Expert Systems with Applications 42 (2015) 2517–2524

Contents lists available at ScienceDirect

Expert Systems with Applications

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

A hybrid evolutionary computation approach with its application for optimizing text document clustering

Wei Song [a,*], Yingying Qiao [a], Soon Cheol Park [b], Xuezhong Qian [a]

[a] School of Internet of Things Engineering, Jiangnan University, China
[b] Division of Electronics and Information Engineering, Chonbuk National University, Republic of Korea

ARTICLE INFO

Article history:
Available online 8 November 2014

Keywords:
Knowledge discovery and management
Evolutionary computation
Particle swarm optimization
Quantum-behaved particle swarm optimization

ABSTRACT

Quantum-behaved particle swarm optimization (QPSO) is a promising global optimization algorithm inspired by concepts of quantum mechanics and particle swarm optimization (PSO). Since the particles are initialized randomly in QPSO, the blindness of initializing particles affects its capacity for complicated optimization. In this paper, we make full use of a hybrid evolutionary computation approach to resolve such an issue. In specific, the robust global search ability of genetic algorithm (GA) improves the initial strategy of particles in QPSO. What is more, the original position update approach of QPSO without the restriction of its upper bound may generate some abrupt features and cause the issue of overstepping boundary, which affects its performance for search of optimum. In this study, a new position update approach is tested to normalize the search range of particles in a proper space. Such an approach enhances its probability to find the optimal solution. Since the clustering problem can be regarded as the centers searching process by using evolutionary optimization approach, the evolutionary process of chromosomes or particles encoded by centers simulates the process of solving clustering problem. In order to testify the clustering performance of our approach, we conduct the experiments on 4 subsets of standard Reuter-21578 and 20Newsgroup datasets. Experimental results show that our method performs better than the state of art clustering algorithms in the light of the evaluations of fitness and F-measure.

1. Introduction

Clustering of text documents plays a vital role in efficient document organization, summarization, topic extraction and information retrieval (Kowalski, 1997). It organizes the text sets in terms of the nature of the text content and makes the entire text collection into several clusters through the corresponding algorithm. As a result, the text documents in the same cluster are more similar to each other than those assigned to different clusters. To differ from the popular categorization approach of being pre-defined as a supervised method, clustering is an unsupervised classification technique without training process and manual annotation to objects. It has certain flexibility and high ability of automatic processing.

The most commonly used clustering algorithm is K-means. This algorithm is very simple and has been widely used for study and application of clustering. However, the clustering result is sensitive to the initial clustering centers and may fall into local optimum (Selim & Ismail, 1984). FCM is another clustering algorithm based on partitioning. Unlike K-means, FCM is a kind of flexible fuzzy partition algorithm. The obvious advantages of FCM are good adaptability and the ability to resist noise, but the defects of K-means also exist in FCM. As a representative of clustering algorithms based on density, DBSCAN has strong noise resistance and high clustering accuracy. But the main threshold parameters are difficult to specify and the time complexity is too high, which often lead to a poor clustering effect. Different from the traditional search algorithms, Genetic Algorithm (GA) is a random search and optimization algorithm presented by professor Holland (1975). GA imitates the process of biologic evolution and inheritance in accordance with Darwin's competition principle of "survival of the fittest, and discard the inferior" (Jones, Robertson, Santimetvirul, & Willett, 1995). It can realize the optimization function through reproductive evolution of the individual advantage in a population (Song, Liang, & Park, 2014). However, its local search ability is relatively weak. As a new stochastic optimization technology, swarm intelligence technology has emerged and been successfully applied to a number of real world clustering applications (Abraham, Das, & Roy, 2008). In recent decade, the implementations of optimization and control algorithms based on swarm intelligence, e.g., Ant Colony Optimization (ACO) and particle swarm optimization

* Corresponding author.
E-mail addresses: songwei@jiangnan.edu.cn (W. Song), yingyqiao@foxmail.com (Y. Qiao), scpark@jbnu.ac.kr (S.C. Park), qxzvb@163.com (X. Qian).

http://dx.doi.org/10.1016/j.eswa.2014.11.003
0957-4174/

(PSO), have been extensively studied. The former is inspired by the foraging behavior of the ant colony and searches the optimal solution by simulating the real collaborative process of ant colony (Mohan & Baskaran, 2012). Since ACO needs large amount of calculation and longer time for searching, as a result the convergence is still slow, and it may fall into local optimum (Mustafa, Mesut, & Ömer, 2012). Particle swarm optimization algorithm imitates the feeding process of the bird population and searches the optimal direction by coordinating the individual and the group (Kuo et al., 2012). In study of cui et al. the PSO was firstly presented as a text documents clustering algorithm (Cui, Potok, & Palathingal, 2005). As the modification of PSO, quantum-behaved particle swarm optimization (QPSO) was proposed by Sun, Feng, and Xu (2004). The inspiration of QPSO stems from quantum mechanics and trajectory analysis of the individual behavior of PSO (Clerc & Kennedy, 2002; Zhang, 2010). Since PSO has been proved to be not a well global convergent algorithm (Van den Bergh & Engelbrecht, 2002), QPSO as a variant of PSO has been proved in order to improve the global search ability of the classical PSO (Sun, Fang, Wu, Palade, & Xu, 2012).

High-dimensional data processing is a contemporary challenge in the scope of engineering (Lu, Wang, Li, & Zhou, 2009; Lu, Wang, Li, & Zhou, 2011). With the rapid growth of text reports and publications on net, the text data sets are usually composed of tens of thousands of keywords. As a result, complex clustering problems with high-dimensional features make the existing clustering algorithms fail to satisfy the request of modern information retrieval system. The performance of single algorithm has its own restrictions based on the respective design of the algorithm. Moreover, the poor optimization ability with premature convergence is the common problem of almost all stochastic optimization algorithms. Thus, combining different swarm intelligence algorithms to improve the optimization ability is a new research hotspot in the current engineering optimization field. It has known that QPSO is a probabilistic global optimization algorithm inspired by the concepts from quantum mechanics and PSO. GA is a stochastic global optimization algorithm guided by the principles of selection and heredity. The robust global search ability is its outstanding advantage. In this paper, a hybrid method combining GA and QPSO (GQPSO) is proposed to tackle with complex text clustering problem. That is, we make full use of GA to solve the blindness problem of initializing particles in QPSO. In specific, a preliminary optimization using GA is performed to yield an initial partition, and particles are then initialized by such a preparation for subsequent evolving process of QPSO. Moreover, the original position update approach of QPSO without the restriction of its upper bound may generate some abrupt features and cause the issue of overstepping boundary, which affects its performance for search of optimum. In this study, a new position update approach is tested to normalize the search range of particles in a sound space. Such an approach can enhance its probability to find the optimal solution.

The remainder of this paper is organized as follows: Section 2 provides a general overview of the related work about this study. The GQPSO clustering algorithm is presented in Section 3. Section 4 describes the methods of representing documents and computing similarity. Section 5 provides the detailed experimental setup and the discussion of the experimental results. The conclusion is given in Section 6.

2. Related work

2.1. Genetic algorithm

Genetic Algorithm (GA) is a kind of effective optimization method based on the principle of natural selection and genetics (Song & Park, 2009). In GA, the parameters of the search space are encoded in the form of strings, called chromosomes. A collection of chromosomes is called a population and a random distributed population is created first, like different points in the search space (Maulik & Bandyopadhyay, 2000; Song, Li, & Park, 2009). The process of solving problems is represented as evolutionary process of chromosomes. At the initial population, a number of individuals are selected with the selection strategy, according to the fitness function associated with each chromosome. Then selection, crossover and mutation operators are applied to produce the population of the next generation. The iteration will go on, until the termination condition is satisfied.

In text clustering problem, a chromosome can be represented as:

$$X_i = (M_{i1}, \ldots, M_{ij}, \ldots M_{ik}) \tag{1}$$

Where $M_{ij} = (w_1, w_2, \ldots, w_n)$ refers to the center vector of the jth cluster in the ith chromosome. n is the total number of terms, and k is the number of centers. It has known that, the clustering problem can be regarded as the searching process of center to corresponding cluster. Note that, each chromosome is encoded with the combination of centers, which represents the candidate solution to the clustering problem. Thus, the evolutionary process of chromosomes encoded by centers imitates the process of solving clustering issue. In GA, an objective and fitness function associated with each chromosome represents the degree of fitness to each solution. That is, through three evolutionary operators, i.e. selection, crossover, and mutation, the objective and fitness function guides the evolving direction of the chromosome. In this study, the fitness function for the ith chromosome is measured as the equation:

$$fitness(i) = \sum_{j=1}^{k} \sum_{d_i \in C_{ij}} \cos(d_i, M_{ij}) \tag{2}$$

where d_i is a document vector belonging to cluster C_{ij}. M_{ij} refers to the jth cluster center vector of the cluster C_{ij}. $\cos(d_i, M_{ij})$ is the cosine measure between d_i and M_{ij}. From Eq. (2) we can see that the bigger of the fitness, the better of the chromosome associated with the solution to the clustering problem.

2.2. Particle swarm optimization algorithm

Inspired by the social behavior of a flock of birds, particle swarm optimization (PSO) was originally developed by Kennedy and Eberhart (1995). In PSO the particles representing potential solutions, move around in a multidimensional search space with a velocity constantly updated by its own experience and the experience of its neighbors or whole swarm (Sun, Fang, Palade, Wu, & Xu, 2011).

PSO has been successfully implemented in many research and application areas. Since it can be easily implemented and is computationally inexpensive, only a small number of parameters have to adjust. When applying it for clustering, each particle represents the candidate solution to the clustering, and the evolutionary process of particles encoded by centers simulates the process of solving clustering issue. PSO also has been proven to be effective for data clustering (Cohen & de Castro, 2006; Oliveira, Britto, & Sabourin, 2005; Shi & Eberhart, 1998; Van der Merwe & Engelbrecht, 2003).

In PSO, the velocity and direction of each particle moving along each dimension of the problem space will be altered with each generation of movement (Cui & Potok, 2005). That is to say, when a particle moves to a new position, a new candidate solution is generated. Every particle in the swarm is updated using Eqs. (3) and (4).