Image Clustering of Immune Genetic Algorithm Based on Simulated Annealing

Fan Kaixiang\textsuperscript{a}, Yang Shuying\textsuperscript{b}, and Zhang Zeyi\textsuperscript{c}

Tianjin University of Technology, 391 Binshui West Road, Xiqing District, Tianjin, China
\textsuperscript{a}779604543@qq.com, \textsuperscript{b}1427956631@qq.com, \textsuperscript{c}565792963@qq.com

Keywords: immune genetic algorithm, cluster analysis, simulated annealing

Abstract: This paper proposes an immune genetic algorithm based on simulated annealing for image clustering. Grid template is used to extract the image features of cluster samples, and the solution of the problem is coded. In order to approach the optimal solution, calculate the distance between different antibodies, and construct the vaccine table according to the shortest principle. First, the crossover operator is performed on the population. According to the simulated annealing principle, the variation operator is used to search in the local area. At the same time, in order to accelerate the search speed of the optimized solution, the operation of vaccination operators is carried out. If the new antibody is superior to the old one, it receives; otherwise, it will be received by the Metropolis criterion. The antibody concentration was calculated and the immune balance operator was used to suppress the high concentration of antibodies. According to the fitness and concentration of the antibody, the selection probability was determined. The operation of immune selection operator was carried out, and the combination became a new generation population. The simulation results show that the algorithm can improve the accuracy and efficiency of image clustering.

1. Introduction

Image clustering is an algorithm to classify several samples into different types and classify objects of the same shape into one class. To make the individual in the same class as similar as possible, while the individual in the same class has a greater difference. Traditional image clustering algorithms such as genetic algorithm immune genetic algorithm, cloning algorithm, simulated annealing algorithm and so on \cite{1-2}. The simulated annealing algorithm is a general optimization algorithm. The simulated annealing algorithm used in Reference \cite{3-4} has carried out multiple mutation operations. At the same time, it destroys the diversity of individuals and ignores the relationship between individual learning and population evolution. The genetic algorithm proposed by Reference \cite{5-6} adopts the search strategy of swarm intelligence. The algorithm can show poor performance, the search speed is relatively slow, prone to precocity and other phenomena. Immune genetic algorithm is an algorithm that introduces immune mechanism in the whole frame of genetic algorithm. Reference \cite{7-8} proposed the immune genetic algorithm. This algorithm increases the operation of vaccination operators. However, this algorithm lacks search depth and antibody diversity.
This paper proposes an immune genetic algorithm based on simulated annealing. This algorithm adopts the group optimization strategy, synthesizes the simulated annealing algorithm, traditional genetic algorithm and immune genetic algorithm. At the same time, it avoids the problem of single algorithm optimization. The algorithm combines the genetic algorithm to automatically determine the clustering number and the ability to obtain the optimal clustering scheme. The calculation of the algorithm is simplified. It has strong searching ability and keeps the population pattern diversified. Efficient and accurate image clustering is achieved.

2. Basic Principles of Immune Genetic Algorithm Based on Simulated Annealing

The immune genetic algorithm based on simulated annealing is a new algorithm which synthesizes the advantages of many algorithms. The implementation steps of clustering algorithm based on immune genetic selection algorithm are presented.

(1) Obtain the number and characteristic of all samples.

(2) Set the initial parameters and randomly generate the initial antibody population. The fitness value and antibody concentration of antibodies in the population were calculated.

(3) Calculate the distance between different antibodies. Build a vaccine table based on the shortest distance principle.

(4) Crossover operator. The antibodies in the antibody population were randomly paired. A cross-gene position is generated by the probability Pc in the antibody gene. Cross-gene genes are intersected until all antibodies in the intermediate population are selected.

(5) Deep search based on simulated annealing. The initial temperature $T_0$, the termination temperature $T$, Boltzmann constant $K$, the number of interactions $n$ at each temperature, and the cooling rate $r$.

1) Mutation operator. All the antibodies in the population cycle each gene bit, producing a random number. The mutation operation is performed on this gene when the random number is less than the set probability $P_d$. Randomly generate a number assigned to this position within the cluster and generate a subgroup.

2) Vaccination operators. For the antibodies in the population, the genes of the antibody were modified by using the similarity of the samples shown in the vaccine table.

3) Recalculate the fitness value of the antibody.

4) Perform simulated annealing algorithm:

When the temperature does not reach the cooling time (that is, $T$ is greater than or equal to $T$),

① Perform the following procedures:

a) On the basis of the initial solution $S$, the new solution $S'$ is generated by random method;

b) If $S'$ is a feasible solution, then the improvement degree of the current solution is calculated, namely, $\delta = f(S) - f(S)$;

c) If $\delta$ is greater than or equal to 0, set $S = S'$, and set the optimal solution to $S^* = S$. Otherwise, set $S = S$ with probability $P_m$;

② Reduce the temperature, that is, $T = r \cdot T$.

(6) Immune detection operator.

The fitness of the antibody was calculated, and the new antibody was retained if the fitness was improved.

(7) Immune balance operator. The antibody concentration is calculated and the selection probability is determined according to the fitness and concentration of the antibody: $P = \alpha \cdot P_f + (1 - \alpha) \cdot P_i$, to calculate the probability of antibody concentration. Among them, $P_f$ is the fitness probability of the antibody. It is defined as the ratio of the fitness and fitness of
the antibody. $P_a$ is the concentration probability of the antibody. The higher the antibody concentration, the more inhibited. The lower the concentration, the better; $\alpha$ is the proportional coefficient. It determines the effect size of fitness and concentration.

(8) Immune selection operator. The selection probability of each antibody was determined according to fitness value and concentration. Based on the selection probability of roulette, the antibody was selected from the antibody population and the combination became a new generation.

(9) Determine whether the termination conditions are met. If not satisfied, slow down the temperature, reset the number of iterations, and go to step (4). Satisfy the end iteration, and output the optimal solution.

3. Based on Simulated Annealing Immune Genetic Algorithm Bionic Computation Cluster Analysis

3.1. Feature Extraction

In this study, the extraction method of characteristic parameters is: a rectangular box is drawn on the outer edge of each graph, and the length and width of the rectangular box are divided into $N$ by $M$. Each graph is called $N \times M$ template. Then calculate the proportion of the black pixels of each small square in this box as the sample eigenvalues.

3.2. Structural Individuals

The symbol code is used to represent the class number of the sample. The serial number of the gene bit represents the number of the sample, and the sequence number of the gene bit is fixed. The category of each sample is changing at any time.

3.3. Calculate Fitness

The calculation process is similar to the fitness value calculation method in genetic algorithm. The calculation formula is shown in Equation (1).

\[
\text{AntiBody}(i).\text{fitness} = \sum_{i=1}^{\text{centerNum}} \sum_{j=1}^{n_i} \left| X_j^{(i)} - C_i \right|^2 = \sum_{i=1}^{\text{centerNum}} D_i
\]  

Among them, centerNum is the total number of clustering categories and the total number of samples belonging to class $i$. $X_j^{(i)}$ is the eigenvalue of the $J$TH sample belonging to class $i$. $C_i$ is the $i$th class center. Its calculation formula is shown in Equation (2).

\[
C_i = \frac{1}{n_i} \sum_{k=1}^{n_i} X_k^{(i)}
\]

The larger $\text{AntiBody}(i).\text{fitness}$ is, the smaller the error of this classification method is, namely the greater the fitness value.

3.4. Crossover Operator

Pairs of individuals in a population at random. Each pair of genes, based on the cross-probability $P_c$, decided to cross-operate the gene bits and then swap them for genetic values. This produces new
3.5. Mutation Operator

For each individual, each gene position is rotated to determine whether the mutation is performed according to the mutation probability. The variation operation is a value of a random feasible solution. Replace the current value to produce a variation effect.

3.6. Vaccination Operators

Vaccination operators are used first to extract the vaccine. The way to extract the vaccine is to build a vaccine table based on the characteristic values of the sample. It then vaccinates the antibody against individuals based on information in the vaccine table, meaning changes in gene values.

3.7. Immune Detection Operator

Compare the fitness value of antibodies before vaccination and after vaccination. If the fitness value of the antibody is increased after vaccination, replace the old antibody with the newly produced antibody.

3.8. Simulated Annealing Algorithm Detection

The vaccinated antibodies are used as the initial feasible solution. The simulated annealing algorithm is simulated by default parameters. These parameters include the initial temperature $T_0$, the termination temperature $t$, the number of interactions at each temperature $n$, the cooling rate $r$, and the constant $K$.

The algorithm was performed $n$ times when the temperature was not cooled.

3.9. Immune Balance Operator

1) Concentration calculation. For each antibody, the number of fitness values in the population and the number of antibodies that is similar is $N_i$. The concentration of $\text{AntiBody}(i).\text{density}$ is $\frac{N_i}{\text{AntiBodyNum}}$.

2) Calculation of concentration probability. Set a concentration threshold $T$. The statistical concentration is higher than the threshold antibody, and the number is $\text{HighNum}$. The probability of high concentration of these antibodies is shown in Equation (3).

$$P_{\text{high}} = \frac{1}{\text{AntiBodyNum}} \left( 1 - \frac{\text{HighNum}}{\text{AntiBodyNum}} \right)$$  \hspace{1cm} (3)$$

The probability of antibody concentration of the remaining $\text{AntiBodyNum-HighNum}$ is shown in Equation (4).

$$P_{\text{lowest}} = \frac{1}{\text{AntiBodyNum}} \left( 1 + \frac{\text{HighNum}}{\text{AntiBodyNum}} - \frac{\text{HighNum}}{\text{AntiBodyNum} - \text{HighNum}} \right)$$  \hspace{1cm} (4)$$
3.10. Immune Selection Operator

Based on the fitness value and antibody concentration, the selection probability was calculated by Equation (5).

\[
\text{AntiBody.P}_{\text{choose}} = \alpha \cdot P_{\text{fitness}} + (1 - \alpha) \cdot P_{\text{fitness}}
\] (5)

Choose the roulette wheel. Based on the selection probability, the antibody was selected and the relative fitness antibody was selected to form the next generation.

3.11. Termination Conditions.

After several iterations, the algorithm gradually converges to the maximum number of iterations. Or the optimal solution does not change during the iteration, that is, the global optimal solution has been found and the iteration terminates.

4. Experimental Results and Analysis

4.1. Algorithm Analysis and Comparison

In order to verify the clustering validity of the simulated annealing immune genetic algorithm proposed in this paper, a set of Lorenz equation is used to generate the data set for detection. \(a = 34, b = 8/3, c = 10\). The initial sequence value is \([-0,1]\). The sampling interval is 0.01. The Lorenz model is solved by the fourth-order runge-kutta algorithm. Generate 979 data points as simulation data. The embedding dimension is 10. Delay time \(\tau = 3\). Generate phase space points for clustering operations. The data are divided into eight categories. The data sets of Lorenz model are clustered by genetic algorithm (GA), immune genetic algorithm (IGA), simulated annealing algorithm (SAA) and immune genetic algorithm based on simulated annealing (IGASA). The clustering results of four clustering algorithms are compared and analyzed.

The parameters of immune genetic algorithm based on simulated annealing are set as follows. Population size \(G = 100\). Evolutionary algebra \(TG = 120\). Cross probability \(Pc = 0.6\). Mutation probability \(Pd = 0.3\). The initial temperature is \(T = 10\). The termination temperature \(t = 3\). The number of interactions at each temperature \(n = 3\). Constant \(K = 1\). Cooling parameter \(\alpha = 0.6\). Reference probability \(Pm = 0.3\). The clustering effect is shown in Figure 1. The parameters of the genetic algorithm and immune genetic algorithm are the same as those of the above algorithms. The parameters of simulated annealing algorithm are the same as the above algorithms. The maximum number of...
iterations for each clustering algorithm is 100. Run each algorithm 20 times. The average recognition accuracy of the clustering results obtained by recording 4 methods and the average iteration times of finding the optimal solution are shown in Table 1.

Table 1 Comparison of experimental results of 4 algorithms.

<table>
<thead>
<tr>
<th>Algorithm contrast</th>
<th>IGP(%)</th>
<th>Find the optimal average iteration number</th>
</tr>
</thead>
<tbody>
<tr>
<td>GA</td>
<td>78.54</td>
<td>242</td>
</tr>
<tr>
<td>IGA</td>
<td>80.62</td>
<td>114</td>
</tr>
<tr>
<td>SAA</td>
<td>69.68</td>
<td>321</td>
</tr>
<tr>
<td>IGASA</td>
<td>92.82</td>
<td>66</td>
</tr>
</tbody>
</table>

According to the data in Table 1, the average recognition accuracy of immune genetic algorithm based on simulated annealing is significantly higher than that of the other three algorithms. Moreover, the average iteration frequency of optimal solution was found to be the least. It is proved that the accuracy of the immune genetic algorithm based on simulated annealing is the best.

4.2. Image Clustering Analysis

In the MATLAB 2010 environment, programming has realized the clustering problem based on the graph of evolutionary programming. The corresponding clustering experiments were conducted on the computer of Inter(R) Core(TM) i5 CPU.

Figure 2 The number of the sample to be tested.

The method of feature extraction in the experiment is to divide the sample into 7 x 8 small squares. The proportion of the square area of black pixels in each square is called the eigenvalue. The rectangle is divided into 56 squares. The graphic experiment is numbered for cluster samples. There are ten samples, and the sample number is shown in Figure 2. In the upper right corner of each sample, different sample Numbers are different, and the number is always fixed.

Each individual contains a classification scheme. The antibody encoding of the initial individual is : (1,1,2,1,2,1,2,3,1,3). Because it is a random initialization, it is still assumed that the classification is not optimal.

Pairs of individuals in a population at random and cross operator operation. For example, the parent individual A is paired with the parent individual B. The random probability of some genes
was compared with $P_c$. The gene that satisfies the exchange probability is exchanged.

Subsequent mutations are performed. For each individual, each gene position is rotated to determine whether the mutation is performed according to the mutation probability. The variation operation is to take a value of the feasible solution at random and replace the current value to produce the effect of variation.

Vaccination operators are adopted for the modified antibodies. The distance between each antibody and other antibodies is calculated and binarized. For each individual, select one (or several) genes at random. By looking up the table, the value of the gene position nearest to the gene is changed to the corresponding number of genes. Thus, the antibody is improved by prior knowledge.

The antibody population after immunization was used as the initial solution $S$. The initial temperature $T=10$, $T=3$, $n=3$, constant $K=1$, and cooling parameter $a=0.6$ for simulated annealing algorithm detection. Finally, the concentration is selected to deal with the population. The immune balance operator and immune selection operator are used to select the antibodies in the population to maintain the diversity of the antibodies in the population. The population is iterated until the global optimal solution is obtained. The experimental results are shown in Figure 3. It can be seen from Figure 3 that the immune genetic algorithm based on simulated annealing can achieve the clustering of different graphs well and achieve better results.

![Figure 3 The optimal solution found based on simulated annealing immune genetic algorithm.](image)

5. Conclusion

In this paper, the bionic computation mechanism of immune genetic algorithm based on simulated annealing is studied. This algorithm introduces the simulated annealing algorithm and the immune principle into the genetic clustering algorithm. Grid template is used to extract the image features of cluster samples, and the solution of the problem is coded. In order to approach the optimal solution, calculate the distance between different antibodies, and construct the vaccine table according to the shortest principle. The diversity of population is maintained, and the immature convergence phenomenon is overcome effectively, and the performance of genetic algorithm is improved. The algorithm has good performance and clustering effect, which lays a foundation for further research on evolutionary computation and its application in other fields.

References


