Tao Yang, Wen Chen*, and Tao Li

A Real Negative Selection Algorithm with Evolutionary Preference for anomaly detection

DOI 10.1515/phys-2017-0013
Received November 8, 2016; accepted November 16, 2016

Abstract: Traditional real negative selection algorithms (RNSAs) adopt the estimated coverage (c_0) as the algorithm termination threshold, and generate detectors randomly. With increasing dimensions, the data samples could reside in the low-dimensional subspace, so that the traditional detectors cannot effectively distinguish these samples. Furthermore, in high-dimensional feature space, c_0 cannot exactly reflect the detectors set coverage rate for the nonself space, and it could lead the algorithm to be terminated unexpectedly when the number of detectors is insufficient. These shortcomings make the traditional RNSAs to perform poorly in high-dimensional feature space. Based upon "evolutionary preference" theory in immunology, this paper presents a real negative selection algorithm with evolutionary preference (RNSAP). RNSAP utilizes the "unknown nonself space", "low-dimensional target subspace" and "known nonself feature" as the evolutionary preference to guide the generation of detectors, thus ensuring the detectors can cover the nonself space more effectively. Besides, RNSAP uses redundancy to replace c_0 as the termination threshold, in this way RNSAP can generate adequate detectors under a proper convergence rate. The theoretical analysis and experimental result demonstrate that, compared to the classical RNSA (V-detector), RNSAP can achieve a higher detection rate, but with less detectors and computing cost.

Keywords: artificial immune, negative selection, subspace, anomaly detection

PACS: 05.20.-y

1 Introduction

Biological Immune System (BIS) can distinguish between self-organization and harmful antigens, and eliminate harmful antigens to ensure biology in health. Inspiration by BIS, formed the research field of Artificial Immune System (AIS) which has attracted more and more researchers to develop many algorithms. In AIS, the Negative Selection Algorithm (NSA) is an important detector generating algorithm which was first proposed by Forrest et al. [1]. The NSA simulates the T cells censoring process in the thymus to generate mature detectors without immune Self-Reaction. It has shown to be efficient for anomaly detection [2, 3], data classification and fault diagnosis [4, 5].

Early negative selection algorithms defined the antibody (detector) and antigen (abnormal data) in binary representation, and used R-Continue-Bits Match Rule to calculate affinity of antibody and antigen [1]. On account of the fact that many applications are natural to be described in real-valued feature space, Gonzalez and Dasgupta proposed a Real Negative Selection Algorithm (RNSA) [6], in which the data samples (detectors and antigens) are normalized into the real-valued feature space $[0, 1]^n$ (n denotes the number of samples dimension), and the affinity is calculated by the Minkowski distance. Some modified versions of RNSA have been proposed, Ji and Dasgupta proposed Real Negative Selection Algorithm with variable detector radius (V-detector) [7], in which the detector radius was dynamically resized to the nearest self-margin; Gong presented a further training method to reduce the computational expense by reducing the self samples [8]; Chen improved the detectors generation efficiency by adopting the hierarchical clustering preprocess of self set [9]; Poggiolini implicated the feature detection rule to RNSA and improved the algorithm performance [10].

The major challenge of NSA is to efficiently generate effective detectors. Traditional RNSA generates detectors randomly, until the Estimated Coverage (c_0) reaches the threshold. In high-dimensional feature space, the distribution of samples is extremely sparse and non-uniform, in that case c_0 could not exactly reflect the coverage for nonself samples, leading the algorithms converge so fast that it is terminated unexpectedly while there only a few detectors generated. Besides, in high-dimension, a large

Tao Yang: College of Computer Science, Sichuan University, China, E-mail: yangtao_cwnu@163.com
***Corresponding Author: Wen Chen:** College of Cybersecurity, Sichuan University, China, E-mail: wenchen@scu.edu.cn
Tao Li: College of Cybersecurity, Sichuan University, China, E-mail: wenchen@scu.edu.cn

number of data samples could "fall" into the subspace, the conventional detectors cannot discriminate these samples without appropriate guidance. These shortcomings cause the traditional negative selection algorithm to perform poorly in high-dimensions feature space, and restrict the application of artificial immune theory in anomaly detection.

The evolution of the immune cell has the evolutionary preference to capture the pathogen antigen. Following from this, this paper proposes a real negative selection algorithm with evolutionary preference (RNSAP) which uses a novel termination condition and a new detector training strategy. First, for guaranteeing the detectors can cover more nonself space and reduce the redundant detectors, RNSAP adds the preference of "cover unknown nonself space" to the conventional detectors by applying redundancy testing, and then uses the redundancy (R) as an algorithm termination threshold to ensure the algorithm is convergent under a prober rate. Second, to effectively cover the low-dimensional subspace where samples might gather in, RNSAP will calculate out the "low-dimensional target subspace" and utilize it as the "spatial preference" to generate the detectors with spatial preference. Lastly, for excluding the "hole" as much as possible, RNSAP will use the "known nonself sample" as "feature preference" to train detectors with feature preference. The theoretical analysis and experimental results suggest: compared to a classical real negative selection algorithm (V-detector) on low-dimension dataset (Haberman's Survival Data Set), RNSAP has a higher detection rate, with fewer detectors and shorter training time; on high-dimension dataset (KDD CUP99), the performance of traditional NSA is very poor while RNSAP performs well.

The rest of this paper is organized as follows: In Section 2, some basic definition are covered; In Section 3 RNSAP is introduced in detail; Experiment results are shown and discussed in Section 4 while Concluding remarks are given in Section 5.

2 Basic definition

The immune system relies on antibody cell discrimination "self" and "nonself" exclude antigenicity, is human's primacy defense against pathogenic organisms and cells. In an Artificial Immune System, data samples are defined as "antigen", normal samples are defined as "self", abnormal samples are defined as "nonself", and antibody is defined as "detector". For faciliting the description, define the basic conceptions of RNSAP as follows:

Definition 1 (antigen): All the character strings abstracted from the feature space constitute the antigen set $U = \{g|(f_1, f_2, \cdots, f_n), f_i \in [0, 1]\}$, where n is the data dimension and f_i represents the $i\text{-}th$ normalized attribute value.

Definition 2 (self/nonself set): The self set $S \subset U$ is the character strings abstracted from normal samples, $r_s \in R^+$ is the variability threshold of the self sample; Nonself set $N = U - S$, which represents character strings abstracted from abnormal samples, and $S \bigcup N = U$, $S \bigcap N = \Phi$.

Definition 3 (detector): Detector $d = (c, r)$, where $c \in N$, c is the central vector of d in the feature space; $r \in R+$ is the detector radius. Antigens which are close to any detector less than r will be identified as nonself elements.

Definition 4 (self-reactive): If any self element located in the detection region of detector d, and then d is a self-reactive detector.

Definition 5 (self/nonself space): In feature space, the part of being covered by self samples is called self space, the rest of space is nonself space; in the nonself space, the part of being covered by detectors is called known nonself space while the remaining part is unknown nonself space.

3 The strategies of RNSAP

The main idea of RNSAP is using the *redundancy (R)* to replace the estimated coverage (c_0) as the termination threshold, and utilizes the "*unknown nonself space*", "*low-dimensional target subspace*" and "*known nonself feature*" as the evolutionary preference to guide the generation of detectors. In that way, RNSAP can generate more effective detectors with the proper rate of convergence.

3.1 The termination condition of detector generation in RNSAP

3.1.1 The influence of dimension on estimated coverage

In a traditional real negative selection algorithm, the coverage rate of detector set for nonself space (p) determines the performance of immune algorithm, and p can be expressed as:

$$p = \frac{V_{covered}}{V_{noself}} = \frac{\int_{covered} dx}{\int_{nonself} dy} \tag{1}$$

Because Eq.1 is difficult to calculate directly, paper [7] proposed a method to evaluate P by using "point estimate":

$$t \geq 1/(1 - c_0) \tag{2}$$

In Eq.2, t is the times of without finding an uncovered random point, and c_0 is the *Estimated Coverage*. If $c_0 = 80\%$, the algorithm will be terminated when continuous find 5 random sampling covered by the detectors set.

After normalisation, the n-dimensional feature space is represented as n-dimensional hypercube $u = [0, 1]^n$, the data samples are represented as the "points" and the detectors are represented the "hyper-sphere". The c_0 represents the coverage situation between "data sample" (point) and "detector" (hyper-sphere). In low-dimension space, if the training samples are distributed uniformly, c_0 can reflect the real coverage rate of detector set for non-self space. However, if training samples are non-uniformly distributed, the c_0 will no longer apply.

Comparison of Fig. 1(a) and Fig. 1(b) reveals that, Fig. 1(a) has 7 self samples (green ".") and 10 nonself samples (red "+") uniform distributed in 2-dimensional feature space, the 12 mature detectors cover almost whole 2-d space; While Fig. 1(b) only has 1 self sample and 1 nonself sample distributed in edge of space (sparse and non-uniform), the 1 big mature detector covers almost the whole 2-d space. Fig. 1 shows that the mature detectors have covered almost all feature space, so the Eq.2 would be easy to meet (assume $c_0 = 80\%$). However in Fig. 1(a) the real coverage rate on this training set is 90% and in Fig. 1(b) is 0.

In high-dimensional space the samples distribution is always sparsely and non-uniform [11]. Firstly, assuming in n-dimensional feature, the value range of data samples on $i\text{-}th$ dimension is $[0, n_i]$, and then the total amount of data (N_a) that n-dimensional feature space could accommodate is:

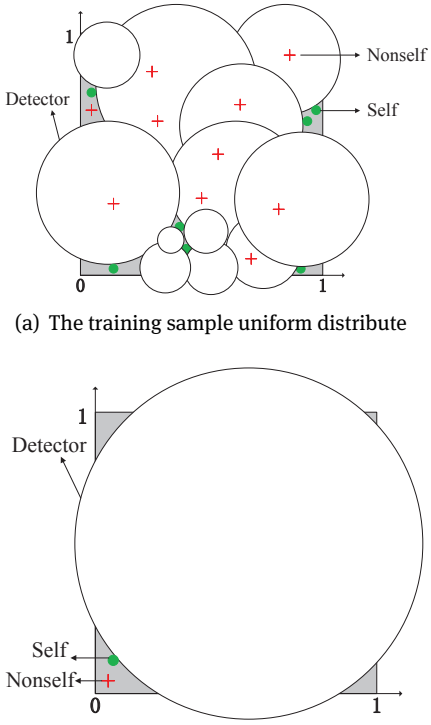
$$N_a = \prod_{i=1}^{n} n_i \tag{3}$$

From Eq.3, N_a grows exponentially with the n, and the number of training data (N_t) is limited by the data set and is customarily much smaller than N_a, so the training data distribution may extremely sparsely in high-dimensional feature space.

Secondly the unitary hypercube $u = [0, 1]^n$ has a total volume of 1, assume u_0 is one other n-dimensional hypercube inside u, then the volume of u_0 is:

$$V_{u0} = \prod_{i=1}^{n} a \tag{4}$$

where a is the side length of the cube u_0, $a \in [0, 1)$. From Eq.4, if $a = 0.9$ and $n = 40$, V_{u0} is approximately 0.015, that means the u_0 (the central part of u) only contains 1.5% samples (assuming a plenty of samples uniform distribute in the u and fill the whole space), and about 98.5% samples locate in the edge marginal area of u. In contrast, the



(a) The training sample uniform distribute

(b) The training sample non-uniform distribute

Figure 1: The influence of sample distribution on detectors generation

n-dimensional detectors (hyper-sphere) with a variable radius $r \in [0, \sqrt{n}]$ in high-dimensional feature space the single detector could have a huge volume and might cover almost whole central part of feature space. In that case, the condition (2) would be satisfied rapidly. It causes algorithm to be terminated unexpectedly while the amount of detectors is not enough. Hence the c_0 is not suited for high-dimensional space (detailed discussion in 4.4).

3.1.2 The redundancy and redundant testing

In fact, the coverage situation between "mature detector" (hyper-sphere) and "candidate detector" (hyper-sphere) is calculated more easily and it is not influenced by the dimension growth. If the most volume of a candidate detector (d_i) overlaps with a mature detector (d_m), the (d_i) could be considered as a "redundant detector". When there are too many redundant detectors in feature space, randomly generated detectors cannot guarantee that the new mature detectors cover the unknown nonself space. In that case, considering the efficiency and performance, the generation of detectors should been terminated. Thus RNSAP

adopts the *redundancy (R)* as the algorithm termination threshold.

For calculating redundancy R, RNSAP divide the "Redundant-Judgment Zone" by using detector radius (r), detector central vector (c) and distant parameters. As shown in Fig. 2(a), the gray zone is the "redundant-judgment zone" which depends on d_m=<c_m, r_m> (the mature detectors), s_m (the most vicinity self-sample of d_m) and a variable parameter $R_c \in [0.1]$. If a new candidate detector d_n=<c_n, r_n> locates in the redundant-judgment zone (d_n satisfy with expression 5), d_n should be judged as a redundant detector and be removed, and the redundant count will be accumulated:

$$(l_{mn} \le r_m * R_c) \bigcap l_{ns} \le r_m \tag{5}$$

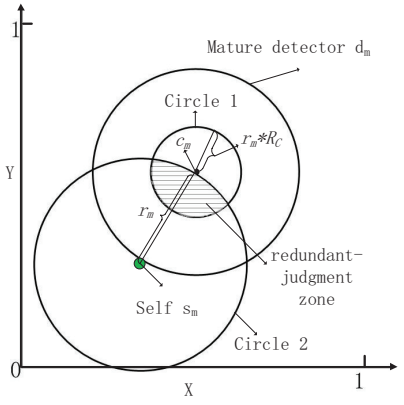
In expression 5, ln_s is the Euclid distant between c_n and s_m , l_{mn} is the Euclid distant between c_n and c_m. $R_c \in [0, 1]$ is variable parameter, the size of redundant-judgment zone is proportional to R_c. If R_c is close to 0, the size of redundant-judgment zone is smaller, the probability of d_n being judged as a redundant detector is lower, the feature space could accommodate more detectors. If R_c is close to 1 the size of redundant-judgment zone is larger, and the probability of d_n being judged as a redundant detector is higher, the feature space could accommodate less detectors.

Fig. 2(b), Fig. 2(c) and Fig. 2(d) show the process of redundancy testing. The testing uses the same d_m and the redundant-judgment zone (gray zone). In Fig. 2(b) candidate detector d_{n1} locates in the redundant-judgment zone, its volume almost overlapping of mature detector d_m, so d_{n1} should be judged as a redundant detector; In Fig. 2(c) and Fig. 2(d), d_{n2} and d_{n3} do not locate in redundant-judgment zone, their volume is just partially covered by d_m, compare to d_{n1} they can cover more known nonself space, thus d_{n2} and d_{n3} will become the new mature detector.

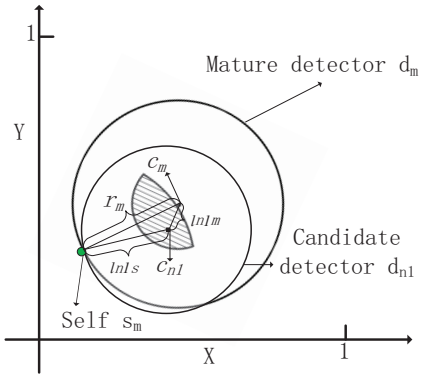
In each round of redundancy testing, if a candidate detector is judged as a redundant detector, the RNSAP will accumulate the "redundant count" (cnt), then calculate the ratio (redundancy R) of cnt and amount of mature detectors (N_m):

$$R = cnt/N_m \tag{6}$$

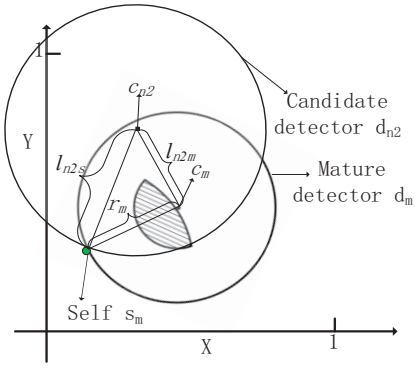
If R reaches the threshold, it means in the last round there were too many redundant detectors. In this situation, obtaining new high quantity (ir-redundant) detectors becomes more difficult, considering the efficiency and performance, RNSAP will terminate the random generation of detectors.



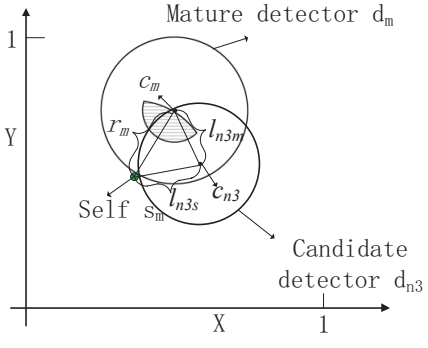
(a) The division of redundant-judgment zone



(b) Redundant testing-1



(c) Redundant testing-2



(d) Redundant testing-3

Figure 2: Redundant-judgment zone and redundant testing

Table 1: The generation of conventional detectors

1.Initialization,cnt=0.
2.Randomly generation a candidate detector d_n.
3.If d_n is a redundant detector, cnt=cnt+1;else goto 5.
4.If redundant reaches the threshold, end;else goto 2.
5.If d_n,matched any self sample, goto 2.
6.Record d_n as a new mature detector,cnt=0,goto 2.

Different from estimate coverage (c_0), redundant testing calculates the cover situation between "mature detector" (hyper-sphere) and "candidate detector" (hyper-sphere). It has nothing to do with data distribution, so it is available in high-dimensional space. There are two obvious advantages in using the redundancy (R) as the termination threshold: 1. it can ensure the algorithm is convergent under a proper rate; 2. by using condition (5) RNSAP adds a preference of "cover unknown nonself space" to the randomly generation process of detectors, therefore the redundant testing not only removes the redundant detectors but also improves the quality of detectors.

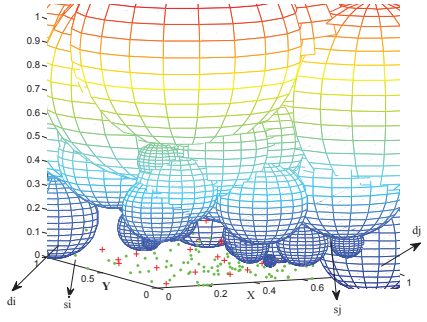
RNSAP calls the detectors which are randomly generated as a conventional detector, the conventional detectors generation algorithm by using redundant testing as shown in Table 1 .

3.2 The detectors with evolutionary preference

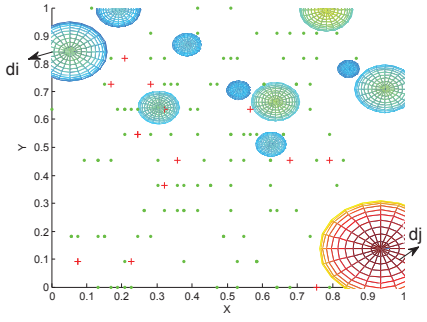
Despite that conventional detectors have been highly redundant in feature space, there might be still some "uncovered nonself space" left. Completely random generation of detectors to cover the uncovered nonself space will cost too many of computing resources. Perelson proposed the Immune Repertoire Model and pointed out that "not all receptor shapes (detectors) need to be made at random" [12]. The latest immunology research results show that: The evolution of Immune cell is not entirely a random process, but rather has an evolutionary preference to capture the pathogen antigen [13]. Based on these theories, RNSAP utilizes the "low-dimensional target subspace" and "known nonself feature" as the evolutionary preference to guide the generation of detectors. The detectors with preference can effectively cover reducing the uncovered nonself space and improve the performance of algorithm significantly.

3.2.1 The detectors with spatial preference

In high-dimension feature space, the training samples would fall into the low-dimension subspace. As shown in Fig. 3(a)(Haberman's Survival dataset), in the 3-dimension (XYZ) feature space, the green '.' represents self sample, the red '+' represents nonself sample, the 3D-sphere represents mature conventional detector generated by V-detector algorithm, (r_s = 0.01, c_0 = 90%). There are many samples falling into the "XY plane", however only a few mature detectors (such as d_i and d_j) intersect with the XY plane. The Fig. 3(b) shows the coverage of mature detectors for the XY subspace from XY-perspective, the conventional detectors are almost not cover the nonself samples in XY subspace. The most effective way to distinguish



(a) The convention detectors(XYZ perspective)



(b) The convention detectors(XY perspective)

Figure 3: The convention detectors (V-detector)

these samples which fall into the subspace is to generate detectors in the aimed subspace, in other words the central vector (c) of candidate detector should locate in the aimed subspace directly. When the one dimension of data get 0 value, the data would fall into the corresponding subspace, assuming P_{i_0} represents the probability of the i_{th} dimension of c gets 0 value, then the probability ($P_{j\ldots k}$) of the any candidate detectors' central vector fall into the

subspace $S_{j\ldots k}$ is:

$$P_{j\ldots k} = \prod_{i=j}^{k} (1 - P_{i\_0}) \prod_{i=1, i\neq j\ldots k}^{n} P_{i\_0} \tag{7}$$

From the Eq. 7, $P_{j\ldots k}$ becomes lower when n growth, so in the high-dimension space it is almost impossible for candidate detectors to cover the subspace by random generation. To deal with this situation, RNSAP analyzes the distribution of the training sample first; and then find out all the "target subspaces" which have high density of samples; at last use "target subspaces" as spatial preference to guide the generation of the detector.

For a single dimension, the density of data distribution can be described by a Jini value [12]. The Jini value of dimension A can be calculated by Eq. 8:

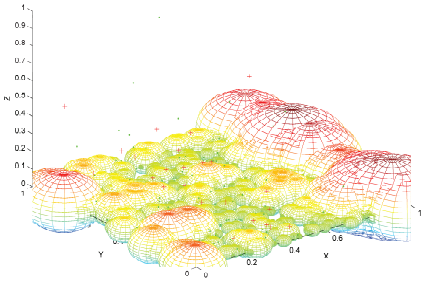
$$Jini(A) = 1 - \sum_{j=1}^{\omega} {p_i}^2 \tag{8}$$

Where ω is the number of equal interval (the "0" is treated a single interval) which is divided in the dimension A, p_i is the proportion of the number of samples which located in the i_{th} interval to the total number. From Eq.8, the smaller Jini value indicates that the distribution of samples is more densely in the dimension A while the larger Jini value indicates the distribution is more dispersed. By calculating the Jini value and presetting the Jini threshold ξ, RNSAP can select out all the dimension in which the distribution of samples is dense. And then, RNSAP will calculate out the cluster center x in each dense dimension. At last, the x will be used to edit the central vector c.

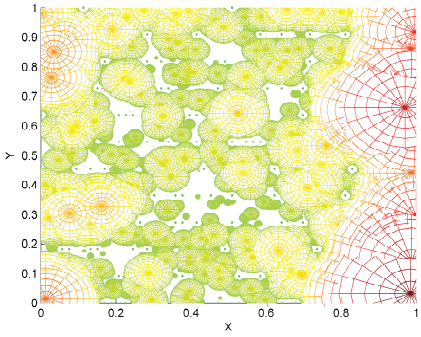
$$c_i = \begin{cases} 0, Jini(A) < \xi \quad and \quad x = 0 \\ x, Jini(A) < \xi \quad and \quad x \neq 0 \\ random[0, 1], \quad Jini(A) > \xi \end{cases} \tag{9}$$

In expression 9, c_i denotes the i_{th} dimension value of central vector c. RNSAP according three cases to set the value of c_i: (1) If dimension A is a dense dimension and samples in A gather in "0", RNSAP sets the $c_i = 0$ for that the candidate detectors will be generated in the low-dimension subspace. (2) If dimension A is a dense dimension and samples gather in x ($x \neq 0$), c_i will take a random value between x-θ and x+θ, for that the candidate detectors will be generated in the region where samples distribute dense. (3) If dimension A is not a densely dimension, A cannot provide any spatial information to guide the generation of detector, c_i will take a random value between [0, 1] as conventional algorithm.

By using expression 9, RNSAP can generate the detector with spatial preference in the target subspace accurately. The detectors with spatial preference are shown in Fig. 4(a) and Fig. 4(b) (for the convenience of observation, all the conventional detectors are not shown). Compare Fig. 3(b) with Fig. 4(b), in XY subspace conventional detectors could barely recognize nonself samples while by training the detectors with spatial preference RNSAP almost covered all nonself samples. It is worthy to notice that: different from the dimension reduction, RNSAP just guides the detectors generated in the target subspace without changing the dimension of the feature space. The algorithm of training detectors with spatial preference is shown in Table 2.



(a) The detectors with spatial preference(XYZ perspective)



(b) The detectors with spatial preference(XY perspective)

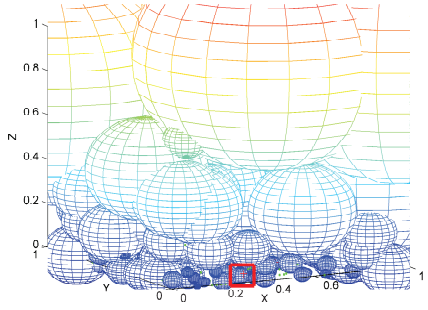
Figure 4: The detectors with spatial preference

Table 2: The algorithm of training detector with spatial preference

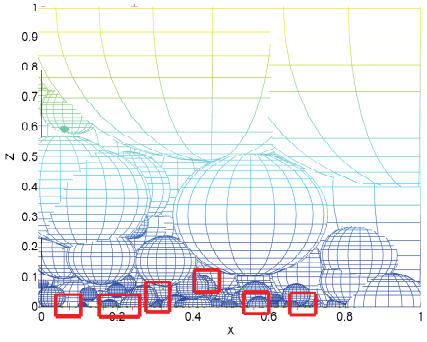
1.According the Eq.8 to calculate out all the densely dimensions .
2.Calculate the x on each densely dimension.
3.set cnt =0, randomly generation a candidate detector $d_n(c_n,r_n)$.
4.According the expression 9 to edit c_n.
5.If d_n is a redundant detector, cnt=cnt+1;else goto 7.
6.In the subspace ,if redundant reaches the threshold, end;else goto 3.
7.If d_n,matched any self sample, goto 3.
8.Calculate the rn and record d_n,cnt=0,goto 3.

3.2.2 The detectors with feature preference

After adding the detector with spatial preference, the mature detectors can almost cover the feature space, however the "holes" still cannot be voided. The holes are tiny gaps of feature space which is not covered by the detectors. To eliminate the holes, a complete training sample set is needed with huge time resource and space resources. In Fig. 5(a) and Fig. 5(b) (Haberman's Survival dataset), the boxes are the "holes" which contain the nonself (red '+').



(a) The "holes" (XYZ perspective)



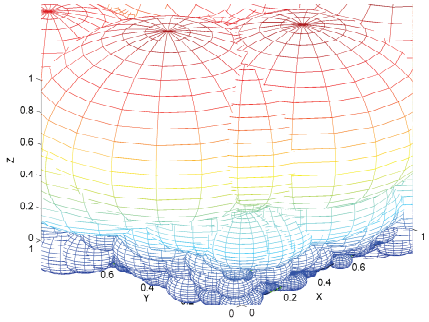
(b) The "holes"(XY perspective)

Figure 5: The "holes"

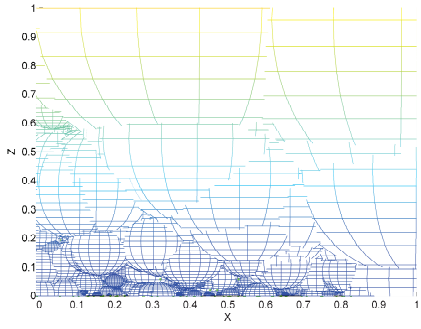
The traditional real negative selection algorithm trains detectors by only using one class samples (self samples), however the abnormal data (nonself samples) are easy to collect in the real practical application of anomaly detection. Similar to "vaccination" in medicine, RNSAP trains the detectors with feature preference by analyzing the "known nonself" samples. Assuming ns_i is a known nonself sample, RNSAP will use the ns_i to test the current mature detector set. If ns_i has been covered by one of the mature detectors, it means current detectors can recognize this abnormal data. Conversely, if ns_i is not covered by any mature detectors, it means the ns_i falls into the hole and this hole might cause the "False negative". In that case, RNSAP will set the feature vector of ns_i as the central vector and calculate the radius to generate detector with preference. Fig. 6(a) and Fig. 6(b)(Haberman's Survival dataset) show the final performance, after combating the conventional detectors, detectors with spatial preference and detector with feature preference. Table 3 shows the algorithm of training detectors with feature preference:

Table 3: The algorithm of training detectors with feature preference

1.If the number of noself training set N_{nt} >0,goto 2;,else end.
2.Pick up a ns_i as the new candidate detector,„N_{nt}=N_{nt}-1.
3.If ns_i covered by any mature detector, goto 1.
4.Caculate the radius r
5.Record the d(ns_i,r) as the detector with feature preference,goto 1.



(a) The final performance of RNSAP (XYZ perspective)



(b) The final performance of RNSAP (XY perspective)

Figure 6: The final performance of RNSAP

4 Experiment and discussion

4.1 Experiment setup

The V-detector algorithm is the latest version of RNSA and has shown excellent classification performance in

previous work [14, 15]. In this section the comparison of V-detector and RNSAP is carried out on a 3-dimension dataset (Haberman's Survival) and 41-dimension dataset (KDD CUP99) which are widely used for testing anomaly detection system. The experiments were repeated 100 times on each dataset and the average value were adopted.

(1)Dataset

Haberman's Survival dataset contains cases from study conducted on the survival of patients who had undergone surgery for breast cancer. This dataset contains 306 records, each records contains 3 continuous fields and 1 class label [16].

KDD CUP99 dataset is consists of real world network traffic data, where each record contains 38 continuous and 3 symbolic fields and 1 class label. The complete KDD CUP99 dataset contains 3925650 abnormal sample (80.14%) and 972780 normal sample (19.86%), where the abnormal sample are partitioned in 4 categories: DOS(about 98.92%), probing(about 1.05%), U2R(about 0.0013%),R2L(about 0.0286%) [17].

(2)Measurement criterion

This paper adopted detection rate (DR), false alarm rate (FA), amount of detectors (M), training time (Ttrain) and testing time (Ttest) to measure the performance of the algorithm. The DR and FA calculate as follows:

$$DR = TP/(TP + FN) \tag{10}$$

$$FA = FP/(FP + TN) \tag{11}$$

In Eq.10 and 11 , if the anomalous sample is classified as the nonself, it is counted as a true positive (TP), if it is classified as the self, it is counted as a false negative (FN); if the normal sample is classified as the self, it is counted as a true negative (TN), if it is classified as the nonself, it is counted as a false positive (FP).

(3)The levels of RNSAP

The complete RNSAP would train three kinds of detectors: 1) The detectors with spatial preference: these detectors are trained by using the subspace information of training samples, so that could cover the subspace more effectively; 2) The detectors with feature preference: these detectors are trained by "known nonself samples", it useful to eliminate the holes; 3) the conventional detectors: these detectors are trained randomly without any other information. In order to show the performance of the RNSAP in detail, according the training process the algorithm is divided into 3 levels:

RNSAP-1: using redundancy (R) as the algorithm termination threshold, only training the convention detectors.

RNSAP-2: using redundancy (R) as the algorithm termination threshold, training the convention detectors and the detectors with spatial preference.

RNSAP-3(complete RNSAP): using redundancy (R) as the algorithm termination threshold, training 3 kinds of detectors.

Table 4: the experiment setting of R_c

Dataset	Training set	Testing set	r_s
Haberman's Survival	*100%normal*	*50%anomalous*	*0.01*
KDDCUP99	*50% normal*	*50%anomalous*	*0.003*

4.2 Parameters setting

(1)The radius of self sample (r_s)

The r_s is an important parameter in any negative selection algorithm, the smaller r_s could cause false positive results while the larger r_s could cause the false negative results. Many previous works have been studied r_s in detail [7] [15] [18], so in this work it is not discussed. According to Eq.12 this paper calculated the minimum distant (d_{min}) between self sample and nonself sample on Haberman's Survival Data Set and KDDCUP99. After being normalized, the d_{min} = 0.018 in Haberman's Survival Data Set, and d_{min} = 0.0056 in KDD CUP99. To equilibrate the false positive and false negative, in the following experiments the r_s =0.01 on the Haberman's Survival Data Set dataset, and r_s=0.003 on KDD CUP99 dataset.

$$d_{min} = min\left(dis\left(s_i, ns_j\right)\right) \tag{12}$$

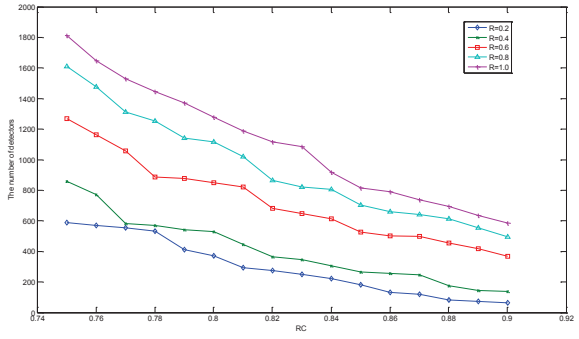
In Eq.12, i∈[1,size of selfset], j∈[1,size of nonselfset], dis(s_i, ns_j)represents the Euclid distance between self s_i and nonself ns_j).

(2)The "Redundant-Judgment Zone" parameter (R_c)

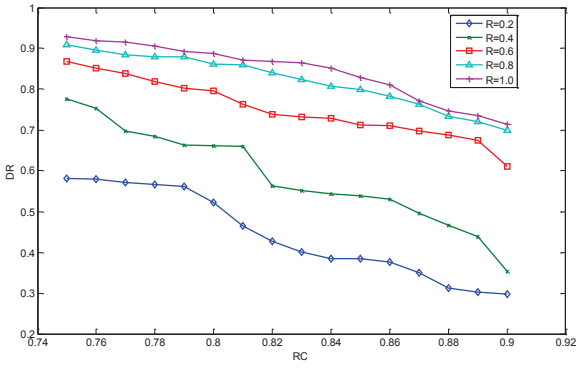
As discussed in section 3.1, the R_c determines the size of "Redundant-Judgment Zone". Under the same experimental condition, the smaller R_c means the algorithm could generate more detectors, while the larger R_c indicates fewer detectors. Fig. 7 shows the influence of R_c on RNSAP-1 in 3-dimensional feature space (Haberman's Survival dataset), while Fig. 8 shows the influence in 41-dimension feature space (KDDCUP99 dataset). In these experiment the redundancy (R) is set from 0.2 to 1 step by 0.2 and other experiment setting as shown in Table 4.

As shown in Fig. 7 (experiment on 3D dataset), under the same redundancy (R), for R_c from 0.9 to 0.75, the number of detectors increase about 200%(Fig. 7(a)). When R>=0.6 and R_c <=0.8, The increasing detectors improve the DR only less than about 7% (Fig. 7(b)), but improve the training time by more than 100% (Fig. 7(c)). The Fig. 8 (experiment on 41D dataset) reflects the similar situation, when R>=0.6 and R_c <=0.62, the number of detectors in-

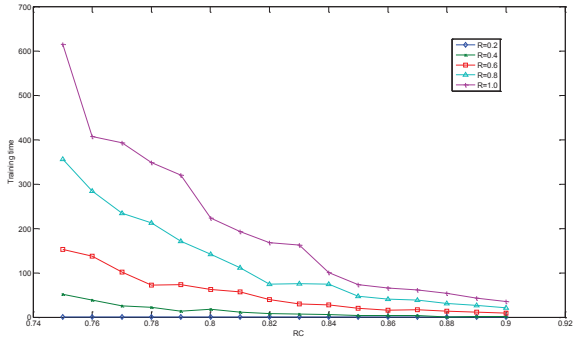
crease 300% (Fig. 8(a)). DR increase only less than 5% (Fig. 8(b)), while the training time increases more than 150% (Fig. 8(c)). Therefore, in order to account for algorithm performance and training cost, in following experiment R_c=0.8 on the Haberman's Survival Data Set dataset, and R_c=0.62 on KDD CUP99 dataset



(a) The number of detectors(3D dataset)



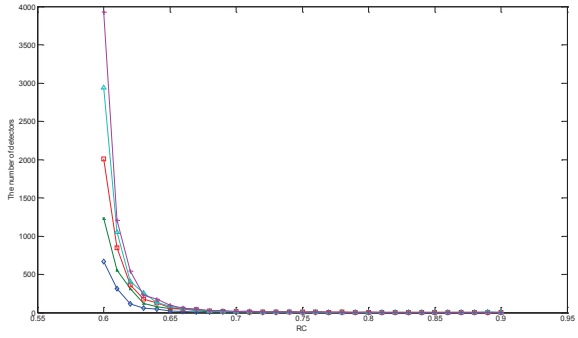
(b) The detection rate(3D dataset)



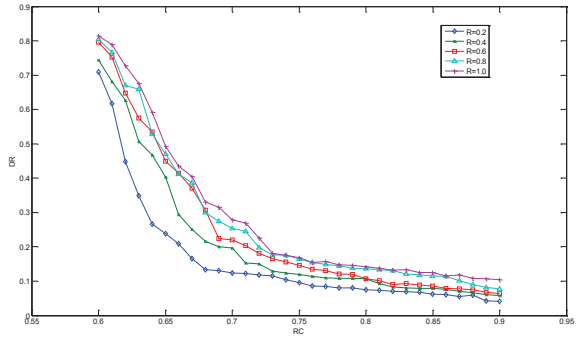
(c) The training time(3D dataset)

Figure 7: The influence of R_c on RNSAP-1(3D dateset)

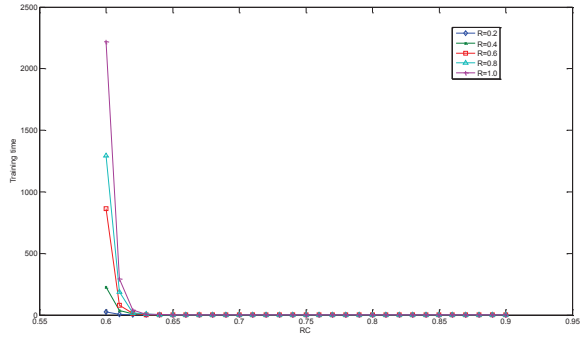
In Fig. 7 and Fig. 8, when R_c=0.8 on the Haberman's Survival Data Set dataset, and R_c=0.62 on KDD CUP99 dataset, RNASP might not get the highest DR, but the training cost is acceptable. Base on the low training cost, RNASP would enhance the performance by generating the



(a) The number of detectors (41D dateset)



(b) The detection rate (41D dateset)



(c) The training time (41D dateset)

Figure 8: The influence of R_c on RNSAP-1(41D dateset)

detectors with evolutionary preference (detail experimentation in 4.3 and 4.4).

4.3 Experiments on Haberman's Survival dataset

In this section, the comparison of V-detector and RNSAP is carried out on 3-dimension dataset (Haberman's Survival). The experiment is divided into 2 parts: (1) The comparison of V-detector and RNSAP-1, the experiment setting is shown in Table 5. The experiment result is shown in Table 6 and Table 7; (2)The comparison of RNSAP-1, RNSAP-2 and

RNSAP-3, the experiment setting is shown in Table 8 and the experiment result is shown in Fig. 9. In these tables and figures c_0 is the estimated coverage, R is the redundancy, DR is Detection rate, NOD is number of detectors, Ttrain is training time and Ttest is the testing time.

Table 5: Experiment setting of V-detector and RNSAP-1 on Haberman's Survival dataset

Algorithm	Training set	Testing set	r_s
V-detector	*100%normal*	*randomly 50%anomalous*	*0.01*
RNSAP-1	*100% normal*	*randomly 50%anomalous*	*0.01*

Table 6: The performance of V-detector

c_0	DR	NOD	Ttrain	Ttest
0.8	12.08%	24	0.03	0.018
0.85	14.17%	37	0.04	0.0285
0.9	19.16%	61	0.08	0.044
0.92	23.75%	113	0.17	0.079
0.95	34.58%	126	0.19	0.087
0.99	62.08%	1055	7.55	0.812
0.992	58.33%	1341	9.52	0.914
0.995	69.92%	2308	27.51	1.431
0.997	77.92%	3746	72.11	2.388
0.999	84.58%	14328	1054.13	8.768

Table 7: The performance of RNSAP-1

R	DR	NOD	Ttrain	Ttest
0.1	48.75%	195	3.58	0.15
0.2	58.33%	379	12.76	0.37
0.3	64.58%	441	19.31	0.52
0.4	71.25%	589	36.84	0.61
0.5	82.08%	798	77.43	0.72
0.6	83.33%	927	97.91	0.84
0.7	83.75%	991	146.24	0.92
0.8	85.42%	1093	173.32	0.97
0.9	88.92%	1134	231.22	1.08
1.0	89.25%	1178	242.35	1.16

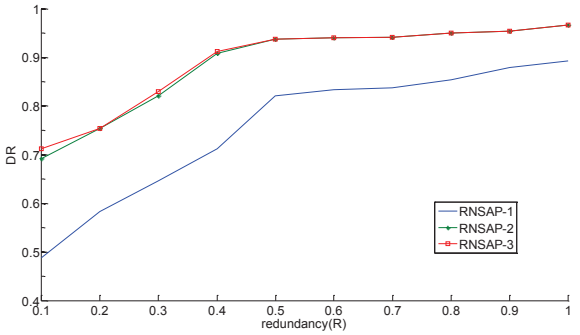
Both V-detector and RNSAP-1 generate detectors randomly, however RNSAP-1 uses the redundancy (R) instead of the Estimate Coverage (c_0) as the algorithm termination condition. In Ttable 6 (V-detector), when (c_0) grew up from 0.997 to 0.999, DR only increased by approximately 6.6%. By contrast, NOD (number of detectors) increased approximately 400%. Ttrain increased approximately 1000% and Ttest increased approximately 300%. This is resultant from highly redundant detectors in feature space. Theses redundant detectors overlapped with each other can hardly improve the detector rate, but wasted lots of training resource. Comparing Table 6 (V-detector) to Table 7 (RNSAP-1), when DR got 84.58% (c_0=0.999), V-detector generated 14328 detectors, and took 1054 seconds; obtained a similar DR 85.42%, RNSAP-1 only generate 1093 detectors and took 209 seconds. RNSAP-1 got the highest DR 89.25% only generate 1178 detectors, and took 274 seconds. The experiment result revealed that by adopting redundancy testing, RNSAP-1 removed a plenty of redundant detectors and enhanced the quality of conventional detectors, so that RNSAP-1 achieved a higher detection rate with less detectors and training time.

Table 8: Experiment setting of 3 levels RNSAP on Haberman's Survival dataset

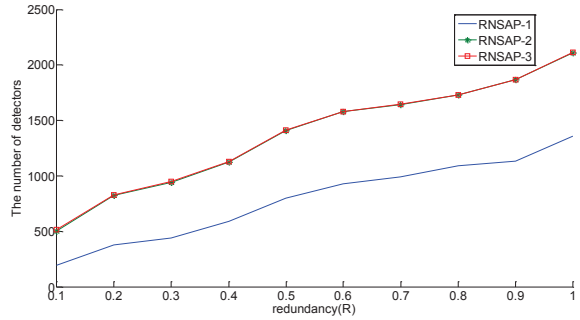
Algorithm	Training set	Testing set	r_s
RNSAP-1	*100%normal*	*50%anomalous*	*0.01*
RNSAP-2	*100%normal*	*50%anomalous*	*0.01*
RNSAP-3	*100%normal,30%anomalous*	*70%anomalous*	*0.01*

As presented in section 4.2, the RNSAP-1 only generates conventional detectors, RNSAP-2 generates both conventional detectors and detectors with spatial preference, and the RNSAP-3(complete RNSAP) generates 3 kinds of detectors. As shown in Fig. 9(a), by training the detectors with spatial preference and feature preference, RNSAP-3 improved the lowest detection rate from 48.75% to 71.23%, and improved the highest detection from 89.25% to 96.72%. In Fig. 9(b) and Fig. 9(c), for training the detectors with evolutionary preference the number of detectors grew, however the training time increased less than 15%. It is worth mentioning in Fig. 9, the detectors with feature preference were rarely generated when R reached 0.5, this because in 3-dimensional space the conventional detectors and the detectors spatial preference had covered nearly all of the feature space.

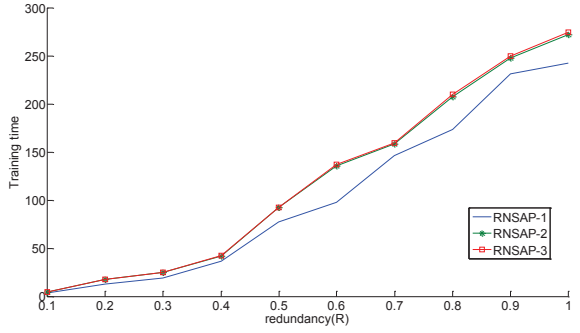
At last, compare Table 6 to Fig. 9, for V-detector, when DR was at 84.58%, 14328 detectors were needed at a cost of 2368 seconds; RNSAP-3(complete RNSAP) improved DR to 96.72 % (R=1), and only needed 2112 detectors at a cost of 274.47 seconds.



(a) The comparison of detection rate



(b) The comparison of number of detectors



(c) The comparison of training time

Figure 9: The comparison of 3 levels RNSAP

4.4 Experiments on KDD CUP99 dataset

In this section, a contrast experiment is carried on the 41-dimension dataset (KDDCUP-99). The same as in section 4.3, firstly a comparison of V-detector and RNSAP-1 (experiment setting is shown in Table 9) is given in Table 10-Table 12. Secondly, a comparison of RNSAP-1, RNSAP-2 and RNSAP-3(the experiment setting as shown in Table 13) is shown in Fig. 10. In these tables, c_0 is Estimate Coverage; R is redundancy; DR is Detection rate; FR is False alarm rate; MND is The maximum number of detectors; $Ttrain$ is training time; $Ttest$ is testing time.

In Table 10, by using the estimated coverage (c_0) as the termination condition, V-detector can barely generate detectors, it leads to the poor performance. For example, when $c_0 = 99.9\%$ only 4 detectors were generated. Although the number of detectors is scarce, the mean radius of theses detectors reached 3. According to the volume of hyper-sphere Eq.13, when dimension n=41 and the hypersphere radius r=3, the *volume* (V) of single 41-dimensional detector is approximately $5^{*}10^{11}$. In contrast, the feature space was a 41-dimension hypercube with total volume=1, so the single detector had covered almost whole feature space (central area). In that situation, the "point estimate" Eq.2 would satisfy quickly. It caused to the algorithm be terminated unexpected when the detectors were not enough. More importantly, as discussed in section 3.1, in high-dimension space a part of the samples would distribute in the subspace (edge area), the few mature 41-dimensional conventional detectors hardly cover these samples.

In Table 11, by using the Maximum Number of Detectors (MND) as the termination condition, V-detector can get DR=76.82%. It reflected that if there were enough detectors, V-detector still can recognize the training samples in high-dimension space. But the disadvantage was obvious by using maximum number of detectors (MND) as the termination threshold: firstly, the MND is difficult to accurately forecast; second, with the detectors overlapping each other in feature space, it cannot guarantee that the mature detectors set can cover enough nonself space when MND reached the threshold; lastly, plenty of redundant detectors cannot improve the detector rate but wasted the calculation resource. In Table 11 after MND reached 10000, the DR fluctuated around 74%. Specially, compared to MND = 50000, when MND= 10000 the detectors increased about 500%, Ttrain increased about 900%, but the detection rate increased less than 3%.

In Table 12, RNSAP-1 adopted the redundancy(R) as the terminate condition to generate 41-dimension conventional detectors. Compared to Table 10 (V-detector using c_0), RNSAP-1 can improve the DR to 76.91% by generating enough detectors. Compared to Table 11 (V-detector using MND), RNSAP-1 can achieve the similar performance with less detectors and training time. In Table 11, when DR=76.82% and FR=1.59%, V-detector generated 50000 detectors and cost 1814 seconds. In Table 12, when DR=76.91% and FR=1.77% (R=1), RNSAP-1 only need 585 detectors and 46.11 seconds.

$$V = \frac{r^n \pi^{n/2}}{\tau\left(\frac{n}{2}+1\right)}, where$$
$$\tau\left(\frac{n}{2}+1\right) = \begin{cases} (n/2)!, n\ is\ even \\ \frac{\pi^{1/2} \star n!}{2^n((n-1)/2)!}, n\ is\ odd \end{cases} \quad (13)$$

Table 9: Experiment setting of 3 levels RNSAP on KDDCUP99

Algorithm	Training set	Testing set	r_s
V-detector	*50% normal*	*randomly 50% anomalous*	*0.003*
RNSAP-1	*50% normal*	*randomly 50% anomalous*	*0.003*

Table 10: The performance of V-detector (c_o)

c_0	DR	FR	NOD	Ttrain	Ttest
0.8	5.71%	0.056%	2	0.039	0.078
0.85	6.49%	0.068%	2	0.042	0.085
0.9	7.92%	0.046%	2	0.040	0.079
0.92	8.32%	0.062%	2	0.044	0.089
0.95	8.96%	0.024%	2	0.047	0.083
0.99	9.72%	0.038%	2	0.049	0.092
0.992	9.82%	0.041%	3	0.052	0.113
0.995	9.93%	0.047%	3	0.056	0.093
0.997	10.51%	0.052%	3	0.057	0.108
0.999	12.07%	0.064%	4	0.088	0.113

Table 11: The performance of V-detector (MND)

MND	DR	FR	Ttrain	Ttest
300	52.12%	0.41%	4.92	1.86
800	61.03%	0.62%	13.81	5.15
1500	66.32%	0.79%	24.47	11.90
3000	63.91%	0.94%	49.45	19.41
5500	68.40%	1.03%	94.83	37.60
10000	73.88%	1.24%	207.22	53.21
20000	74.42%	1.41%	452.76	84.53
30000	75.49%	1.45%	805.68	171.96
40000	76.53%	1.57%	1308.16	336.68
50000	76.82%	1.59%	1814.13	575.38

Table 12: The performance of NSAP-1

R	DR	FR	NOD	Ttrain	Ttest
0.1	63.91%	0.66%	264	10.63	2.914
0.2	64.62%	0.73%	313	11.25	3.231
0.3	66.52%	0.81%	321	13.93	3.337
0.4	71.24%	1.09%	402	19.32	3.466
0.5	71.93%	1.18%	424	21.93	3.486
0.6	73.69%	1.17%	474	27.20	3.626
0.7	75.18%	1.29%	492	33.84	3.748
0.8	75.51%	1.43%	517	38.27	3.799
0.9	76.63%	1.61%	536	41.95	3.887
1.0	76.91%	1.77%	585	46.11	3.953

The comparison of the 3 levels RNSAP shown in Fig. 10. The same as the low-dimension experiment, by training the detectors with spatial preference and feature preference, RNSAP-3 can improve the DR with similar FR and acceptable training cost. In Fig. 10(a), RNSAP-3 improved the lowest detection rate from 63.91% to 86.84%, and improved the highest detection from 76.91% to 91.24%. Although the DR had been improved more than 15%, the *FR* only had been raised less than 0.15% (Fig. 10(b)). In Fig. 10(c), on each redundancy(R) about more than 1000 detectors with evolutionary preference were generated, however the training time increased less than 300 seconds.

Table 13: Experiment setting of 3 levels RNSAP on KDDCUP99

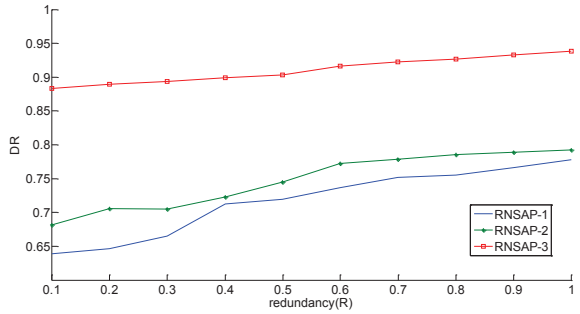
Algorithm	Training set	Testing set	r_s
RNSAP-1	*100%normal*	*50%anomalous*	*0.003*
RNSAP-2	*100%normal*	*50%anomalous*	*0.003*
RNSAP-3	*100%normal,30%anomalous*	*70%anomalous*	*0.003*

At last, compare Table 10 to Fig. 10, if V-detector adopted c_0 as the termination condition, the DR can only achieve about 12%, the shorter training time was due to almost no detectors being generated; V-detector adopted MND as termination condition, when DR=76.82% and FA=1.59%, 50000 detectors need generated and cost 1814 seconds; RNSAP-3(complete RNSAP) improved DR to 91.24% (R=1), need only 2086 detectors and cost 313.46 seconds.

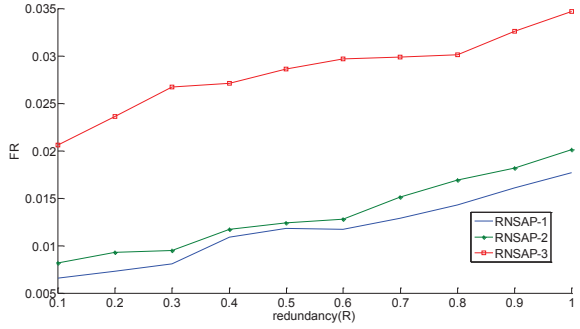
5 Conclusion

The negative selection algorithm has caught the attention of researchers due to its unique property of anomaly detection. However, the problem about how to generate effective detectors in high-dimensional space has not been solved properly in previous research work and artificial immune applications. This paper introduces a real negative selection algorithm with evolutionary preference (RNSAP). By using redundant as the algorithm termination threshold and generating detectors with evolutionary preference, RNSAP can cover the nonself space more effectively in high-dimensional space. Theoretical analysis and experimental results show that RNSAP has better time efficiency and detector quality compared with classical negative selection algorithms, and it can be competent in the task of anomaly detection for both low-dimensional space and high-dimensional space.

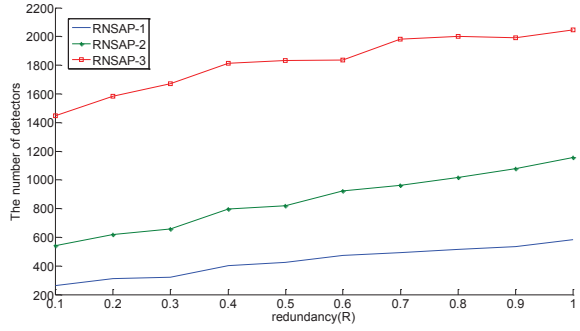
Acknowledgement: This work has been supported by the National Key Research and Development Program



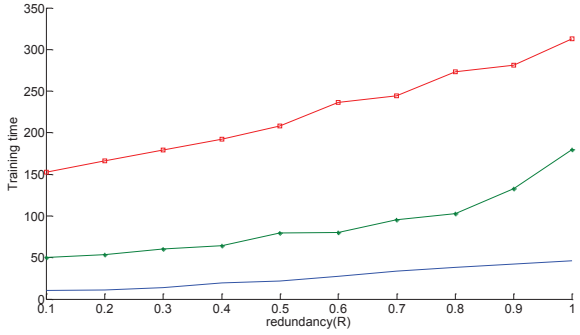
(a) The comparison of detection rate (DR)



(b) The comparison of false alarm rate (FR)



(c) The comparison of the number of detectors (NOD)



(d) The comparison of the training time ($Ttrain$)

Figure 10: The comparison of the 3 levels RNSAP.

of China under Grant No. 2016YFB0800604 and No. 2016YFB0800605, National Natural Science Foundation of China under Grant no.61173159, the National Natural Science Foundation of China under Grant no.614020308.

References

[1] Forrest S., Perelson A.S., Lawrence A., Cherukuri R., Self- Nonself Discrimination in a Computer, Proceedings of IEEE Computer Society Symposium on Research in Security and Privacy, (16-18 May 1994, DC, USA), DC, 1994, 202-212.

[2] Laurentys C.A., Ronacher G., Palhares R.M., Caminhas W.M., Design of an Artificial Immune System for fault detection: A Negative Selection Approach, Expert Syst. App., 2010, 37, 5507-5513.

[3] Jinquan Z., Zhiguang Q., Weiwen T., Anomaly Detection Using a Novel Negative Selection Algorithm, J. Comput. Theor. Nanosci., 2013, 10, 2831-2835.

[4] Idris I., Selamat A., Omatu S., Hybrid email spam detection model with negative selection algorithm and differential evolution, Eng. Appl. Artif. Intell., 2014, 28, 97-110.

[5] Hualong W., Bo Z., Overview of current techniques in remote data auditing, Appl. Math. Nonlinear Sci., 2016, 145-158.

[6] Gonzalez F., Dasgupta D., Nio L.F., A Randomized Real- Valued Negative Selection Algorithm, Lect. Notes. Comput. Sc., 2003, 2787, 261-272.

[7] Ji Z., Dasgupta D., Real-Valued Negative Selection Algorithm with Variable-Sized Detectors, Lect. Notes. Comput. Sc, 2004, 3102, 287-298.

[8] Maoguo G., Jian Z., Jingjing M., Licheng J., An efficient negative selection algorithm with further training for anomaly detection, Knowl-Based. Syst., 2012, 30, 185-191.

[9] Wen C., Tao L., XiaoJie L., Bing Z., A negative selection algorithm based on hierarchical clustering of self set, Adv. Mater. Res., 2013, 56, 1-13.

[10] Poggiolini M., Engelbrecht A., Application of the featuredetection rule to the Negative Selection Algorithm, Expert Syst. App., 2013, 40, 3001-3014.

[11] Fernandez M., A survey on fractal dimension for fractal structures, Appl. Math. Nonlinear Sci., 2016, 1, 437-472.

[12] Perelson A.S., Weisbuch G., Immunology for physicists, Rev. Mod. Phys., 1997, 69, 1219-1267.

[13] Yang Z., Meyerhermann M., George L.A., Figge M.T., Khan M., Goodall M., et al., Germinal center B cells govern their own fate via antibody feedback, J. Exp. Med., 2013, 210, 457-464.

[14] Ji Z., Dasgupta D., Estimating the detector coverage in a negative selection algorithm, Proceedings of Genetic and Evolutionary Computation Conference (25-29 June 2005, Washington DC, USA), New York, 2005, 281-289, DOI: 10.1145/1068009.1068056.

[15] Ji Z., Dasgupta D., V-detector: An efficient negative selection algorithm with "probably adequate" detector coverage, Inform. Sciences, 2009, 179, 1390-1406.

[16] Haberman datasets. http://archive.ics.uci.edu/ml/datasets/Haberman.

[17] Kddcup datasets. http://archive.ics.uci.edu/ml/datasets/KDD+Cup+1999+Data.

[18] Stibor T., Timmis J., Eckert C., On the Use of Hyperspheres in Artificial Immune Systems as Antibody Recognition Regions, Proceedings of International Conference on Artificial Immune Systems (4-6 September 2006, Portugal), Portugal, 2006, 215-228.