Neurocomputing 172 (2016) 371–381

Contents lists available at ScienceDirect

Neurocomputing

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

Binary grey wolf optimization approaches for feature selection

E. Emary [a,1], Hossam M. Zawbaa [b,c,*,1], Aboul Ella Hassanien [a,b,1]

[a] Faculty of Computers and Information, Cairo University, Egypt
[b] Faculty of Computers and Information, Beni-Suef University, Egypt
[c] Faculty of Mathematics and Computer Science, Babes-Bolyai University, Romania

ARTICLE INFO

Article history:
Received 13 February 2015
Received in revised form
28 June 2015
Accepted 29 June 2015
Available online 6 August 2015

Keywords:
Grey wolf optimization
Binary grey wolf optimization
Feature selection
Bio-inspired optimization
Evolutionary computation

ABSTRACT

In this work, a novel binary version of the grey wolf optimization (GWO) is proposed and used to select optimal feature subset for classification purposes. Grey wolf optimizer (GWO) is one of the latest bio-inspired optimization techniques, which simulate the hunting process of grey wolves in nature. The binary version introduced here is performed using two different approaches. In the first approach, individual steps toward the first three best solutions are binarized and then stochastic crossover is performed among the three basic moves to find the updated binary grey wolf position. In the second approach, sigmoidal function is used to squash the continuous updated position, then stochastically threshold these values to find the updated binary grey wolf position. The two approach for binary grey wolf optimization (bGWO) are hired in the feature selection domain for finding feature subset maximizing the classification accuracy while minimizing the number of selected features. The proposed binary versions were compared to two of the common optimizers used in this domain namely particle swarm optimizer and genetic algorithms. A set of assessment indicators are used to evaluate and compared the different methods over 18 different datasets from the UCI repository. Results prove the capability of the proposed binary version of grey wolf optimization (bGWO) to search the feature space for optimal feature combinations regardless of the initialization and the used stochastic operators.

1. Introduction

Feature selection provides a way for identifying the important features and removing irrelevant (redundant) ones from the dataset [1]. The feature selection objectives are data dimensionality reduction, improving prediction performance, and good data understanding for different machine learning applications [2]. In the real world applications, data representation often uses too many features with redundancy features, which means certain features can take the role of another and the unnecessary features can be removed. Moreover, the relevant (interdependence) features have an influence on the output and contain important information that will be obscure if any of them is excluded [3].

Previously, an exhaustive search for the optimal set of features (attributes) in a high dimensional space may be unpractical. Many researches try to model the feature selection as a combinatorial optimization problem, which the set of features lead to the best feature space separability [4]. The objective function can be the classification accuracy or some other criterion that might consider the best trade-off between attribute extraction computational burden and efficiency [5].

The classical optimization techniques have some restriction in solving the problems, so that evolutionary computation (EC) algorithms are the alternative for solving these limitations and searching for the optimum solution of the problems. Evolutionary computation (EC) algorithms are inspired from nature, social behavior, and biological behavior of (animals, birds, fish, bat, firefly, wolves, etc.) in a group. Many researchers have proposed different computational methods, in order to mimic the behavior of these species to seek for their food (optimal solution) [6].

Various heuristic techniques mimic the behaviour of biological and physical systems in the nature and it has been proposed as strong methods for global optimizations. Genetic algorithms (GA) was the first evolutionary based algorithm introduced in the literature and has been developed based on the natural process of evolution through reproduction. GA has the ability to solve the complex and non-linear problems. Moreover, GA has some disadvantages such as low performance and sticking in local minima [7]. Particle swarm optimization (PSO) is one of the well-known swarm algorithms. In particle swarm optimization (PSO), each solution is considered as a particle with specific characteristics (position, fitness, and a speed vector) which defines the moving direction of the particle [8].

* Corresponding author at: Mihail Kogalniceanu 1, Cluj-Napoca, 400084, Romania. Tel.: +40 754 887 686.
E-mail address: hossam.zawbaa@gmail.com (H.M. Zawbaa).
[1] Scientific Research Group in Egypt (SRGE), http://www.egyptscience.net

http://dx.doi.org/10.1016/j.neucom.2015.06.083
0925-2312/

Artificial bee colony (ABC) is a numerical optimization algorithm based on foraging behavior of honeybees. In ABC, the employer bees try to find food source and advertise the other bees. The onlooker bees follow their interesting employer, and the scout bee fly spontaneously to find the best food source [9]. A virtual bee algorithm (VBA) is applied to optimize the numerical function in 2-D using a swarm of virtual bees, which move randomly in the search space and interact to find food sources. From the interactions between these bees results the possible solution for the optimization problem [10]. A proposed approach based on natural behavior of honeybees, which randomly generated worker bees are moved in the direction of the elite bee. The elite bee represents the optimal (near to optimal) solution [11].

In optimization algorithms, it is essential to have a convenient balance between exploration and exploitation. In a bee swarm algorithms, different behaviors of the bees give us the possibility to create robust balancing technique between exploration and exploitation [12]. Artificial fish swarm (AFS) algorithm mimics the stimulant reaction by controlling the tail and fin. AFS is a robust stochastic technique based on the fish movement and its intelligence during the food finding process [13].

A binary version of the particle swarm optimization (BPSO) modifies the old version of PSO algorithm to deal with the binary optimization problems [14]. Moreover, an extended version of BPSO is used to deal with feature selection problems [15]. The search space in BPSO is considered as a hypercube; a particle may be seen to move to nearer or farther corners of the hypercube by flipping various numbers of bits [16]. Furthermore, a binary version of the gravitational search algorithm (BGSA) is used for feature selection issue [17]. A binary version of the bat algorithm (BBA) is applied for feature selection purposes, where the search space is modelled as an n-cube. It is important to assign for every bat a set of binary coordinates that indicate if this feature will belong to the final feature set. So, the optimal (near to optimal) solution corresponds to one hypercube's corner [18]. There are many other researches with the same idea, applied in a wide range of bio-inspired algorithms [19–21,23–27].

Grey wolf optimization (GWO) is a newly introduced evolutionary algorithm, which proposes that the grey wolves have a successful reproduction more than hunting in the pack. Two grey wolves (male and female) have a higher position and managing the other wolves in the pack [6]. In this paper, a novel binary version of the grey wolf optimization is proposed to find optimal regions of the complex search space. Grey wolf optimizer is one of the latest bio-inspired techniques, which simulate the hunting process of a pack of grey wolves in nature. The binary version introduced here is performed using two different approaches.

The organization of this paper as the following: Section 2 presents the background of continuous grey wolf optimization (CGWO). The proposed new version of grey wolf optimization (GWO) describes in Section 3. Section 4 presents a binary version of grey wolf optimization (BGWO) for feature selection. The experimental results are discussed in Section 5. Finally, conclusions are stated in Section 6.

2. Continuous grey wolf optimization (CGWO)

Mostly, grey wolves prefer to live in a pack. The group size is 5–12 on average. They have very strict rules in social dominant hierarchy. According to [22] grey wolf pack consists of the following:

1. The alphas are leading the pack, the alpha wolves are responsible for making decisions. The alphas decisions are dictated to the pack.
2. The betas are subordinate wolves that help the alpha in decision making or other activities. The beta can be either male or female, and he/she is probably the best candidate to be the alpha.
3. The omega play the role of scapegoat. Omega wolves always have to submit to all the other dominant wolves. They are the last wolves that are allowed to eat.
4. The deltas have to submit alphas and betas, but they dominate the omega. *Scouts*, *sentinels*, *elders*, *hunters*, and *caretakers* belong to this category. *Scouts* are responsible for watching the boundaries of the territory and warning the pack in case of any danger. *Sentinels* protect and guarantee the safety of the pack. *Elders* are the experienced wolves who used to be alpha or beta. *Hunters* help the alphas and betas when hunting prey and providing food for the pack. Finally, the *caretakers* are responsible for caring for the weak, ill, and wounded wolves in the pack.

In the mathematical model for the GWO the fittest solution is called the alpha (α). The second and third best solutions are named beta (β) and delta (δ), respectively. The rest of the candidate solutions are assumed to be omega (ω). The hunting is guided by α, β, δ, and ω follow these three candidates.

In order for the pack to hunt a prey they first encircling it. In order to mathematically model encircling behavior, the following Eqs. (1)–(4) are used.

$$\vec{X}(t+1) = \vec{X}_p(t) + \vec{A}.\vec{D}, \tag{1}$$

where $\vec{D}$ is as defined in Eq. (2), t is the iteration number, $\vec{A}$ and $\vec{C}$ are coefficient vectors, $\vec{X}_p$ is the prey position, and $\vec{X}$ is the grey wolf position.

$$\vec{D} = |\vec{C}\cdot\vec{X}_p(t) - \vec{X}(t)|, \tag{2}$$

the $\vec{A}$, $\vec{C}$ vectors are calculated as in Eqs. (3) and (4).

$$\vec{A} = 2a\cdot\vec{r_1} - a \tag{3}$$

$$\vec{C} = 2\vec{r_2}, \tag{4}$$

where a is linearly decreased from 2 to 0 over the course of iterations, and r_1, r_2 are random vectors in [0, 1]. The hunt is usually guided by the alpha. The beta and delta might also participate in hunting occasionally. In order to mathematically simulate the hunting behavior of grey wolves, the alpha (best candidate solution), beta (the second best candidate solution), and delta (the third best candidate solution) are assumed to have better knowledge about the potential location of prey. The first three best candidate solutions obtained so far and oblige the other search agents (including the omegas) to update their positions according to the position of the best search agents. So the updating for the wolves positions is as in Eq. (5).

$$\vec{X}(t+1) = \frac{\vec{X_1} + \vec{X_2} + \vec{X_3}}{3}, \tag{5}$$

where $\vec{X_1}$, $\vec{X_2}$, $\vec{X_3}$ are defined as in Eqs. (6)–(8), respectively.

$$\vec{X_1} = |\vec{X_\alpha} - \vec{A_1}\cdot\vec{D_\alpha}|, \tag{6}$$

$$\vec{X_2} = |\vec{X_\beta} - \vec{A_2}\cdot\vec{D_\beta}|, \tag{7}$$

$$\vec{X_3} = |\vec{X_\delta} - \vec{A_3}\cdot\vec{D_\delta}|, \tag{8}$$

where $\vec{X_\alpha}$, $\vec{X_\beta}$, $\vec{X_\delta}$ are the first three best solutions in the swarm at a given iteration t, $\vec{A_1}$, $\vec{A_2}$, $\vec{A_3}$ are defined as in Eq. (3), and $\vec{D_\alpha}$, $\vec{D_\beta}$, $\vec{D_\gamma}$ are defined using Eqs. (9)–(11), respectively.

$$\vec{D_\alpha} = |\vec{C_1}\cdot\vec{X_\alpha} - \vec{X}|, \tag{9}$$