Applied Soft Computing 35 (2015) 629–636

Contents lists available at ScienceDirect

Applied Soft Computing

journal homepage: www.elsevier.com/locate/asoc

Improved particle swarm optimization algorithm and its application in text feature selection

Yonghe Lu*, Minghui Liang, Zeyuan Ye, Lichao Cao

School of Information Management, Sun Yat-sen University, GuangZhou, China

ARTICLE INFO

Article history:
Received 8 September 2014
Received in revised form 6 June 2015
Accepted 6 July 2015
Available online 15 July 2015

Keywords:
Text classification
Text feature selection
Particle swarm optimization algorithm
Constriction factor

ABSTRACT

Text feature selection is an importance step in text classification and directly affects the classification performance. Classic feature selection methods mainly include document frequency (DF), information gain (IG), mutual information (MI), chi-square test (CHI). Theoretically, these methods are difficult to get improvement due to the deficiency of their mathematical models. In order to further improve effect of feature selection, many researches try to add intelligent optimization algorithms into feature selection method, such as improved ant colony algorithm and genetic algorithms, etc. Compared to the ant colony algorithm and genetic algorithms, particle swarm optimization algorithm (PSO) is simpler to implement and can find the optimal point quickly. Thus, this paper attempt to improve the effect of text feature selection through PSO. By analyzing current achievements of improved PSO and characteristic of classic feature selection methods, we have done many explorations in this paper. Above all, we selected the common PSO model, the two improved PSO models based respectively on functional inertia weight and constant constriction factor to optimize feature selection methods. Afterwards, according to constant constriction factor, we constructed a new functional constriction factor and added it into traditional PSO model. Finally, we proposed two improved PSO models based on both functional constriction factor and functional inertia weight, they are respectively the synchronously improved PSO model and the asynchronously improved PSO model. In our experiments, CHI was selected as the basic feature selection method. We improved CHI through using the six PSO models mentioned above. The experiment results and significance tests show that the asynchronously improved PSO model is the best one among all models both in the effect of text classification and in the stability of different dimensions.

1. Introduction

Particle swarm optimization algorithm (PSO) is an evolutionary computing algorithm proposed by James Kennedy and Russell Eberhart in 1995[1]. PSO was originally used to graphically simulate the process of bird folks finding food. By observing flock of birds' behaviors, the researchers found that sharing information in groups was beneficial to gain advantage during evolution. This made up the basis of PSO [2].

The academic research about modifying PSO mainly focuses on improving its parameters by theoretical analysis, mathematical inference and empirical research. For example, Angeline et al. [4] introduced standard selection mechanism into PSO, the optimum particle was selected after each iteration and copied it to next generation. This method was particularly beneficial for complex function optimization. Lovbjerg et al. [5] quoted the concept of the sub-population and the reproductive into PSO in order to get faster convergence. Higashi et al. [6] used Gaussian mutation to redesign the changing rules of particle in terms of position and speed, and got better results on unimodal and multimodal function. Baskar et al. [7] designed two particle swarms which worked in parallel and exchanged information to overcome the PSO shortcoming of premature convergence. In addition, there were studies of quantum PSO [8], Niche PSO [9], Division of Labor in PSO [10], Hierarchical PSO [11] and other improved algorithms. Clerc added constriction factor K into PSO to ensure convergence [12]. Bergh studied the effect of randomness on particle trajectories in iteration [13]. Bergh et al. also proved that the original version of PSO could not converge to the global optimum, and proposed an improved PSO which ensured convergence to local optimal solution [14]. As for the inertia weight in PSO, Chatterjee et al. proposed an inertia weight with nonlinear variation [15]. Besides, Nickabadi et al. came up with an adaptive inertia weight [16]. Bansal et al. conducted comparing experiments for 15 kinds of method of inertia

* Corresponding author. Tel.: +86 13610219952.
E-mail address: luyonghe@mail.sysu.edu.cn (Y. Lu).

http://dx.doi.org/10.1016/j.asoc.2015.07.005
1568-4946/

weight calculation [17]. The results showed that the chaotic inertia weight was the best in terms of effect, and the random inertia weight was the highest in efficiency.

Classification is the process in which ideas and objects are recognized, differentiated, and understood [18]. Classification implies that objects are usually grouped into required categories for some specific purpose. There are many kinds of classification, including power quality classification, nonstationary power signal time series data classification [19], text classification, etc. Recently, biological evolution algorithms are more commonly used in classification to improve accuracy. For example, B. Biswal et used Bacterial Foraging Optimization Algorithm in classification of power quality data [20]. Luo Xin et used Ant Colony Optimization Algorithm to construct a text classification model [21]. Biswal et used Particle Swarm Optimization Algorithm to improve Fuzzy C-Means Algorithm. This improved algorithm was utilized into Time Frequency Analysis and Non-Stationary Signal Classification and Power Quality Disturbance Classification [22,23]. Lu Yonghe and Liang Minghui utilized Genetic Algorithm to optimize text feature selection method in text classification [24].

In this paper, we focus on using Particle Swarm Optimization Algorithm to improve performance of text classification. Under the given classification system, text classification was defined as a process which automatically identify text category according to the text content [25]. Traditionally, scholars put forward many improved methods for text classification by optimizing mathematic model. For example, Lu Yonghe and Li Yanfeng optimized text feature weighting method based on TF-IDF algorithm [26]. Lu Yonghe and He Xinyu added similarity matrix and dimension index table into KNN to improve KNN classification algorithm [27,28]. Wei Tingting et added WordNet and lexical chains to optimize text clustering model [29]. Currently, in the field of text classification research, the improved PSO has three aspects of application. The first one is to optimize the classic text classifier, such as KNN, SVM, etc. [30]. Li Huan et al. proposed a simplified PSO KNN classification algorithm [31]. Tang Zhaoxia used the PSO algorithm to find k neighbors in order to improve the efficiency of web text classification [32]. Tuo Shouheng improved the inertia weight in PSO and added this method into SVM classification [33]. The second one is to build a text classifier using PSO. Luo Xin constructed the text classification model based on PSO [34]. Tong Yala et al. proposed extraction method for classification rule based on chaos PSO [35]. Similarly, Tan Dekun also proposed a text classification method based on chaos PSO [36]. The last one is the use of PSO for text feature selection [30]. Chih-Chin Lai et al. studied the application of PSO in text feature selection for spam classification [37]. This was a specific application using PSO to optimize text feature selection. Yaohong Jin et al. used the improved PSO in Chinese text feature selection [38]. Likewise, Zahran et al. proposed an improved feature selection method based PSO in order to solve the problem of Arabic text feature selection [39]. HK Chantar et al. also analyzed the characteristics of the Arabic text classification and used binary PSO for text feature selection to improve accuracy of text classification [40].

In this paper, we firstly improve particle swarm optimization algorithms, and then apply them to text feature selection. Finally, we analyze the application effect of various improved particle swarm programs by using KNN classifier.

2. Improvement of PSO

2.1. Traditional particle swarm optimization algorithm

PSO uses a number of particles, which constitute a swarm moving around in the search space, to look for the best solution. Each particle is treated as a point in a D-dimensional space, which adjusts it's "flying" according to its own flying experience as well as the flying experience of other particles. The particles flight with a certain velocity in the D-dimensional space to find the optimal solution. The velocity of particle i expresses as $V_i = (v_{i1}, v_{i2}, \ldots, v_{iD})$, the location of particle i expresses as $(x_{i1}, x_{i2}, \ldots, x_{iD})$, the optimal location of particle i expresses as $P_i = (p_{i1}, p_{i2}, \ldots, p_{iD})$, it is also called p_{best}. The global optimum position of all particles expresses as $P_g = (p_{g1}, p_{g2}, \ldots, p_{gD})$, it is also called g_{best}. Each particle in group has a fitness function to calculate the fitness value. In standard PSO, the velocity update formula of the dimension d shows in formulae (1) and (2):

$$v_{id} = w \times v_{id} + c_1 \times rand(\ ) \times (p_{id} - x_{id}) + c_2 \times Rand(\ ) \times (p_{gd} - x_{id}) \tag{1}$$

$$x_{id} = x_{id} + v_{id} \tag{2}$$

PSO parameters include: Q (Population Quantity), w (inertia weight), C_1 and C_2 (acceleration constants), $v_{\max}$ (the maximum velocity), $G_{\max}$ (the maximum number of iterations), rand ( ) and Rand ( ) are random functions with values in [0,1]. The value of C_1 and C_2 usually takes constant 2 [3].

2.2. Improve inertia weight

Inertia weight is an important parameter of the standard PSO. It determines the operating results of PSO. Fixed inertia weight make the particles always have the same exploration competence in flight. Formula (3) is the velocity formula with a fixed inertia weight in traditional PSO [3]:

$$v_{id}(t+1) = w \times v_{id}(t) + c_1 \times rand(\ ) \times (p_{id} - x_{id}(t)) + c_2 \times Rand(\ ) \times (p_{gd} - x_{id}(t)) \tag{3}$$

According to experience, w generally takes between 0 and 1 [3]. Thus, w is 0.9 in this paper. The velocity formula (3) becomes formula (4). (Here, we call it Program 1):

$$v_{id} = 0.9 \times v_{id} + 2 \times rand(\ ) \times (p_{id} - x_{id}) + 2 \times Rand(\ ) \times (p_{gd} - x_{id}) \tag{4}$$

Currently, the conventional strategy of improving inertia weight is LDIW (Liner Decreasing Inertia Weight) [41]. The changing way of w appears in formula (5). (Here, we call it Program 2):

$$v_{id} = w \times v_{id} + 2 \times rand(\ ) \times (p_{id} - x_{id}) + 2 \times Rand(\ ) \times (p_{gd} - x_{id})$$

$$\begin{cases} w = w_{\text{end}} + (w_{\text{start}} - w_{\text{end}})(1 - \dfrac{T}{G_{\max}}) & \text{if } p_{gd} \neq x_{id} \\ w = w_{\text{end}} & \text{if } p_{gd} = x_{id} \end{cases} \tag{5}$$

where T is the number of iterations, $T \in [0, G_{\max})$, p_{gd} is global best position, w_{start} is the initial inertia weight and w_{end} is the value of evolution in the maximum iterations. w_{start} and w_{end} are calculated in subsequent chapters.

2.3. Improve constriction factor

Because the dimensions of a text feature vector in Text Categorization are usually very high, the particles in PSO will gather into a point when it is not yet to find the global optimum [42]. Thus, Clerc introduced constriction factor K into PSO to ensure the best