

Text duplicate detection model divides into three modules: 1)text preprocessing. 2) build the component histogram map. 3) Calculate the distance between text in database and detected text. The framework of this model is as shown in Figure 3. When two texts are prepared, the number, English characters, and the stop and useless words are deleted first. So there are only Chinese characters retained. After the preprocessing, all Chinese characters in texts are split into components through the mathematical expression of Chinese characters. Then the occurrence frequency of each component will be counted for building the histogram maps. At last, all the component histogram maps of each pair of texts are matched to get the text similarity. The core module of this model is

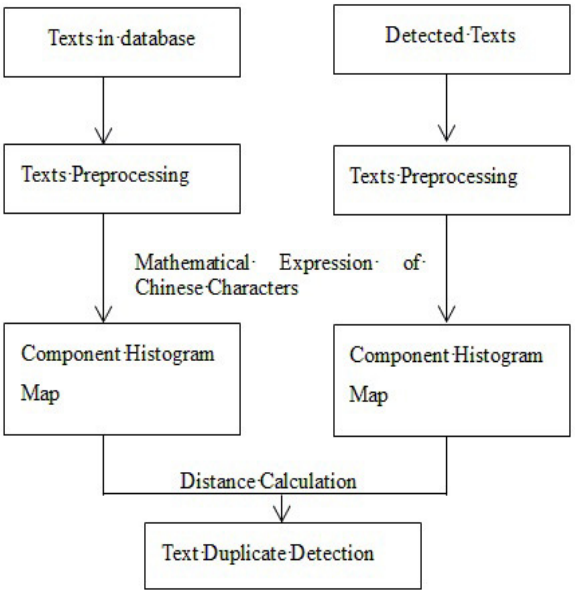

Figure 3: The text duplicate detection model

text duplicate detection using component histogram map.

3.2 Component Histogram map

Rule 1. Assuring that c denotes component and T is a text, then c_i is the ith component and the text T can be regarded as a set of components, so $T = \{c_1, c_2, c_3, \cdots, c_i, \cdots, c_n\}$.

Rule 2. T is the preprocessing text, W is a set of words appearing in text T. Ω is a basic component. w_i is an element of W. $t(c, w_i)$ is the number of c appear in word w_i. $N(c)$ denotes component c appear in text T:

$$N(c) = \sum_{w_i \in \Omega} N(w_i) \times t(c, w_i).$$

Definition 1. *Component histogram map is defined as* $H = \{f_{c_1}, f_{c_2}, \cdots, f_{c_n}\}$, *where* f_{c_i} *is the frequency of the* i*th component in text* T. *Figure 4 is a CHM of the text extracted from experiment data corpus.* f_{c_i} *is counted in the following.*

$$f(c_i) = \frac{N(C_i)}{\sum_{i=1}^{n} N(c_i)}.$$

From the definition of component histogram map, there are some properties.

Property 1. A component histogram map only reflects components frequency in a text. The location of a component appear in the text do not depict in the map.

Property 2. The mapping relation between component histogram map and text is many to one. A text only has a component histogram map, but different texts may have the same component histogram map.

Property 3. A sub component histogram map in a text can into the whole map.

As the properties are shown in the former section, this method may bring a false negative, which a text is not a duplicated text, but it is detected as a duplicated one.

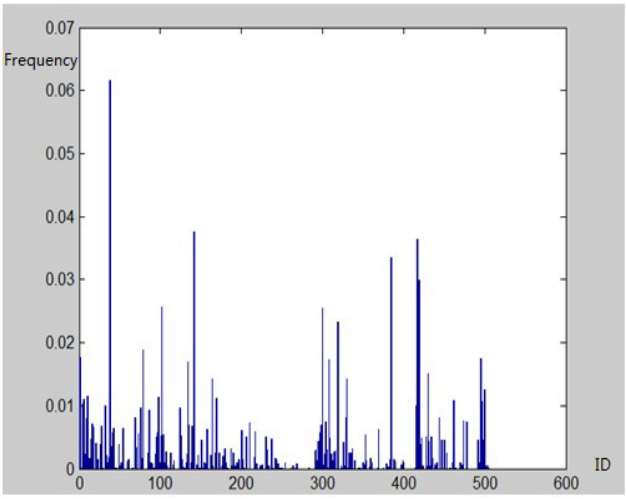

Figure 4: Component histogram map

3.3 Distance Calculated

Text feature representation and similarity detection are two very important steps in the process of text duplicate detection. We use the component histogram map as feature representation. Now, we take the similarity detection into account. To two feature vectors, the common method is distance calculation between the two vectors. So we choose distance calculation to measure the similarity between the two texts.

Assuming text $T1$ and text $T2$, the component histogram map is denote $H1$ and $H2$ by each. So the distance between $T1$ and $T2$ can be denoted as follows.

$$D = Dis(H1, H2).$$

If the value of D is equal to 0, the texts are complete similarity. If the value of D is equal to 1, the texts are completely different. Others may use a threshold α to determine the texts are belonging to. In order to reduce the false positive and false negative, we select fours distance calculation formulas, Correlation, Chi-Square, Intersection, and Bhattacharyya. In the following section, we will show which is the best distance calculation formula used in our method. The four formulas are shown as follows.

Correlation:

$$D(H_1, H_2) = 1 - \frac{\sum_{i=1}^{n}(H_1(i)-\bar{H_1})(H_2(i)-\bar{H_2})}{\sqrt{\sum_{i=1}^{n}(H_1(i)-\bar{H_1})^2)\sum_{i=1}^{n}(H_2(i)-\bar{H_2})^2)}}.$$

Chi-Square:

$$D(H_1, H_2) = \sum_{i=1}^{n}\frac{(H_1(i)-H_2(i))^2}{H_1(i)+H_2(i)}.$$

Intersection:

$$D(H_1, H_2) = 1 - \frac{\sum_{i=1}^{n} min[H_1(i), H_2(i)]}{\sum_{i=1}^{n} H_1(i)}.$$

Bhattacharyya:

$$D(H_1, H_2) = \sqrt{1 - \frac{\sum_{i=1}^{n}\sqrt{H_1(i)\cdot H_2(i)}}{\sqrt[n]{\bar{H_1}\bar{H_2}}}},$$

where $\bar{H_k} = \frac{\sum_{i=1}^{n} H_k(i))}{n}$.

4 Experiments Results and Performance Analysis

4.1 Performance Analysis

The experiment text corpus includes 200 pair entries collected from the Internet. 200 entries are collect from Baidu [1] and the same entries come from Baike [2]. The experiment tools include MATLAB 7.0 and C#.

In this research, we use precision P, recall R and $F1$-Measure for analyzing the results of experiment. This three indexes are most commonly used in the field of information retrieval and natural language processing. Precision is the fraction of retrieved instances that are relevant, while recall is the fraction of relevant instances that are retrieved. $F1$ is a synthesis evaluation parameter of precision and recall. The specific calculation formulas are as follows:

$$\begin{aligned} P &= \frac{truepositives}{truepositives + falsepositives} \\ R &= \frac{truepositives}{truepositives + falsenegatives} \\ F1 &= \frac{2 * P * R}{P + R}. \end{aligned}$$

4.2 Experiment Results And Analysis

Firstly, we use the four distance formulas to calculate the 200 pair texts entries and select ten entries of them shown in Table 1. As shown above, the distance value is smaller, the two texts are similarity. When the two texts are different from the content, the distance value will larger. So, we can see from the table, the distance value of text pair number 5 equal to 0, the content of the text are similarity. We analyzes the two texts by manual, and found the two texts are similarity. Another number 3, the distance value is larger to 0, so the two texts are different from each other in contend. This is fit to our manual analysis.

The threshold α is important criteria in our detection algorithm. The criteria will affect the parameters of our detection algorithm. So, it is important to select appropriate threshold α. Firstly, we select distance formula *Bhattacharyya* to show the threshold α effect the parameters of algorithm. The parameters of precision P, recall R and $F1$-Measure with different threshold α is shown in Figure 5. From this graphic, with the threshold α larger, the parameters P, R and $F1$ are close to 1. When the threshold α is 0.3, the three parameters equal to 1. But if the threshold α is becoming smaller, the parameters P, R and F1 reduce very fast. So, we choose the threshold $\alpha = 0.1$ to test the four distance formula. The experiment result is shown in Figure 6.

In order to get the best performance, we think about the threshold α and $F1$. The experiment result is list in Table 2. From this table, when the distance formula is *Bhattacharyya* and the threshold $\alpha = 0.1$, $F1 = 0.9$.

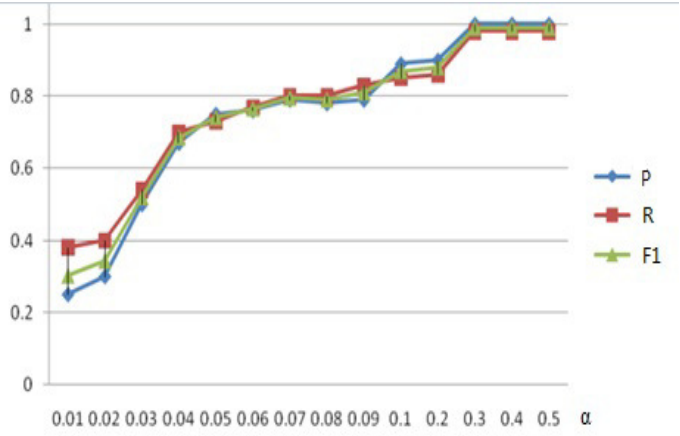

Figure 5: Threshold α

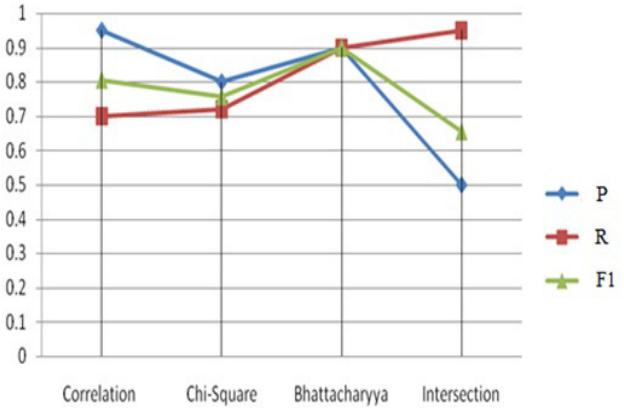

Figure 6: Distance formulas

Table 1: The distance of four formulas

Pairs of Numbers	Correlation	Chi-Square	Intersection	Bhattacharyya
1	0.045617	0.11053	0.16319	0.18374
2	0	0.00020945	0.006156	0.0072509
3	0.071436	0.14858	0.20078	0.22477
4	0.025078	0.076275	0.13112	0.15592
5	0	0	0	0
6	0.030073	0.096455	0.14462	0.17483
7	0.011809	0.03041	0.087298	0.092447
8	0.001564	0.0052661	0.03533	0.036437
9	0.014378	0.046964	0.0998042	0.11998
10	0.04337	0.15721	0.18893	0.22385

Table 2: Threshold α and F1

Distance formula	α	F1
Correlation	0.2	0.6
Chi-Square	0.15	50.8
Bhattacharyya	0.08	0.8
Intersection	0.1	0.9

After the parameter of our detection algorithm is found. We select *Cosine* algorithm and *Jaccard* method proposed in literature to prove ours are better than the two methods in many areas.The parameters P, R, $F1$ is shown in Figure 7. From this graphic, ours method is better than Cosine and *Jaccard*.

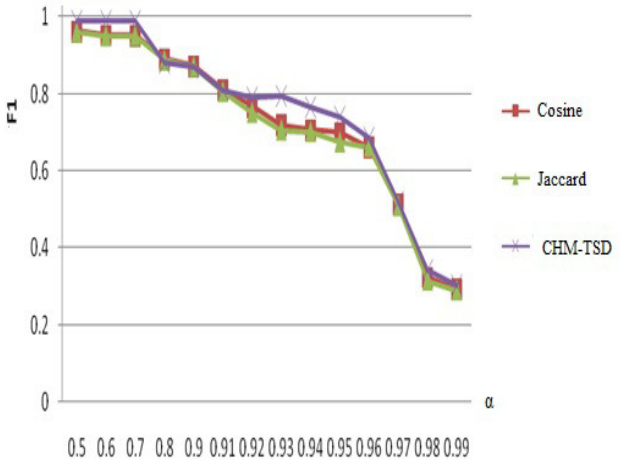

Figure 7: F1 to different methods

5 Conclusion

Text duplicate detection is mainly used for copy detection and webpage de-duplicate. It is also an effective ways for maintaining information quality. This paper put forward a new algorithm of text duplicate detection after the analysis and research on the structure of Chinese characters. CHM-TSD starts a new view of Chinese text similarity detection research. Chinese characters in text are split into components to build CHM. The texts similarity is obtained by computing the distance of all text CRM and duplicated text CHM. The experimental results show that CHM-TSD performs better than cosine theorem and *Jaccard* coefficient.

This paper provides a new idea of the natural language processing. The method can be used for text information processing and duplicated webpages deletion. In our future work, we will improve the efficiency of component decomposing and enhance the precision of the algorithm on the detection of the two texts that have a large variation on the number of words.

Acknowledgment

This study is supported by National Natural Science Foundation of China (No. 61304208, 61202496), Hunan Province Natural Science Foundation of China (No. 13JJ2031), and Youth Scientific Research Foundation of Central South University of Forestry & Technology (No. QJ2012009A).

References

[1] "Baidu," `http://www.baidu.com`, 2014. [Online; accessed 3-MARCH-2014].

[2] "Baike," `http://www.baike.com`, 2014. [Online; accessed 3-MARCH-2014].

[3] J. P. Bao, J. Y. Shen, and X. D. Liuand, Q. B. Song, "A survey on natural language text copy detection," *Journal of Software*, vol. 14, no. 10, pp. 1753–1760, 2003.

[4] C. L. Chen, F. S. C. Tseng, and T. Liang, "An integration of fuzzy association rules and wordnet for document clustering," *Journal of Knowledge and Information Systems*, vol. 28, no. 4, pp. 687–708, 2011.
[5] C. H. Huang, J. Yin, and F. Hou, "A text similarity measurement combining word semantic information with TF-IDF method," *Chinese Journal of Computers*, vol. 34, no. 5, pp. 856–864, 2011.
[6] X. G. Peng, S. M. Liu, and Y. Song, "A copy detection tool for chinese documents," in *Proceedings of the 2nd International Conference on Education Technology and Computer*, pp. 125–129, 2010.
[7] J. Shi, Y. F. Wu, L. K. Qiu, and X. Q. Niu, "Chinese lexical semantic similarity computing based on large-scale corpus," *Journal of Chinese Information Processing*, vol. 27, no. 1, pp. 1–6, 2013.
[8] C. N. Sun, C. Zhang, and Q. S. Xia, "Chinese text similarity computing based on lda," *Journal of Computer Technology and Development*, vol. 23, no. 1, pp. 217–220, 2013.
[9] X. M. Sun and J. P. Yin, "On mathematical expression of a Chinese character," *Journal of Computer Research and Development*, vol. 39, no. 5, pp. 707–711, 2004.
[10] M. V. Thada and M. S. Joshi, "A genetic algorithm approach for improving the average relevancy of retrieved documents using jaccard similarity coefficient," *International Journal of Research in IT Management*, vol. 11, no. 3, pp. 50–55, 2011.
[11] L. X. Wang, H. T. Gong, K. Sun, and X. Zhang, "Auto-detection technology of text divulgence based on natural language processing," *Computer Engineering and Design*, vol. 32, no. 8, pp. 2600–2603, 2011.
[12] Z. G. Wang and M. Wu, "Similarity checking algorithm in item bank based on vector space model," *Computer System Application*, vol. 19, no. 3, pp. 213–216, 2011.
[13] P. Y. Zhang, "A hownet-based semantic relatedness kernel for text classification," *TELKOMNIKA Indonesian Journal of Electrical Engineering*, vol. 11, no. 4, pp. 1909–1915, 2013.
[14] X. C. Zhang, W. Xu, and L. Gao, "Combining content and link analysis for local web community extraction," *Journal of Computer Research and Development*, vol. 49, no. 11, pp. 2352–2358, 2012.
[15] D. P. Zhao, L. J. Cai, and P. Li, "A similar text detection algorithm based on newshingling," *Journal of Shenyang Jianzhu University (Natural Science)*, vol. 27, no. 4, pp. 771–775, 2011.

Huajun Huang is currently a faculty member in the college of Computer and Information Engineering at Central South University of Forestry & Technology.His overall research area include of Webpage information hiding and hidden information detection, XML Watermarking, Anti-phishing, Mobile Device Forensics. Dr. Huang received his Ph.D. from Hunan University in 2007, M.S. degrees from Hunan University in Software Engineering (2004), and a B.A. in Applied Physics from Yunnan University (2001).

Shuang Pang is currently a postgraduate student in the college of Computer and Information Engeering at Central South University of Foresty &Technology.Ms Pang received her B.A.in Software Engeineer from Central South University of Foresty & Technology in 2014.

Qiong Deng is currently a postgraduate student in the college of Computer and Information Engeering at Central South University of Foresty &Technology.Ms Deng received her B.A.in Software Engeineer from Central South University of Foresty & Technology in 2014.

Jiaohua Qin received her BS in mathematics from Hunan University of Science and Technology, China, in 1996, MS in computer science and technology from National University of Defense Technology, China, in 2001, and PhD in computing science from Hunan University, China, in 2009. She a professor in College of Computer Science and Information Technology, Central South University of Forestry and Technology, China. Her research interests include network and information security, image processing and pattern recognition.