Expert Systems with Applications 38 (2011) 2412–2420

Contents lists available at ScienceDirect

Expert Systems with Applications

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

Evolving boundary detector for anomaly detection

Wang Dawei [*], Zhang Fengbin, Xi Liang

Department of Computer Science and Technology, Harbin University of Science and Technology, P.O. Box 258, 52 Xuefu Road Nangang District, Harbin City 150080, Heilongjiang Province, PR China

ARTICLE INFO

Keywords:
Artificial immune systems
Anomaly detection
Real-valued negative selection
Evolutionary search
Hole
Deceiving anomaly

ABSTRACT

In real-valued negative selection algorithm, the variability of self sample would result in the holes on the boundary between the self and non-self region and the deceiving anomalies hidden in the self region. This paper analyzes the reason for the difficulty in handling these problems by traditional evolved detectors, and then proposes a method of evolving boundary detectors to solve them. This method uses an improved detector generation algorithm based on evolutionary search to generate boundary detectors. The boundary detectors constructed by an aggressive interpretation are allowed to cover a part of self region. The aggressiveness controlled by boundary threshold can convert some volume of self sample into the fitness of boundary detector. This makes them enable to eliminate the holes on the boundary and have an opportunity to detect the deceiving anomalies hidden in the self region. Experiments are carried out using both 2-dimensional dataset and real world dataset. The former was designed to demonstrate intuitively that boundary detectors can cover the holes on the boundary, while the latter was to show that boundary detectors can detect the deceiving anomalies.

1. Introduction

The anomaly detection problem can be stated as a two-class problem: given an element of the space, classify it as *normal* or *abnormal* (Patcha & Park, 2007). A very common approach of anomaly detection is to specify a range of variability for each parameter of the system, and if the parameter is out of a range, it is considered to be an abnormality (Lane & Brodley, 1999). There exist many approaches for anomaly detection which include statistical (Denning, 1987), machine learning (Chan & Lippmann, 2006), data mining (Costantina et al., 2007). Unfortunately, these proposed techniques might have difficulty in many anomaly detection applications, because abnormal samples are not available at the training stage. The immunological inspired techniques have been successfully to perform anomaly detection (Forrest, Perelson, Allen, & Cherukuri, 1994; Hofmeyr & Forrest, 2000; Simon & He, 2008). The task of anomaly detection may be considered as analogous to the immunity of natural systems, while both of them aim to detect the abnormal behaviors of system that violate the established policy (Boukerche, Machado, & Juca, 2007; Dasgupta & Gonzalez, 2002).

Artificial immune systems (AIS) is a relatively new field that tries to exploit the mechanisms present in the biological immune system (BIS) in order to solve computational problems (Gonzalez, Dasgupta, & Gomez, 2003). The vast majority of developments within AIS focused on three main immunological theories: clonal selection, immune network and negative selection. They can roughly be classified into two major categories: techniques inspired by the self/non-self recognition mechanism and those inspired by the immune network theory (Gonzalez et al., 2003). The negative selection algorithm (NSA) was proposed by Forrest and her group (Forrest et al., 1994). This algorithm is inspired by the mechanism of T-cell maturation and self tolerance in the immune system, and believed to have distinct process from alternative methods and be able to provide unique results with better quality (Garrett, 2005). Different variations of NSA have been used to solve problems of anomaly detection, fault detection, to detect novelties in time series, and even for function optimization (Zhou & Dasgupta, 2007).

The two major data representations of NSA are (low-level) binary representation and (high-level) real-valued representation. Most works in NSA used the problem in binary representation (Esponda, Forrest, & Helman, 2004). Binary representation provides a finite problem space that is easier to analyze, and straightforward to use for categorized data. However, NSA in binary representation can hardly process many applications that are natural to be described in real-valued space (Zhou & Dasgupta, 2007), and generates a higher false alarm rate when applied to anomaly detection for some data sets (Dasgupta, Yu, & Majumdar, 2003). Gonzalez, Dasgupta, and Kozma (2002) introduced a real-valued representation, called real-valued negative selection (RNS) algorithm to alleviate the scaling issues of binary representation. Real-valued

[*] Corresponding author. Tel.: +86 159 0461 2835.
E-mail addresses: stonetools1982@yahoo.com.cn (D. Wang), zhangfb@hrbust.edu.cn (F. Zhang), xljyp@yahoo.com.cn (L. Xi).

0957-4174/$ - see front matter © 2010 Elsevier Ltd. All rights reserved.
doi:10.1016/j.eswa.2010.08.030

representation provides some advantages such as increased expressiveness, the possibility of extracting high-level knowledge from the generated detectors, and, in some case, improved scalability (Gonzalez & Dasgupta, 2003). Zhou and Dasgupta (2004) proposed a RNS with variable-sized detectors (V-detector). V-detector uses variable-sized detectors and terminates training stage when enough coverage is achieved.

Detector generation with real-valued representation can employ either random search (Gonzalez et al., 2002; Gonzalez & Dasgupta, 2003; Zhou & Dasgupta, 2004) or evolutionary search (Dasgupta & Gonzalez, 2002). Random search is known as classical generation-and-elimination strategy. Only the qualified detectors that do not match the self are selected and used to detect abnormal behavior of the new incoming data. Unfortunately, these randomly generated detectors cannot be guaranteed to cover the non-self region in the most efficient way. Evolutionary search used a genetic algorithm (GA) to generate detectors. The fitness function is based on the number of elements in the training set that belongs to the subspace represented by the detector and the volume of the subspace represented by the detector. A niching algorithm is applied to get the different detectors. This approach employs hypercube as the shape of detector and enables to cover the non-self region with fewer detectors.

In this paper, we show the issues in real-valued negative selection algorithm (RNS) caused by the variability of self sample, which include the holes on the boundary and the deceiving anomaly. However they can hardly be solved by the traditional detectors generated by evolutionary search. Then we propose a method which evolves aggressive boundary detectors to cover the non-self region. This approach improves the detector generation method shown in Dasgupta and Gonzalez (2002), while it employs hypersphere as the shape of detector and evaluates the fitness of detector via the actual covering volume. The aggressive boundary detectors can convert some volume of the self region on the boundary into the fitness of themselves. The increase of the fitness of the detectors on the boundary can decrease the holes and have a chance to detect the anomaly hidden in the self region.

The remaining sections of the paper are structured as follows: Section 2 shows improved detector generation algorithm based on evolutionary search. This is then followed in Section 3 with the problem with the issues in real-valued negative selection algorithm caused by the variability of self sample. In Section 4 we introduce the aggressive boundary detector. We carry out the experiment on synthetic data and Fisher's Iris data in Section 5. Finally, some concluding remarks are given in Section 6.

2. Detector generation algorithm based on evolutionary search

Detector generation based on evolutionary search shown in Dasgupta and Gonzalez (2002) uses hypercube as the shape of detector. In this work, we employ hypersphere which is the same shape of self sample as the shape of detector. The hypersphere detectors are generated using evolutionary search driven by two main goals (Gonzalez & Dasgupta, 2003):

1. Move the detector away from the self samples.
2. Maximize the covering of non-self region and minimize the overlap among the detectors.

Fig. 1 illustrates the flow chat of detector generation based on evolutionary search, where

coverage current volume of non-self region covered by detectors
desiredCoverage desired volume of non-self region covered by detectors
minFitness minimum fitness allowed for a detector to be include in the mature detector set
maxAttempts maximum number of attempts to try to evolve a detector with a fitness greater than *minFitness*

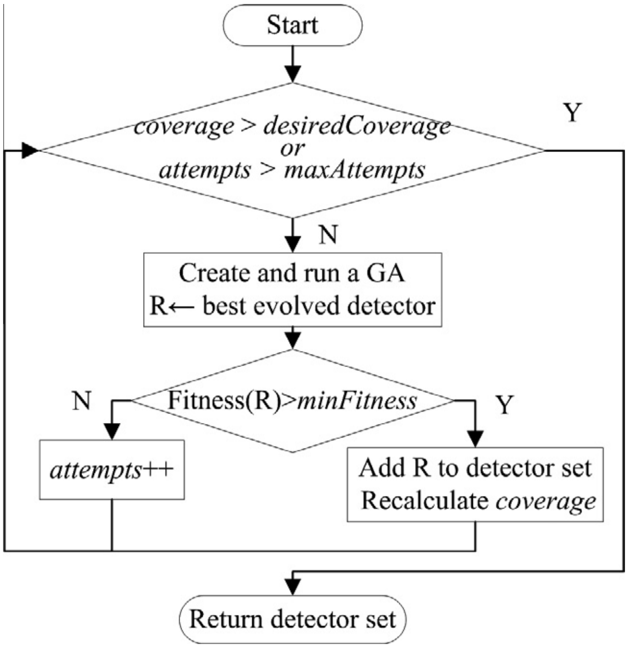

Fig. 1. Detector generation using evolutionary search.

The initial population of algorithm is a set of randomly generated candidate detectors $D_{\text{candidate}} = \{d^1, d^2, \ldots, d^m\}$, where d^j denotes a hypersphere detector whose center is a n-dimensional point $(d^j_1, d^j_2, \ldots, d^j_n)$ and radius is r^j_d. After crossover and mutation operations, the candidate detector which is the best evolved one in the population will be added into the mature detector set, if its fitness is greater than *minFitness*; otherwise, *attempts* which denotes the number of failing to evolve a detector with fitness greater than *minFitness* will increase. The convergence condition of algorithm is that either *coverage* is greater than *desiredCoverage* or *attempts* is greater than *maxAttempts*.

The fitness calculation is to evaluate the quality of a candidate in a population. Based on their fitness, the candidate may be selected for crossover and mutation operations or even becoming a mature detector. In this work, the fitness of a detector is evaluated by the actual volume of the non-self region covered by the detector. The fitness function is described by the following equation:

$$fitness(d) = \begin{cases} volume(d) - overlap(d) & \text{if } coverage(d, S) = 0 \\ -1 & \text{if } coverage(d, S) > 0 \end{cases} \quad (1)$$

where d denotes the detector whose fitness need to be evaluated and S is self sample set.

This fitness function is guided by the two main goals mentioned in the previous section. The fitness of d is equal to -1, if d covers self region. Such a determination is in accord with the principle of NSA. The fitness can be divided into two parts, if d does not cover any self samples. A detector is rewarded for covering the more non-self region, while it will be penalized if it overlaps with other mature detectors. Then we estimate $volume(d)$ which denotes the volume of d inside the shape space and $overlap(d)$ which is the volume overlapped with other mature detectors to evaluate the fitness of d using Monte Carlo method.

3. Issues in evolving detectors

AIS has been successfully to perform many anomaly detection application in which other techniques such as statistical and