Chinese Text Classification Based on Particle Swarm Optimization

LUO Xin [1, a]

[1]College of Business Administration

South China University of Technology, SCUT

Guangzhou, China

[a]Email:luoxin@scut.edu.cn

Keywords: Text processing; Classification; Artificial Intelligence; Particle swarm optimization (PSO)

Abstract. For mass, heterogeneous and dynamic text information, it has important significance to the text classification. In recent years, the theory and method of Swarm Intelligence has been developed gradually, which provides a new method for text classification. In this paper, the intelligent algorithm of swarm intelligence, Particle Swarm Optimization (PSO), is introduced into the field of text classification. A text classification model Text PSO-Miner based on PSO is constructed and tested on the Chinese text set. The results show that Text PSO-Miner can be well applied to Chinese text classification.

Introduction

With the development of Internet, the number of various electronic documents is increasing, the demand for automatic classification and management of documents becomes urgent. Text classification is one of the key techniques to manage the vast amount of information, which can solve the problem of "information explosion".

Swarm Optimization Particle (PSO) is an effective global optimization algorithm, which was first proposed by American social psychologist Kennedy and electrical engineer Eberhart in 1995[1]. PSO preserves the global search strategy based on population, using velocity displacement model, avoiding the complex genetic operation. At the same time, because of its memory, so that it can track the current search situation and dynamically adjust the search strategy, with strong global convergence ability and robustness, has become a research hotspot in the field of artificial intelligence.

A number of scholars have proposed some improved methods in the application of PSO algorithm to different fields[2][3]. Although PSO has achieved a series of good results in solving the above objectives, data clustering, combination optimization and network routing, the research on classification rule mining is very limited. Rule mining based on particle swarm algorithm was originally proposed by Sousa in 2003[4], but the algorithm is based on the binary encoding, the application is less. Holden Nicholas proposed the improvement in 2008[5], making the algorithm to deal with continuous data in the real number system encoding. PSO is simple, efficient and easy to implement. It has been applied in many fields, such as multi-objective optimization, constrained optimization and trajectory recognition. It provides a new intelligent method for text classification. Unfortunately, there is very limited research in this field.

PSO Text Classification Model

The Basic Theory of PSO.

The basic theory of particle swarm algorithm is derived from such a scenario. A flock of birds in only a piece of food area random foraging, all the birds do not know where to find food, but they know how far the current position is from the food. Then the optimal strategy for finding food is to search for the nearest area of the food. PSO is obtained from the model and used to solve the classification problem.

In PSO, the solution of each classification problem is considered as a bird in the search space, the bird is the particle. First, build a initial bird group. That is to randomly initialize a group of particles in the feasible solution space. Each particle is a feasible solution of the classification problem, and a fitness value is determined by the objective function. Each particle is moving in the solution space, and its flight direction and distance are determined by the movement speed. In each iteration, the particles will follow the two values to update itself. One is the best solution to the particle's own search. One is the optimal solution for the whole population, in the present, that is the global extreme value.

Classification Model (Text PSO-Miner).

In this paper, the rule mining algorithm based on particle swarm was introduced into text classification, and established the text classification model, Text PSO-Miner.

In Text PSO-Miner, each particle corresponds to a path, producing a classification rule. Rule is a line connecting the attribute node and class node. Each attribute node appears only once or not, and must have a class node. Attribute node corresponds to the text characteristic value. As showed in Fig.1.

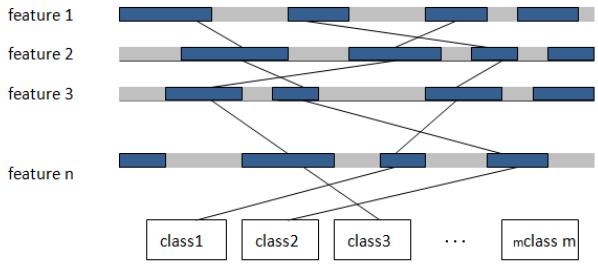

Fig.1 Classification Rules of Text PSO-Miner

In Fig.1, each path corresponds to a classification rule. Mining of classification rules can be understood as the search for the optimal solution in the multi dimension space. The classification rules are as follows:

IF ＜ term$_1$AND term$_2$ AND… ＞ THEN ＜ class ＞.

Where, term$_i$ is the condition item, term=<feature number, operator, feature value>.

PSO treats each solution as a particle in a *D* dimensional space without a volume. In Text PSO-Miner, each particle corresponds to a classification rule. Each rule need to find out optimal eigenvalue range $[x_-,x_+]$ for each feature. x_- is the lower bound of the optimal range, x_+ is the upper bound.

Suppose that there are *m* particles, each particle is composed of three parts, namely the particle position, speed and fitness value. Particle *i* can be expressed as:

$$Particle(i)=\{location[],velocity[],fitness\} \quad (1)$$

location[] is the location vector of particle *i*. Expressed as:

$$Particle(i).location[]=\{x_{-i1},x_{+i1}, x_{-i2},x_{+i2,...,} x_{-id},x_{+id}\} \quad (2)$$

velocity[] is the velocity vector of particle *i*. Expressed as:

$$Particle(i).velocity[]=\{x_{-i1},x_{+i1}, x_{-i2},x_{+i2,...,} x_{-id},x_{+id}\} \quad (3)$$

Particle(*i*).*fitness* is the fitness value of particle *i*, it is a real number.

In the process of evolution, each particle also has a self optimal solution P_i, which records the optimal position of the particle and the corresponding fitness value. P_i expressed as:

$$P_i=\{location[],fitness\} \quad (4)$$

There is a global optimal solution P_g for the whole particle swarm, and the optimal position and corresponding fitness value of the particle swarm are recorded. P_g expressed as:

$$P_g=\{location[],fitness\} \quad (5)$$

Training Process of Text PSO-Miner.

The training process of Text PSO-Miner requires 3 stages: rule structure, fitness calculation, and the coverage of training data.

Rule structure process simulates birds foraging behavior. The optimal upper bound and optimal lower bound for the particle are searched in the range of each characteristic value. If text data is a

d-dimensional feature, then search the optimal value in 2*n*-dimensional space. The optimal interval of each feature is connected with "AND", and then the text class is joined to form a rule. Fitness is used to measure the position of the particle, it is used to determine the direction of flight. So it is very important to choose the fitness function to solve the problem. The coverage of the training data is to put a global optimal solution P_g into the rule set *RS*, and then the data is shifted out of the training data P_g with a sequence coverage method. If the number of data in the training data is less than the threshold, then terminate the rule mining of this class, and turn to the next class, until the end. So, the training process of Text PSO-Miner is crucial, its pseudo code is as follows:

```
Procedure：Text PSO-MinerTrain
RS= Ø%%Initialization rule set RS is empty
FOR（i=1,i++,i<=M）%%The training set consists of M classes
    TS=Train Set %%Set TS as training text vector set
    WHILE(TSi >threshold) %%When the number of text vector of class i is greater than the threshold
BestRule＝PSORuleConstructer（）%%Run the program, Fig. 2
        RS=RS＋BestRule %%Update rule set RS
        TS=TS–CTR%%CTR is a text vector that has been covered by the found rule
    END WHILE
END FOR
```

In the pseudo code, PSORuleConstructer() flow chart is shown in Fig.2:

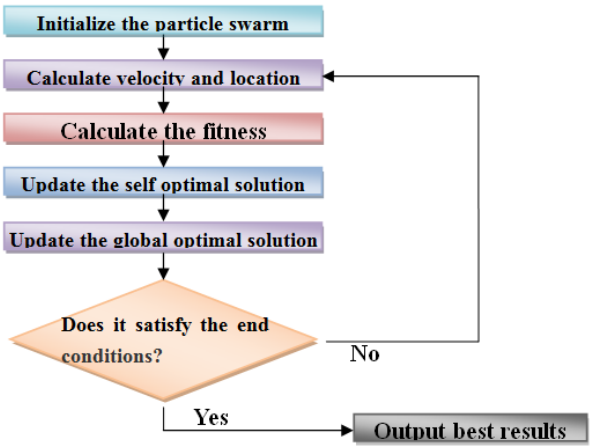

Fig.2 PSORuleConstructer() flow chart

The operations to be carried out are:

Initialize the particle swarm. The initial particles are randomly distributed in the 2*d*-dimensional space, the initial particle I position is:

$$Particle(i).location[]=\{x_{-i1},x_{+i1}, x_{-i2},x_{+i2},..., x_{-id},x_{+id}\} \quad (6)$$

$$x_{-ij}=term_{jmin}()+rand()\times(term_{jmax}()-term_{jmin}()) \quad (7)$$

$$x_{+ij}=term_{jmin}()-rand()\times(term_{jmax}()-term_{jmin}()) \quad (8)$$

Where, x_{-ij} and x_{+ij} are the lower and upper bounds of the *i* particle on the characteristics of *j*. If $x_{-ij} > x_{+ij}$, then x_{-ij} and x_{+ij} exchange. *rand*() is a random number in [0,1]. $term_{jmax}()$ and $term_{jmin}()$ represent the minimum eigenvalue and the maximum eigenvalue of the feature *j* respectively.

The initial velocity of the particle *i* is:

$$Particle(i).velocity[]=\{x_{-i1},x_{+i1}, x_{-i2},x_{+i2},..., x_{-id},x_{+id}\} \quad (9)$$

$$v_{-ij}=rand()\times v_{-jmax} \quad (10)$$

$$v_{+ij}=rand()\times v_{+jmax} \quad (11)$$

Where, v_{-jmax} and v_{+jmax} are the maximum speed of the lower bound and upper bound. The fitness value of each particle is calculated, and the value is assigned to a self optimal solution P_i. The optimal fitness value of the entire particle swarm is calculated, and the value is assigned to the global optimal solution P_g.

Calculate velocity and location. The velocity and location of the particles are updated according to the following.

$$Particle(i).velocity[] = \omega Particle(i).velocity[]$$

$$+\eta_1 rand()(P_i.location[] - Particle(i).location[])$$
$$+\eta_2 rand()(P_g.location[] - Particle(i).location[]) \quad (12)$$

$$Particle(i).location[] = Particle(i).location[] + Particle(i).velocity[] \quad (13)$$

Where, ω indicates the inertia weight, the coefficient of keeping the current velocity. η is a learning factor. η_1 represents the importance of the self optimal solution P_i. η_2 indicates the importance of the global optimal solution P_g, representing information sharing and cooperation among particles.

Calculate the fitness. The fitness value of each particle is calculated according to the following formula.

$$Particle(i).fitness = Recall_i \times Precision_i \quad (14)$$

Update the self optimal solution. If the particle fitness value is better than its self optimal solution P_i, then update P_i, otherwise it will not be updated.

Updated the global optimal solution. The optimal fitness value of all particles is calculated. If the value is better than the global optimal solution P_g, P_g is updated or not.

If the end condition is reached (good enough location or maximum number of iterations), then the end, otherwise returns to step "calculate velocity and location".

Model Validation

Verification steps.

In this paper, the Chinese text data set is provided by Ronglu Lee, working at the natural language processing group of the international database center, the computer information and Technology Department of Fudan University. From this data set, we randomly select 3240 documents as the experimental data, Including military, sports, transportation, economy. Firstly, the experimental data set is divided into the training set A and the test set B according to the ratio of 1:1.Then, the A and B are input to the preprocessing program, and the information gain, the χ^2 statistics, the mutual information, the text evidence right, the expected cross entropy were selected as feature selection method. 5 sets of vector space matrix corresponding to the feature selection method are obtained, Training set vector space matrix, denoted as A_j(j=1,2,3,4,5), Test set vector space matrix, denoted as B_j(j=1,2,3,4,5). Then A_j(j=1,2,3,4,5) and B_j(j=1,2,3,4,5) are input to WEKA[15] and SVMlight[16]. Classification results C_j(j=1,2,3,4,5) of SVM, KNN and NB are obtained. Choose the best results of C_k as a benchmark for evaluating the effect of Text PSO-Miner, then, input A_k (corresponding to C_k) to Text PSO-Miner to get classification rules. Use this rule to classify B_k (also corresponding to C_k), get the classification results D_k. Compare C_k and D_k.

Experimental parameters is set to: feature dimensions: 50; KNN adjacent values: 35; kernel function of SVM: radial basis function (RBF); training volume datasets: 1620; test set data: 1620; class number: 4; the inertia weight ω=0.8; the learning factor η_1=η_2=2; the maximum number of iterations: 200; the remaining the small sample number: 5.

Evaluation criteria.

Classification results are evaluated by macro average recall rate (MR), macro average precision (MP), micro averaged recall (mR), micro precision average (mP), macro average measure F value (MF) and micro average measure F value (mF). Classification results of Naive Bayesian (NB), k nearest neighbor algorithm (KNN), support vector machine (SVM) will be used as a benchmark for evaluating the effect of Text PSO-Miner.

Experimental results.

SVM, KNN, and NB classification models were selected to obtain the mean value of MF and mF. By the feature selection method, such as information gain, χ^2 statistics, mutual information, text evidence right, expected cross entropy, etc..In comparison, the effect of using the information gain on the data set is the best. Therefore, the feature selection method for the following classification models will adopt the information gain. See Table 1.

Table 1 MF and Mf mean values of each model

	information gain	χ^2 statistics	mutual information	text evidence right	expected cross entropy
SVM	0.930	0.920	0.366	0.860	0.924
KNN	0.922	0.909	0.618	0.854	0.905
NB	0.742	0.693	0.647	0.656	0.699

After training, Text PSO-Miner found a total of 15 classification rules. The confusion matrix of Text PSO-Miner classification results is shown in Table 2. Text PSO-Miner classification performance index values, as shown in Table 3.

Table 2 The confusion matrix of Text PSO-Miner

	Military	Sports	Transportation	Economy
Military	121	3	0	1
Sports	6	218	0	1
Transportation	6	3	99	4
Economy	7	6	4	136

Table 3 Classification performance index values of each model

	MR	MP	mR	mP	MF	mF
KNN	0.917	0.927	0.929	0.914	0.922	0.921
NB	0.757	0.814	0.768	0.642	0.784	0.699
SVM	0.932	0.929	0.937	0.922	0.930	0.930
PSO	0.947	0.933	0.947	0.933	0.940	0.940

Conclusion

In this paper, we develop a more mature ant colony algorithm in the field of text classification, and construct a text classification model based on ant colony intelligence. The results show that the Text PSO-Miner text classification model can be applied to text classification. Limited to the author's ability and time, the work of this paper is to introduce ant colony algorithm to text classification. The research is just a start, and a lot of work needs to be further studied, for example, the selection of the target function and the formulation of information update and group collaboration mechanism.

Acknowledgment

The research is financially supported by the National Natural Science Foundation of China (Grant No. 51209095, 51210013,51579105), the Fundamental Research Funds for the Central Universities (2014ZZ0027).

References

[1] Kennedy J,Eberhart R C.Particle swarm optimization[C].Procerence of IEEE Int’1 conference on Neural Networks.Piscataway,NJ,WA Australia:IEEE service center,Vloume 4,p.1942-1948,1995.

[2] S. Payan, A. Farahmand, S.M. Hosseini Sarvari, "Inverse boundary design radiation problem with radiative equilibrium in combustion enclosures with PSO algorithm",International Communications in Heat and Mass Transfer, Volume 68, Pages 150-157, November 2015.

[3] Abdulla Amin Aburomman, Mamun Bin Ibne Reaz ,"A novel SVM-kNN-PSO ensemble method for intrusion detection system",Applied Soft Computing, In Press, Uncorrected Proof, Available online 23 October 2015

[4] T. Sousa, A. Silva, and A. Neves, "Particle swarm based data mining algorithms for classification tasks", Parallel Computing, vol. 30, no. 5-6, p. 767-783, 2004.

[5] Nicholas Holden and Alex A. Freitas, "A Hybrid PSO/ACO Algorithm for Discovering Classification Rules in Data Mining", Journal of Artificial Evolution and Applications, Volume 2008 (2008), Article ID 316145, 11 pages. http://dx.doi.org/10.1155/2008/316145