ISSN 2616-5775 Vol. 7, Issue 4: 72-79, DOI: 10.25236/AJCIS.2024.070410

A similar Chinese character detection algorithm that fuses Chinese character images and encoding

Chun Wang[1,a,*], Dejun Chen[1,b]

[1]*Wuhan University of Technology, Wuhan City, Hubei Province, China*
[a]*3351908022@qq.com,* [b]*1151959979@qq.com*
[*]*Corresponding author*

Abstract: *In the past, research on similar Chinese characters often focused on extracting the contours and structures of Chinese character images for comparison, or encoding the sound, form, and meaning of Chinese characters to measure their string similarity. However, similar characters found by such methods often have homophones but different structures. In this paper, we propose a fusion algorithm that combines Chinese character images with Chinese character encoding to fully extract the structural, stroke, stroke order, and contour features of Chinese characters. Based on these features, we can find a list of characters that are more similar to the original characters in terms of character structure and stroke order. The algorithm first selects specific pixel lattice to measure the similarity of Chinese character glyphs based on different Chinese character structures. Then, combined with the four corner stroke order encoding of Chinese characters, the structural stroke order similarity of Chinese characters is calculated through Jaro Winkler Distance. Finally, the fusion of the two results in the similarity of two Chinese character glyphs. The experimental results show that the similarity algorithm, which incorporates the advantages of image and encoding, can better obtain a list of similar characters with the same glyph structure and similar stroke order.*

Keywords: *Quadrangle encoding, Jaro Winkler Distance, glyph, similarity, Chinese character encoding*

1. Introduction

Chinese characters are composed of strokes, components, and whole characters. The basic unit of Chinese characters is stroke [1]. Stroke forms various components through arbitrary combinations, and components are part of the whole character. Different component combinations form different Chinese characters. There are only 32 types of Chinese character strokes, and random combinations can form similar components. When these similar components are combined, they can produce similar Chinese characters. Similar Chinese characters often encounter recognition errors during image recognition and text information processing. Although the technology of recognizing Chinese character images based on CNN networks is gradually maturing [2], the recognition of Chinese characters with the same shape in Chinese character images is not high. This is because there are subtle differences in these similar shapes, and the network can only extract the approximate outline of the shape. In the process of matching with the template, it cannot perceive internal differences. This is a hot topic in the field of recognition research. At the same time, similar Chinese characters also pose a great challenge for primary and secondary school students who are learning Chinese characters. The overlapping of two strokes between similar characters can easily lead to errors or confusion in writing. Therefore, studying the differences between similar characters can not only make existing recognition systems more accurate for handwritten Chinese characters, but also help students become more familiar with the characteristics of each character and write Chinese characters correctly. The definition of similar characters is that Chinese characters with the same glyph structure and similar strokes are considered similar characters. Therefore, this article combines the glyph outline of Chinese characters and their structural stroke order information to find a list of similar characters in Chinese characters from multiple dimensions.

For shape recognition, Chinese characters are replaced with a string of English letters in the early stages, and then two Chinese characters are converted into strings of different digits. The minimum number of editing operations required to convert one string into another is counted to measure the similarity between the two, that is, the Levenshtein Distance between the two [3] [4]. Diao Xingchun first encodes Chinese characters into English strings through their Pinyin and Wubi, and then measures the similarity between the two glyphs through Levenshtein Distance. Subsequently, Chen Ming et al. divided

ISSN 2616-5775 Vol. 7, Issue 4: 72-79, DOI: 10.25236/AJCIS.2024.070410

the Pinyin of Chinese characters into initials, finals, and consonants, and used four corner encoding to replace the joint mapping of Chinese character structures into strings. They used Levenshtein Distance to measure the similarity between the two. Recently, with the increasing richness of the Chinese semantic knowledge base [7] (HowNet) and the increasing similarity between word meanings, Wang Huamin et al. [8] combined Chinese phonetic codes with HowNet for Chinese word similarity detection. However, the Levenshtein Distance only reflects local similarity, often ignoring the overall impact on similarity. Therefore, foreign scholar LazReg M B [9] measured the similarity of encoded Chinese characters based on context, and the similarity results were significantly improved compared to measuring independent Chinese characters. However, the above measurement methods combined with phonetic codes indicate that due to the significant proportion of Chinese Pinyin coding, similar characters found may have homophones but different shapes, which is not ideal. The above is based on encoding Chinese characters, and some have also started from the aspect of Chinese character images. Zhao Jian et al. first extracted the structure of Chinese character strokes from handwritten and printed Chinese character images, and then found important parts of the structure, such as finding the corners and intersections of Chinese characters to obtain contour information. Recently, Liu Mengdi [11] combined a knowledge graph to divide Chinese characters into components and radicals, and constructed a set relationship between the two. The similarity of Chinese character shapes was calculated using the 2CTransE model. This will result in more similar Chinese characters with the same glyph structure, but it cannot guarantee having the same components.

Therefore, based on the advantages and disadvantages of the existing methods mentioned above, this paper combines image and Chinese character encoding to comprehensively extract information such as the outline of Chinese characters, the structure of Chinese characters, and the stroke order of strokes. Due to the fact that previous distance calculation methods such as Levenshtein Distance cannot measure the similarity degree based on the order and length of strings, the stroke order of Chinese characters, which is the order in which each stroke appears, is also a part of the similarity degree between the two Chinese characters. Therefore, this paper uses Jaro Winkler Distance [12] to calculate the similarity between the two strings, and the obtained similarity result ensures that the stroke order of Chinese character structure is the same in a larger range. The algorithm process in this article is as follows:

(1) Comparing the pixels of a specific font Chinese character image yields a feature matrix that represents the differences in the glyph contours between two Chinese characters. The feature matrix is divided by the sum of pixels to obtain the similarity of the glyph features between two Chinese characters.

(3) Merge the four corner encoding and stroke order encoding of each Chinese character into a string, and use the Jaro Winkler algorithm to find the similarity of stroke order between the two encoded Chinese character structures.

(4) The fusion similarity of two Chinese characters is obtained by weighting the similarity between the glyph features and the stroke order of the Chinese character structure.

2. Data

2.1 Data preparation

Prepare a font library of 20000 commonly used Chinese characters, each of which is replaced by a separate index in the font library. Use Python's pygame package to convert all the characters in the Chinese character library into Microsoft Yahei's Chinese character image database. Set the Chinese character font in the image to black, the image background to white, and the naming of each Chinese character image to be replaced by a Chinese character number.

Based on the structure, stroke order, and corner coding information of each Chinese character, create a Chinese character structure JSON table, stroke order JSON table, and corner coding JSON table. Replace 8 Chinese character structures with numerical values, where 1 represents the left and right structure, 2 represents the up and down structure, and 0 represents the remaining structures. The stroke order JSON table encodes the strokes of each character based on the writing order of the character. The encoding of each stroke is shown in the following figure:

For example, According to the relationship between strokes and characters shown in Figure 1. the Chinese character "tu" is encoded as "jfj". The four corners of the soil are encoded as "4010", so combining the two codes together forms an encoding string that preserves both the Chinese character shape structure and stroke features. The final code for character "tu" is "jfj4010". Encode all Chinese

ISSN 2616-5775 Vol. 7, Issue 4: 72-79, DOI: 10.25236/AJCIS.2024.070410

characters in the font library according to the above method and create a Chinese character encoding database.

a(㇋)	b(㇄)	C(㇕)	d(㇀)	o(㇈)	j(㇐)	l(㇏)	r(㇆)	f(㇑)
g(㇚)	k(㇔)	s(㇓)	n(㇜)	x(㇉)	w(㇌)	z(㇉)	i(㇀)	t(㇓)
y(㇟)	v(㇅)	e(㇇)	p(㇊)	q(㇍)	h(㇙)	m(㇛)	u(㇗)	

Figure 1: Character codes for each stroke, replaced by 32 different English letters based on the 32 strokes in Chinese.

2.2 Data processing

Perform image processing on the Microsoft Yahei Chinese character image for each Chinese character, including cropping, interpolation, and grayscale processing. Cropping and interpolation resolution are achieved using the resize function in Python v2. The image is cropped to 12 pixels in size and width, followed by bicubic interpolation to complete the pixel values of the image. Finally, the image is converted into a grayscale image. The following is a comparison of the first word in the font library, the original image, and the processed image.

The original Microsoft Yahei Chinese character image is reduced in size after processing, and only the important feature pixels of the Chinese character in the image are retained. At this time, the pixel values in the grayscale image are only 1 and 0.

3. Similarity calculation model

At the end of this article, a list of similar characters with the same character structure and similar stroke order is obtained. Therefore, before calculation, the structure of the two Chinese characters is also judged. Only Chinese characters with the same structure can be used for subsequent calculations. For Chinese characters with the same structure, first measure the similarity of image features between two Chinese characters, then encode the stroke order and four corners of the Chinese characters, calculate the encoded similarity value using the Jaro Winkler algorithm, and finally calculate the weighted sum of the two similarities to obtain the comprehensive similarity size of the two Chinese characters. Determine whether the two Chinese characters are similar based on the set threshold.

3.1 Calculation of similarity in Chinese character image features

Chinese character images often contain the glyph features of the character. After cropping and grayscale conversion, a binary image containing only Chinese character features can be obtained. The main feature is reflected in the distribution of pixels in the binary image. Different structures of Chinese characters represent different numerical values of the glyph in the pixels of the image

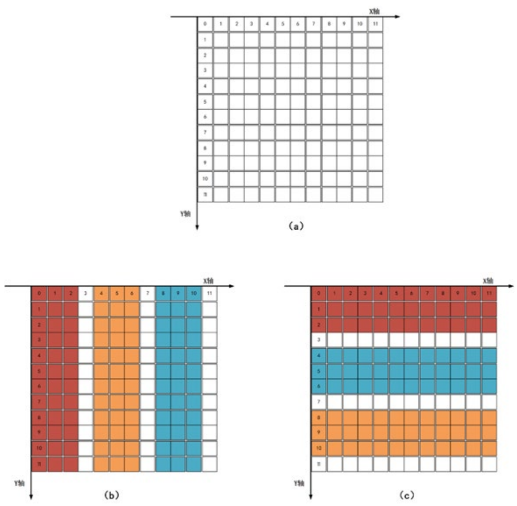

Figure 2: Pixel Calculation Network for Chinese Characters with Different Structures.

ISSN 2616-5775 Vol. 7, Issue 4: 72-79, DOI: 10.25236/AJCIS.2024.070410

Figure 2 shows the pixel lattice comparison area of three binary image structures, namely the pixel computing network. Each pixel lattice contains a total of 12 x 12 pixels. Binary Chinese character images have specific values in specific areas within the pixel lattice, and each lattice has values of 0 and 1. The method of comparing the feature similarity between two Chinese character images is to compare the numerical values in the same area. If the values are the same, return 1. If the values are different, return 0. The returned results are stored in the feature vector. Figure (a) shows a Chinese character pixel calculation network with structure 0, comparing the pixel values of 144 full pixel positions and returning a vector with a length of 144. Figure (b) shows the division of comparison regions in the pixel matrix of Structure 1. Only regions with x [1,2], [4,5,6], [8,9,10] are compared, and feature vectors with a length of 108 are returned. Figure (c) shows the pixel lattice comparison region of Structure 2, only comparing regions with y values of [1,2], [4,5,6], [8,9,10], and finally returning a feature vector of length 108. The final feature vector is summed and divided by its total length to obtain the similarity between two Chinese character images.

3.2 Calculation of stroke order similarity for four corner strokes

This section calculates the similarity of Chinese characters after encoding. The encoding method is four corner encoding plus the stroke order encoding of Chinese characters. The final encoding format here is stroke order encoding plus four corner encoding. The encoded string is calculated for similarity through Jaro Winkler.

The Jaro Winkler algorithm is an improvement of Winkler on Jaro distance. It can comprehensively measure the similarity of strings based on the length and order of characters. Therefore, based on the above characteristics, it can better measure the similarity of similar Chinese characters with the same stroke and little difference in stroke order. Jaro Winkler consists of two parts. The first part is the Jaro distance, and the second step is the Winkler distance. The expression for the Jaro algorithm is as follows:

$$d_j = \begin{cases} 0, m = 0 \\ \frac{1}{3}\left(\frac{m}{|s_1|} + \frac{m}{|s_2|} + \frac{m-t}{|m|}\right), m > 0 \end{cases} \tag{1}$$

In equation 1, s1, and s2 are the lengths of two strings, m is the number of identical characters in both strings, and t is the number of permutations. When calculating the distance between two strings, Jaro distance determines the maximum distance range for matching characters through a window function, with the matching window being:

$$MW = \left\lfloor \frac{\max\left(|s_1|, |s_2|\right)}{2} \right\rfloor - 1 \tag{2}$$

If there are the same characters in two strings, and the positions of the same characters in these two strings are different and within the range of MW, the number of identical characters with such characteristics is counted as the transposition operator. Therefore, the transposition number t can be obtained, and the formula is as follows:

$$t = \frac{t_j}{2} \tag{3}$$

By using MW to find the matching number of identical characters m and the number of substitutions t, the Jaro similarity values of two strings can be obtained by substituting them into the Jaro Winkler formula as follows:

$$d_w = d_j + \left(lp\left(1 - d_j\right)\right) \tag{4}$$

In equation (4), the range of p is generally set between 0 and 0.25, and in this article, the fixed value of p is 0.1; L is the common prefix length, which is the same number of characters from left to right in two strings.

ISSN 2616-5775 Vol. 7, Issue 4: 72-79, DOI: 10.25236/AJCIS.2024.070410

3.3 Algorithm process

This article combines the above two similarity calculations to find similar characters for the Chinese character. The algorithm flowchart is as follows:

1) Input a Chinese character;

2) Randomly extract a Chinese character from the font library;

3) Randomly select a Chinese character from the font library and compare their structures. If the structures are the same, proceed to step 3. Otherwise, return to step 2

4) Obtain two Chinese character images and encode them.

5) According to the structural encoding (0,1,2) of two Chinese characters, match the corresponding pixel lattice to calculate the grid, and the similarity between the two glyph contours is denoted as.

6) Calculate the structural stroke similarity between two Chinese characters using the Jaro Winkler Distance algorithm.

7) Calculate the fusion similarity using the following formula

$$Sim(x,y) = Sim1(x,y)\times\alpha + Sim2(x,y)\times\beta \tag{5}$$

In equation (5), x and y represent two Chinese characters, where $Sim1(x,y)$ is the similarity of the image feature glyphs of the two Chinese characters, $Sim2(x,y)$ is the similarity of the stroke order encoding of the four corners of the two Chinese characters, and α,β are the weighting factors of the two similarities, satisfying $\alpha+\beta=1$. Here, α is set to 0.45 and β is set to 0.55.

8) Based on the final weighted score, determine whether it is greater than the threshold. If it is greater than the threshold, place the selected characters from the font library into the initial similar Chinese character result list, where the threshold is set to 0.85.

9) Traverse the Chinese character strokes in the initial result list and calculate the error with the input Chinese character strokes. If the stroke difference is within 1 stroke, it is placed in the final similar character result list. Otherwise, it is discarded.

4. Experimental Results and Analysis

4.1 Chinese character image similarity test

This article divides Chinese characters into three types of structures: left and right structure, up and down structure, and other structures, which are replaced by 1, 2, and 0 respectively. 20000 Chinese characters are classified into different structures, and image similarity is calculated for similar Chinese characters. The different structures are shown in the table below:

Left and right structures	Upper and lower structures	Other structures
61%~95%	63%~93%	43%~96%

From the table, it can be seen that there is not much difference in the similarity between the first two types of images, because for the pixel lattice grid division of the left and right structures and the upper and lower structures, the feature information of the Chinese character images in their left and right and upper and lower parts can be well matched, and the divided network area can match the structural characteristics of the Chinese characters well. For example, dividing the pixel computing network of categories 1 and 2 into another form. A computing network with a structure of 1, dividing the x-axis into [1,2], [4,5], [7,8], [10,11]; The structure of the network is 2, and the y-axis is divided into [1,2], [4,5], [7,8], [10,11]. The total number of calculated lattice points is 96, while the rest of the structures remain unchanged. The following similarity results are obtained:

Left and right structures	Upper and lower structures	Other structures
60%~93%	62%~92%	43%~96%

From this, it can be seen that the location and range size of the calculation network partition will affect the calculation results of Chinese character image similarity. The reason for the large difference in the similarity results of other structural images is that the Chinese characters in other structures include

ISSN 2616-5775 Vol. 7, Issue 4: 72-79, DOI: 10.25236/AJCIS.2024.070410

a large number of Chinese characters such as upper, middle, and lower structures, left, middle, and right structures, full enclosing structures, half enclosing structures, interspersed structures, and pin shaped structures. Different structures of Chinese characters have different pixel values distributed in 144 pixel grids. For characters with significant structural differences, such as "full enclosing structures" and "upper, middle, and lower structures", "full enclosing structures" and "left, middle, and right structures", their Chinese character components have different distribution areas throughout the entire Chinese character, resulting in significant differences in pixel values in the grid. However, for Chinese characters with the same or similar structures, such as "fully enclosed" or "semi enclosed" structures, the calculation results tend to be higher.

4.2 Four corner stroke order similarity experiment

The four corner encoding represents the glyph structure of Chinese characters, and its encoding order is sequentially from the top left corner, top right corner, bottom left corner, and bottom right corner. If two Chinese characters have the same components, such as those with the same radical, their four corner codes will be partially the same, whose four corner codes are "2420" and "2421", respectively. After being measured by the Jaro Winkler algorithm, this type of Chinese character has a high similarity. If the components of two Chinese characters are almost identical and have the same stroke order prepared, respectively, the similarity reaches 0.9636.

Therefore, for the above analysis, the compared Chinese characters are divided into the following categories under the same structure of two Chinese characters: (A) different components, (B) a small number of identical components with varying stroke counts, (C) the same components with similar stroke counts and significant differences in stroke order, (D) the components are basically the same, with little difference in stroke counts and stroke order, (E) the components are the same, with the same stroke counts and stroke order. The experimental results of these categories are shown in the table below:

A	B	C	D	E
30%~60%	53%~77%	71%~87%	77%~91%	greaterthan94%

The 5 types of comparisons designed reflect the changes in similarity between two Chinese characters from uncorrelated to correlated processes. From the table, it can be seen that the stroke order of the components of the two Chinese characters is gradually unifying, and their similarity measurement intervals are constantly increasing. The upper and lower limits are constantly adjusting upwards, indicating that the components, strokes, and stroke order of the two Chinese characters will affect their similarity. A component is composed of one or more strokes, with multiple strokes forming a single component in a certain order. Therefore, having the same component means having the same stroke order coding, which enhances the similarity of Chinese characters. The comprehensive encoding will be calculated using the Jaro Winkler algorithm, and its prefix coefficient will affect the similarity size of the strings. If the prefix coefficient is the same, and the number of identical letters from left to right in two strings is the same, the higher the similarity. Therefore, for Chinese characters with the same component and stroke order, the final calculation result will be extremely high, almost always above 94%.

4.3 Fusion similarity experiment

The fusion similarity experiment outputs an initial list of similar Chinese characters with the same structure and a similarity greater than 0.85. The following table lists the search results for a small number of initial similar Chinese characters.

Character	similarity number
Tu	4
Jin	4
Wei	5
Feng	3
Yin	5
Bai	4

The above table shows the initial similarity list of several different results of Chinese characters.

The left side is the input Chinese character, and the right side is the number of similar characters found by the input Chinese character. It can be seen that in the initial similarity list found by fusing similarity, all similar characters found have the same components. After experiments, it was found that reducing the length of the common prefix in Jaro Winkler would affect the overall similarity value, which

ISSN 2616-5775 Vol. 7, Issue 4: 72-79, DOI: 10.25236/AJCIS.2024.070410

in turn would affect the number of similar Chinese character lists found. This effect has a significant impact on characters with more input strokes and left, right, and up and down structures. For example, when the current prefix length is set to 2, it will reduce the score of fusion similarity and thus reduce the number of familiar Chinese characters found. But when the prefix length is set to 5, But when the prefix length is set to 5, it can match more identical characters. For two Chinese characters with the same radical and multiple strokes, if their radical is around 5 and the rest of the components of the two characters are not significantly different, their similarity is very high.However, similar characters with significantly different strokes will appear in the initial list, and these characters will be filtered out through subsequent stroke determination rules to find the final similar characters with the same structure, components, and stroke order; Or similar characters with the same structure, components, and slight differences in stroke order.

After the final stroke threshold judgment, the final list of similar characters for the Chinese characters listed above is as follows:

Character	similarity number
Tu	3
Jin	2
Wei	3
Feng	1
Yin	2
Bai	3

It can be seen that for each input Chinese character, after the final stroke error judgment, the output of similar characters will be fewer, and the component structure of the output similar characters is almost the same as the original input characters, except that some characters may have a stroke error of one.

The similar characters obtained above have a good effect, and then the algorithm in this article is used to traverse 20000 character libraries to find a similar list for each character library. Finally, a similar character library with similar strokes will be obtained. Compared with other algorithms, it eliminates the influence of homophonic characters, such as the similarity search algorithm based on sound, form, and meaning encoding. This algorithm will find many homophones with different glyph structures, and according to the definition in the previous text, characters with different structures do not belong to similar characters

This experiment first calculates the similarity of words in each type of structure, and extracts similar words with a calculation result exceeding 90% from each structure for statistics. After statistics, there are 34 pairs with a similarity exceeding 0.99, 15 pairs with a similarity exceeding 0.98, 300 pairs with a similarity exceeding 0.96, and 6230 pairs with a similarity exceeding 0.9.

5. Conclusion

Starting from the structure and stroke order of Chinese characters, this article proposes a hybrid algorithm based on Chinese character image features and four corner stroke order encoding, which can effectively remove other methods of finding similar characters with homophones but different structures. The Chinese characters found in this algorithm not only have the same structure but also almost the same stroke order, achieving true visual similarity. Through two decomposition experiments and a comprehensive experiment, the following conclusions can be drawn: (1) The binary Chinese character image can reflect the font structure and contour information of a Chinese character, and comparing the pixel distribution of the binary images of two Chinese characters can extract the difference information in structure and contour between the two; (2) Different Chinese character images are divided into different pixel calculation networks based on their structural characteristics, which can extract key information from characters with different structures; (3) The four corner stroke order coding defines the similarity between two Chinese characters in terms of writing degree. From the similarity results after coding, two Chinese characters with the same stroke order coding also have similar four corner codes. There are only slight differences in writing, and the similarity between the two characters is extremely high. Therefore, this article combines two aspects of Chinese character images and strokes to measure the similarity of Chinese characters. Compared to using only one method before, it has made a greater improvement in image or Chinese character character encoding, and the search effect is better.

ISSN 2616-5775 Vol. 7, Issue 4: 72-79, DOI: 10.25236/AJCIS.2024.070410

References

[1] Li Zeyao, Li Chengcheng. Handwritten Chinese Character Component Extraction Algorithm Based on Structural Knowledge [J]. Computer Engineering and Design, 2023, 44 (05): 1479-1486.
[2] Xu Qi. Handwritten Chinese Character Recognition Based on Convolutional Neural Networks [J]. Electronic Technology and Software Engineering. 2022 (09): 190-193.
[3] Zhao Zhijing, Jiang Di. Research on Language Classification Based on Editing Distance [J]. Language Research. 2020, 40 (02): 43-50.
[4] Zhang Shengnan. Research on String Similarity Algorithm Based on Editing Distance [J]. Computer Application Research [J]. 2023, 29 (14): 23-26.
[5] Diao Xingchun, Tan Mingchao, Cao Jianjun, et al. A String Similarity Calculation Method Integrating Multiple Editing Distances [J]. Computer Applications and Research. 2010, 27 (12): 4523-4525.
[6] Chen Ming, Du Qingzhi, Shao Yubin, Long Hua et al. A Chinese character similarity comparison algorithm based on phonetic codes [J]. Information Technology, 2018 (11): 73-75.
[7] Yunnian Din, Yangli Jia, and Zhenling Zhang. A conceptual similarity and correlation discrimination method based on HowNet[C]. MATEC Web of Conferences 309, 03020 (2020).
[8] Wang Huamin, Huang Mengxing, Feng Wenlong, Feng Siling. Chinese word similarity detection algorithm based on improved phonetic code HowNet [J]. Computer simulation. 2022, 39 (08): 460-465.
[9] Lazreg M B, Goodwin M, Granmo O C. Combining a context aware neural network with a denoising autoencoder for measuring string similarities [J]. Computer Speech & Language, 2022, 60.
[10] Zhao Jian, Feng Qiaosheng, He Juanjuan. New Features and Extraction Methods for Chinese Character Recognition [J]. Software, 2015, 36 (03): 31-36.
[11] Liu Mengdi, Liang Xun. A similarity calculation method for Chinese character shapes based on radical knowledge representation learning [J]. Chinese Journal of Information Science. 2021, 35 (12): 47-59.
[12] Zhang Huanqing, Zhang Man, Feng Ning, etc. Research on Standardization Technology of Pressure Plate Names Based on Jaro Winkler Distance Algorithm and Improved Processing Flow [J].Electrical Technology. 2019, (14): 69-73