Applying both Positive and Negative Selection to Supervised Learning for Anomaly Detection

Xiaoshu Hang & Honghua Dai
School of Information Technology,
Deakin University
221 Burwood Highway,Burwood,
VIC.3125, Australia
{xhan, hdai}@deakin.edu.au

ABSTRACT
This paper presents a novel approach of applying both positive selection and negative selection to supervised learning for anomaly detection. It first learns the patterns of the normal class via co-evolutionary genetic algorithm, which is inspired from the positive selection, and then generates synthetic samples of the anomaly class, which is based on the negative selection in the immune system. Two algorithms about synthetic generation of the anomaly class are proposed. One deals with data sets containing a few anomalous samples; while the other deals with data sets containing no anomalous samples at all. The experimental results on some benchmark data sets from UCI data set repertory show that the detection rate is improved evidently, accompanied by a slight increase in false alarm rate via introducing novel synthetic samples of the anomaly class. The advantages of our method are the increased ability of classifiers in identifying both previously known and innovative anomalies, and the maximal degradation of overfitting phenomenon

Categories and Subject Descriptors: I. Computing Methodologies- artificial intelligence.

General Term: Algorithm

Keywords
Artificial immune system, supervised learning, anomaly detection, positive selection, negative selection.

1. INTRODUCTION
Learning or detecting rare events from observed data has drawn a lot of attention in recent years. Rare events are the events that occur very infrequently, i.e. their frequency ranges from 0.1% to less than 5%. However, when they do occur, their consequences can be quite dramatic and quite often in a negative sense. Such rare events are in general called anomalies. They might be network intrusions[4][5], financial/telecom fraudulent transactions[6] or other risky events in the corresponding domains. Detecting such rare events has been investigated via either supervised learning or unsupervised learning approaches. Unsupervised learning methods, particularly anomaly detection, have dominated the research stream. Anomaly-based approaches build models based on only the normal data. The advantages are that they do not require any prior knowledge about the anomalies and can detect innovative ones. However, they tend to either create a large number of detectors which cause the efficiency problem, or result in excessive false alarms. Supervised learning approaches build models for rare events based on labeled data (the training data) and use them to predict anomalous event(the testing data). The advantage is their efficient prediction of previously known anomalies, but the defect is their incapability in identifying innovative anomalies. In machine learning domain, such supervised learning tasks are remarkably characterized with highly class-skewed distribution, generally over 95% of the normal data versus less than 5% of the anomalous data. This causes classifiers biased to the normal class (the majority) and to ignore the anomaly class (the minority), and consequently results in a relatively poor performance on identifying anomalies [1].

In some domains, the anomalous events keep changing over time. New computer virus, new network attacks and new fraudulent transactions occur incessantly, frustrating the users in a variety of ways due to their significant difference from those samples in the observed data. These new samples can be regarded as subcategories of the anomaly in a sense and cause classifiers ineffective in recognizing new patterns. This problem can be solved to some extent by feeding the classifier with artificial anomalous examples in the training phase. The artificial anomalous samples contain the patterns of potentially innovative anomalies.

An extreme case is that the observed data contains no examples of the anomaly class at all. The reason might be the difficulty or the very high cost of obtaining anomalous samples, or the extremely rare occurrence of the events that are viewed as anomaly. In this case, it is reasonable to generate artificial anomalous samples to train the learner. The difficulty in this situation is where the synthetic samples are located in the data space.

In this paper, we are motivated to balance the data sets by generating synthetic anomalous samples. The strategy of generating synthetic samples is inspired from the two regulations in the human immune system: positive selection and negative selection. The advantage of the immunology-inspired synthetic generation of anomalous samples is that it is suitable for data sets with or without examples of the anomaly class. If the training data

GECCO'05, June 25–29, 2005, Washington , D.C., USA.

set contains no examples of the anomaly class, it is viewed as the extreme case with class skew ratio n:0, where n refers to the number of examples of the normal data. The data sets dealt with in the paper are categorical/discrete, but this does not mean that the immunology-based approach is limited to this sort of data.

The remainder of the paper is organized as follows: Section 2 briefly reviews the previous work and anomaly detection via classification. Section 3 briefly introduces the two regulations in the human immune system. In section 4, a co-evolutionary genetic algorithm for evolving patterns of the normal class and two algorithms for generating synthetic samples are introduced. In section 5, experiments on benchmark data sets are conducted to test the effectiveness of our approach. Section 6 provides the conclusions.

2. DETECTING ANOMALIES VIA SUPERVISED LEARNING

In this paper, we treat the problem of anomaly detection equivalent to supervised learning from class-imbalance data sets (see Figure. 1). The problem has been investigated in a number of ways in the domain of machine learning. In general, the methods fall into two categories: at data level and at algorithm level. At data level, work in the past mainly focused on re-sampling strategies[1][2]. There are three resampling strategies: under-sampling the normal class, over-sampling the anomaly class and their combination. Balancing a class-skewed data set by over-sampling the anomaly class is beneficial in detecting potentially new anomalies because the novel synthetic samples can be The performance of a classifier for anomaly detection can be evaluated by the following two measures:

$$TP\_Rate = \frac{TP}{FN + TP}$$

$$FP\_Rate = \frac{FP}{FP + TN}$$

where TP_rate is equivalent to the detection rate and FP_rate corresponds to the false alarm rate in anomaly detection systems.

With regard to anomaly detection via supervised learning, we also are concerned with the ability of a classifier to identify innovative anomalies. A formal definition of innovative anomaly is defined as follows:

viewed as new anomalies in a sense. However, oversampling can also result in ovetfitting which is prone to generating more and longer classification rules. Under-sampling normal samples may remove some important examples, resulting in the loss of information. Therefore, intelligent over-sampling methods can generate valid samples of the anomaly class and in the meanwhile maximally degrade overfitting.

SMOTE[3], standing for Synthetic Minority Over-sampling TEchniques, is a representative of over-sampling strategy in machine learning community. It generates synthetic samples by operating the "feature space" rather than "data space". It over-samples the minority (anomaly) class by taking each minority class sample and introducing synthetic examples along the line segment joining any/all of the *k* minority class nearest neighbours. It is claimed that a combination of SMOTE and under-sampling can achieve better classifier performance than only under-sampling strategy. In [7], artificial anomalies are regarded as potential network intrusions and used to feed the inductive learner to learn the boundary between the normal and the anomaly class. The only work using immunology-inspired strategy to generate synthetic samples of the anomalous data is from Gonzalez, Dasgupta and Kozma[8]. They use negative selection algorithm to generate non-self samples, and then apply a classification algorithm to generate the characteristic function of the self (or non-self). However, it faces the efficiency problem of generating a large number of valid samples when the amount of self data is large. And it does not avoid the overfitting problem.

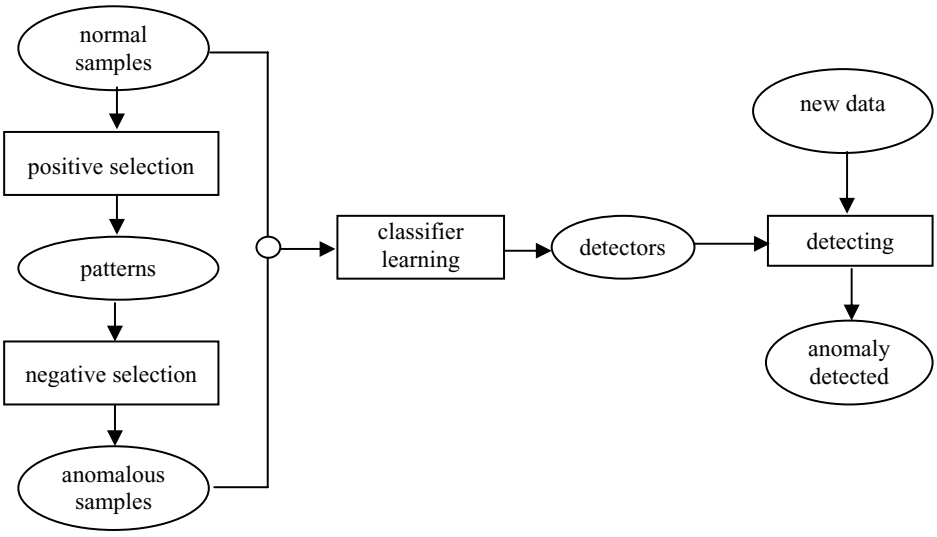

Figure 1. Diagram of immunology-based re-sampling strategy for anomaly detection

Definition 1. An innovative anomaly is defined as a group of anomalous samples which can not be covered by any patterns which are learnt from existing examples of the abnormal class.

Let $R_A=R_1 \cup R_2$ denote the set of rules about the abnormal class learnt from a synthetically balanced dataset, where R_1 cover (but not limit to) the non-synthetic anomalous examples whilst R_2 cover only the synthetic anomalous examples. $|R_2|$ and $|R_A|$ represent the number of rules, respectively. The ability of a classifier in identifying innovation anomalies are calculated as:

$$r = \frac{|R_2| \times Q}{|R_A| \times P}$$

where Q is the number of synthetic anomalous samples covered by R_2 and P is the total number of examples of the abnormal class.

The main problem of supervised learning for anomaly detection is the deficiency of anomalous examples in the training phase, which causes the classifier unable to discover the boundaries between the two classes. The natural way to solve the problem is to generate artificial anomalous samples to feed the learner and the objective of doing this is to form an as large decision region as possible for the anomaly class. The flaw of SMOTE is that for each minority class sample it generates synthetic samples only within a convex polygon with the neighbors as vertexes, and as a whole, it also generates synthetic samples within a bigger a convex polygon circumscribed by the seed anomalous examples. This restricts its ability of generating novel anomalies.

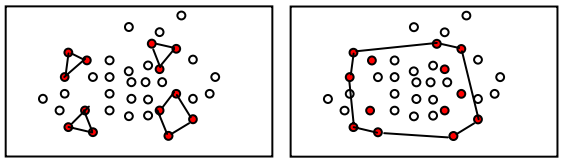

Figure 2. The way of SMOTE generating synthetic samples

However, randomly generating synthetic samples tends to result in overfitting, especially when the anomaly class is rare cases. Overfitting here is caused by inserting synthetic anomalous samples into the regions of the normal class, breaking the purity of the regions, and consequently results in more and longer classification rules which are error prone in prediction. Overfitting is considered as one of the evaluation criteria of resembling strategy. In decision tree, overfitting is eliminated by post-pruning the tree. Therefore, the tree size and the number of instances pruned can be used to measure the degree of overfitting caused by over-sampling the anomaly class. Figure 2 shows the way of SMOTE generating synthetic samples. In figure 3, a number of synthetic examples are appropriately generated.

In the extreme case that no anomalous examples obtained in the training phase, synthetic anomalous data are generated based on the regularity of negative selection in the immune system. That is, the randomly generated instances are checked to avoid colliding with the patterns of the normal class.

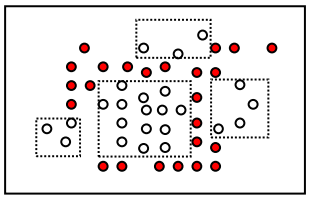

Figure 3. Generating appropriate anomalous samples

3. INSPIRATION FROM HUMAN MMUNE SYSTEM

In the natural immune system, when an *antigen presenting cell* (APC) roams the body, *T-cells,* which have receptor molecules that enable each of them to recognize a different peptide-MHC combination, are activated and emit chemical signals to other immune cells. The *B-cells*, which also have receptor molecules of a single specificity on their surface, then respond to those signals. When activated, the B-cells divide and differentiate into *plasma cells* that secrete *antibody proteins.* By binding to the antigens they find, antibodies can neutralize them or precipitate their destruction by *scavenging cells*. Some T-cells and B-cells become *memory cells* that persist in the circulation and boost the immune system's readiness to eliminate the same antigen if it presents itself in the future.

From the viewpoint of pattern recognition, the most important feature of the immune system is that B-cells and T-cells have receptors on their surfaces. These receptors can recognize non-self antigens at the molecular level and based on the *shape complementary* between the binding site of the receptor and a portion of the antigen called an epitope.

Negative selection: T-cells undergo a process called negative selection before they develop into mature immune cells. During the process of negative selection, immature T–cells in the thymus are tested to see if they bind to self antigens. If the T-cells bind to any self antigens they are eliminated, otherwise they become mature and then distributed to lymph notes for detecting non-self antigens. Negative selection make mature T-cells have the feature of self tolerance.

Positive selection: Non-self antigens presented to T-cells for binding are carried by Antigen Presenting Cells(APCs). APCs are special cells that engulf non-self antigens distributed throughout the body and convey engulf antigens to a specific form that allow T-cells to bind them. The MHC molecules of APCs perform a key role in this transformation. Positive selection selects only those T-cells that bind to self-MHC/peptide binding on APCs in the thymus. The T-cells which do not bind self–MHC/peptide are eliminated.

Although the results of positive selection are some specific T-cells, MHC/peptide plays the key role in positive selection, which inspires us to find out them in an artificial immune system. We think that, in artificial immune systems, MHC/peptide refers to the boundary between the normal class and the anomaly class. The boundary is determined by the patterns of the normal data. Learning self- MHC/peptide in an artificial immune system is implemented by learning the patterns of the normal class.

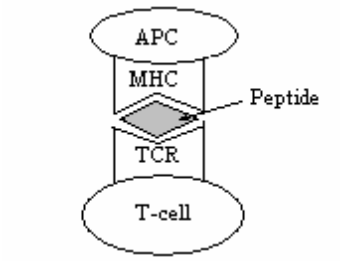

Figure 4. MHC/Peptide bind to TCR

4. GENERATING SYNTHETIC ANOMALOUS SAMPLES

4.1 Co-evolving Patterns of the Normal Class

Immunology-based synthetic generation of anomalous samples includes two separate phases: learning patterns of the anomaly class via co-evolutionary genetic algorithm and generating synthetic samples of the anomaly class based on the regulation of negative selection.

A pattern (schema) in a *n*-dimensional symbolic space refers to a region filled with examples of one class. In genetic algorithm, a schema is defined as a hyperplane. For instance, a number of patterns in a 5-dimensional symbolic space are represented as follows:

$A_1 * C_3D_4E_5$
$A_1 ** D_4E_5$
$A_1 *** E_5$
$A_1 ****$

The symbol "*" in the above expressions represents the irrelevance of the corresponding attribute to the schema. The order of a schema is defined as the number of symbol "*" the pattern contains.

In this paper, we exploit a co-evolutionary genetic algorithm to evolve a number of patterns about the normal class. The population consists of a number of non-interbreeding subpopulations of species. Each represents only a part of the problem, and there is neither cooperation nor competition among subpopulations. Although nothing explicitly prevent multiple subpopulations from containing the identical schema, in practice, each subpopulation tends to be dominated by one species. In our work, each subpopulation is randomly initialized with a species which will converge on a specific schema after some generations of evolution. All the schemas together form the decision boundary of the normal class, analogous to self MHC/peptide in the natural immune system. This approach has been applied to concept learning[12] and Web document classification[13].

Algorithm_1: Co-evolve patterns of the normal data

Input : A data set and a number of parameters.
Output: A group of patterns

```
1   Encode the data set into binary strings;
2   Initialize the first subpopulation;
3   While the number of patterns is less than the threshold
4      For each subpopulation
5         calculate the fitness of each individual;
6         do selection, crossover and mutation;
7      endfor
8      calculate the total fitness of the N populations;
9      if the total fitness fails to increase for a number of
   consecutive generations
10        remove the individuals in that do not contribute to
   the total fitness;
11        add a subpopulation with a new species;
12     endif
13  endwhile
```

Figure 6. Algorithm of co-evolving patterns of the normal data

Individuals are designed to consist of four sections (see Figure 5). Each attribute in data sets is encoded as three binary bits, which can encode 8 different values. Both the gene section and the mask section have 3 times bits as many as the number of the attributes. The schema is obtained by replacing the consecutive three bits with *** if the corresponding three bits in the mask section are 000. The order is obtained by converting the binary value into a decimal one and then divided by 3. The first bit in fitness section is a sign, 1 for negative and 0 for positive, the rest 15 bits encode the decimal value which could be either positive or negative.

The two fitness functions in the above algorithm are designed as follows

$$fitness_{individual}(x) = k - \delta, \quad \delta = \begin{cases} 10000, & if\ x\ covers\ \ abnormal\ samples \\ 0, & otherwise \end{cases}$$

$$fitness_{total} = \sum_{i=1}^{n} fitness_{individual}$$

where k is the number of examples of the normal class covered by the individual and δ is a punish factor which is big enough to lead to a negative value if an individual covers any samples of the anomaly class

:

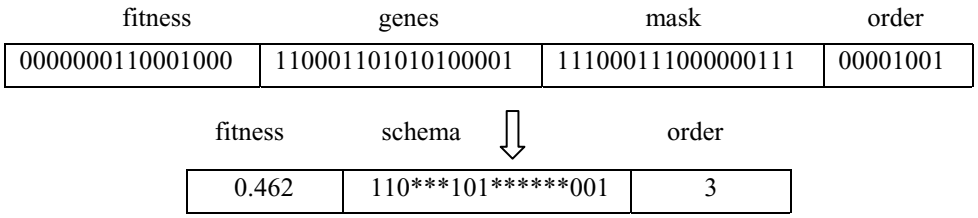

Figure 5. Encoding scheme

Within a subpopulation, genetic operations include selection, crossover and mutation. Children are created by selecting two parents from the same species via fitness-proportionate selection with balanced linear scaling and then using uniform crossover and bit flipping mutation. The subpopulation tends to maintain the schema once it find and also has the ability of evolve out a new schema. In our experiments, subpopulation size = 100, crossover rate= 0.65, mutation =0.15 for each data set.

4.2 Synthetic Generation of Anomalous Samples

4.2.1 Synthetic Generation with Seed Examples

The algorithm starts with collecting all the vacant neighbours of the examples of the anomaly class, leading to a candidate set *C* of synthetic anomalous samples. In a n-dimensional space, a data point maximally has 2 neighbours in each dimension, thus 2n neighbours in the space. A vacant neighbour means the neighbour is neither an instance of the normal class nor an instance of the anomaly class. If a neighbour is empty, then label it as a synthetic sample of the anomaly class and store it in set *C* as a candidate. The algorithm then checks each candidate sample in *C* to see if it is covered by any schema of the normal class. Those candidate samples covered by patterns of the normal class are removed. The algorithm probabilistically removes some synthetic samples which are not covered by any patterns of the normal class at all. This operation avoids the situation that, for some data sets of high dimensionality or/and small sample size, the synthetic samples are prone to being generated by only a few examples of the anomaly class. If the number of the synthetic samples is less than the required number for balancing the data set, repeat this process until the data set is balanced. The algorithm is described as follows:

Algorithm_2: Synthetic generation with seed examples

Input: a set S of patterns, a set A of examples of the anomaly class and *N*
Output : a set *E* of synthetic samples of the anomaly class

```
1.  E=A; count=0;
2.  while count < N
3.      C =Valid_neighbour (A );
4.      for each element c in C
5.          for each s in S
6.              Match(c, s );
7.          endfor
8.          if c matches any s then
9.              remove c;
10.         else
11.             remove c probabilistically;
12.         endif
13.     endfor
14.     count = count + |C| ;
15.     E=E ∪C;
16.     if count < N then
17.         A=C;
18.     endif
19. endwhile
20. output E
```

Figure 7. Algorithm of generating synthetic anomalies

4.2.2 Synthetic Generation without Seed Examples

Data sets in the extreme case contain no examples of the anomaly class at all. The set of patterns input to Algorthm_3 is slightly different from the set of patterns input to Algorithm_2 since there is no constraint of anomalous samples during the process of co-evolutionary. The algorithm begins with randomly selecting a vacant position in the space as a seed and labeling it as a sample of the anomaly class. A seed position is not covered by any pattern of the normal class. All the neighbours of the seed position are collected and checked if they are covered by the patterns. A neighbour without being covered by any pattern is probabilistically selected as a synthetic sample. The algorithm repeats the process of negative selection until a balanced data set is obtained.

Algorithm_3: Synthetic generation without seed examples

Input: a set S of patterns, *N*: the number of synthetic samples
Output: a set *E* of synthetic samples of the anomaly class

```
1.  E=∅; count=0;
2.  while count < N
3.      b=randomly_generate_seed( );
4.      C=valid_neighbour (b);
5.      for each element c in C
6.          for each s in S
7.              Match(c, s );  /* collision check */
8.          endfor
9.          if c matches any s then
10.             remove c; break;
11.         else
12.             remove c probabilistically;
13.         endif
14.     endfor
15.     count = count + |C| ;
16.     E=E ∪C;
17. endwhile
18. output E
```

Figure 8. Algorithm of generating synthetic anomalies without seeds.

5. EXPERIMENTAL RESULTS

We assessed the effectiveness of our method by conducting experiments on some UCI datasets

We choose 14 data sets from UCI data set repertory. 8 of them consist of nominal attributes and the rest 6 consist of discrete attributes. If a dataset is multi-class, we mapped it into a two-class dataset with class-skewed distribution by labeling the instances of one class or two as *anomaly* and the reminder as *normal*. Table 1 shows the class natural distributions and the class extreme distributions. For each dataset, we have three versions: the first with class natural distribution, the second obtained by balancing the first version and the third version by balancing the extreme class distribution.

We then applied C4.5 and Naïve Bayes to each version to examine the classification performance, the ability of identifying innovative anomalies and the overfitting degree. The TP_rate and

FP_rate in Table 2 correspond to the detection rate and the false alarm rate, respectively. In order to understand the increased ability of classifiers in identifying innovative anomalies, we first tested the learner's ability of identifying the original anomalous examples with the rules learnt in version 3 where the original anomalous examples in each data set were removed for testing. The results are shown in Figure 9 from which we can see that over 70% of the original anomalous samples are predicted. We then sort out the rules generated by C4.5 in each data set that cover only synthetic anomalous samples and calculate the value of *r* (see Figure 10). We found that *r* is strongly related to the class-skewed ratio in version 1. *Mushroom* and *crx* have the lowest values of *r* because of their closer to the class balanced distributions. The results show that the higher the class-skewed ratio, the higher the value of *r*. Overfitting is measured by both the fraction of the synthetic anomalous samples pruned in the classification and the tree size (see Table 3). The f.a.s.p in table 3 denotes the fraction of anomalous samples pruned. The tree size is represented as leaf nodes/total nodes.

Table 1. Data sets from UCI data set repository

data set	natural distri.	extreme distri.
breast-cancer	201: 85	201: 0
car	1835 : 134	1835 : 0
mushroom	4208 : 3916	4208 : 0
post-operative	88 : 2	88 : 0
primary-tumor	325 : 14	325 : 0
splice-jxn	2423 : 767	2423 : 0
tic-tac-toe	626 : 332	626 : 0
voting	267 : 168	267 : 0
breast-wisc.	458 : 241	458 : 0
crx	383 : 307	383 : 0
german	700 : 300	700 : 0
lung-cancer	23 : 9	23 : 0
lymphography	142 : 6	142 : 0
soybean	290 : 17	290 : 0

Table 2. Performance of classifiers on each data set

Data set	Classifier	version1		version2		version3	
		TP_rate	FP_rate	TP_rate	FP_rat	TP_rate	FP_rate
breast-cancer	C4.5	0.235	0.075	0.816	0.159	0.915	0.124
	Naivebayes	0.435	0.144	0.896	0.189	0.930	0.085
car	C4.5	0.785	0.011	0.993	0.015	0.997	0.028
	Naivebayes	0.355	0.000	0.967	0.028	0.886	0.267
mushroom	C4.5	1.000	0.000	0.992	0.000	0.964	0.000
	Naivebayes	0.921	0.008	0.928	0.025	0.945	0.047
post-operative	C4.5	0.000	0.031	0.558	0.375	0.806	0.219
	Naivebayes	0.038	0.016	0.529	0.484	0.847	0.172
primary-tumor	C4.5	0.000	0.000	0.790	0.164	0.905	0.085
	Naivebayes	0.000	0.000	0.729	0.194	0.936	0.079
splice-jxn	C4.5	0.925	0.061	0.965	0.067	0.885	0.083
	Naivebayes	0.813	0.036	0.877	0.037	0.729	0.044
tic-tac-toe	C4.5	0.741	0.008	0.842	0.133	0.660	0.163
	Naivebayes	0.413	0.155	0.714	0.268	0.690	0.296
voting	C4.5	0.958	0.045	0.963	0.064	0.910	0.086
	Naivebayes	0.917	0.119	0.948	0.124	0.928	0.124
breast-wisc.	C4.5	0.921	0.046	0.976	0.059	0.996	0.055
	Naivebayes	0.988	0.033	0.993	0.033	1.000	0.037
crx	C4.5	0.980	0.008	0.982	0.010	0.956	0.051
	Naivebayes	0.997	0.000	0.997	0.003	0.948	0.042
german	C4.5	0.380	0.133	0.776	0.250	0.931	0.073
	Naivebayes	0.470	0.144	0.743	0.254	0.984	0.036
lung-cancer	C4.5	0.667	0.174	0.870	0.261	0.739	0.217
	Naivebayes	0.667	0.304	0.739	0.174	1.000	0.087
lymphography	C4.5	0.500	0.000	0.993	0.028	0.880	0.085
	Naivebayes	1.000	0.021	0.986	0.021	0.965	0.056
soybean	C4.5	0.882	0.003	0.973	0.031	0.890	0.062
	Naivebayes	0.941	0.000	0.987	0.000	0.986	0.024

Note: The classification were conducted by using Weka Knowledge Explorer

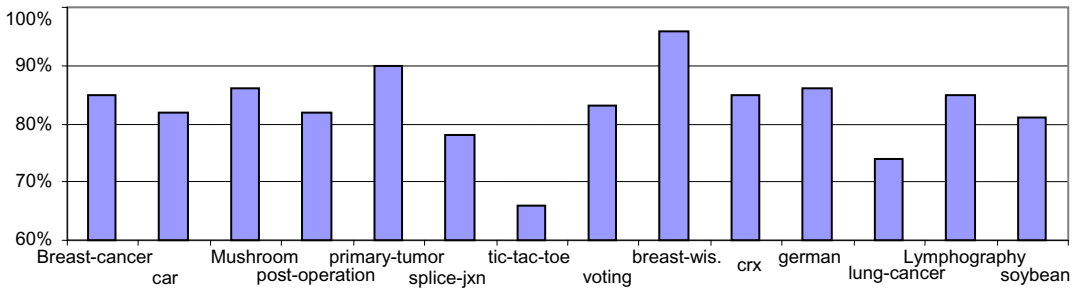

Figure 9. Percentage of the original anomalous samples identified by the rules learnt in version 3

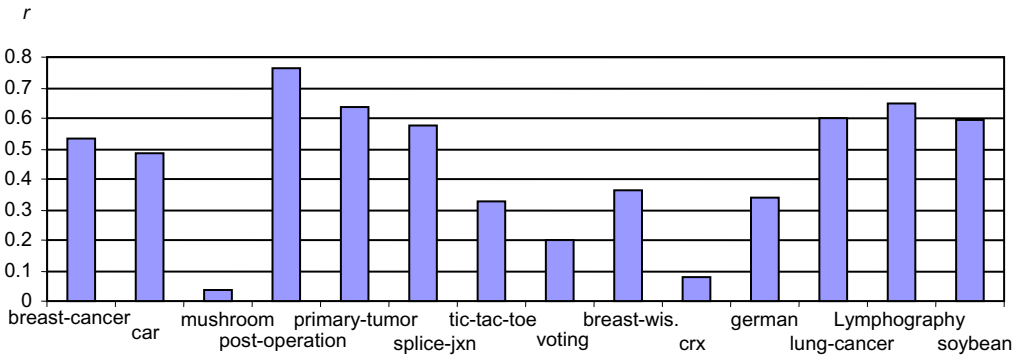

Figure 10. The ability of the classifier identifying innovative anomalous samples measured by *r*

Table 3. Decision tree size and the fraction of anomalous samples pruned in C4.5

Data set	Version 1		Version 2		Version 3	
	Tree size	f.a.s.p	tree size	f.a.s.p	tree size	f.a.s.p
breast-cancer	4/6	61/85	45/59	11/201	23/29	7/201
car	44/63	10/134	55/79	8/1835	71/102	0/1835
mashroom	25/30	0/3916	70/86	0/4208	85/108	2/4208
post-operative	1/1	2/2	22/32	9/88	12/17	6/88
primary-tumor	1/1	9/14	44/78	16/325	24/44	24/325
splice-jxn	61/81	36/767	58/77	28/2423	213/452	46/2423
tic-tac-toe	95/142	27/332	97/145	27/626	133/199	32/626
voting	11/16	3/168	14/19	3/268	17/23	7/268
breast-wisc.	28/31	13/241	46/51	5/458	37/41	7/458
crx	12/17	3/307	12/17	3/383	17/26	7/383
german	69/101	128/300	206/288	53/700	43/71	75/700
lung-cancer	5/7	1/9	8/12	2/23	5/7	2/23
lymphography	3/4	3/6	7/10	1/142	14/22	8/142
soybean	16/22	1/17	18/25	3/290	31/44	7/290

note: f.a.s.p = the anomalous samples pruned/the total anomalous samples

The experimental results are analysed by comparing each two of the three groups:

version1 vs. version 2.

In version 1, both C4.5 and Naive Bayes produce quite low FP_rates in most of the cases, which means that the normal classes are well classified, or the false alarm rates are quite low. But the TP_rates are not satisfying, which means that the detection rates are not satisfying. This explains that classifiers are biased to the normal class. The poor performance on the anomaly classes is also exhibited by the higher values of f.a.s.p in version 1. In version 2, the TP_rates are increased evidently in most of the cases, and accordingly the FP_rate are also slightly increased. This explains that the improvement in detection rate is often accompanied by a slight sacrifice in false alarm rate. The values of f.a.s.p in version 2 degrade sharply and the tree sizes are appropriately augmented. The ability of generating innovative

anomalous samples in version 2 is highly related to the class skew ratio of the data set.

version 1 vs. version 3.

Since all the examples of the anomaly class in version 3 are synthetic ones, we are concerned with the TP_rate, FP_rate and the ability of the classifier in predicting real anomalies. The TP_rates in version 3 are fairly high and the false alarm rates are also slightly higher than those in version 1. We found that data sets in version 1, e.g. breast-cancer, car, german, post-operative and primary-tumor, are poorly classified but well classified in version 3. And the tree size and f.a.s.p in such a case are quite satisfying. The results in Figure 9 show that over 70% of the original anomalous examples can be identified by the rules learnt from the artificial anomalies. This actually demonstrates the ability of the classifier in predicting innovative anomalies and also validates our motivation of generating synthetic anomalous samples.

version 2 vs. version 3.

The difference between version 3 and version 2 for each data set is that in version 2 the original anomalous samples are used as seeds for synthetic generation and used to train the learners, whereas all the anomalous samples in version 3 are artificial ones. In version 2, the synthetic samples are generated to fill in the vacant neighbors of the seed examples of the anomaly class; in version 3, however, they are generated based on negative selection in the human immune system. The classification performances in version 2 are slightly better than those in version 3. This can be explained that the existing samples of the anomaly class provide information in determine the boundaries between the classes.

6. CONCLUSIONS

This paper applied both positive selection and negative selection to supervised learning for anomaly detection via generating synthetic anomalous samples which are viewed as potentially new anomalies. In general, both the existing and synthetic anomalous samples provide important information for determining the boundary between the normal class and the anomaly class, and make the detection more effective. In normal case, the synthetic samples are generated around the seed examples of the anomaly class, whereas the artificial anomalies are generated completely based on negative selection. We are concerned with the ability of classifiers in predicting both previously known and innovative anomalies. Our method is empirically validated via experiments on some symbolic/discrete data sets from UCI data repository. Experimental results show that over 70% of the original anomalous examples can be predicted by the rules learnt from pure artificial examples. The f.a.s.p values in both version 2 and version 3 decrease greatly, accompanied with the appropriate augment of the decision tree size. The advantages of our method include (1) an evident improvement of the performance of classifiers on the anomaly class, (2) the generation of as large decision regions for the anomaly class as possible and, (3) the maximal degradation of overfitting phenomenon

4. REFERENCES

[1] Chawla, N. V., Japkowicz N. and Kotcz. A. Editorial : special issue on learning from imbalanced data sets

[2] Weiss. G. Mining with rarity: A unifying framework.SIGKDD Exploration, 6(1):7-9,2004.

[3] Chawla,N.V. Bowyer, K.W., Fall,L.O.&Kegelmeyer,W.P. (2002), SMOTE: synthetic minority over-sampling Techniques, journal of artificial intelligence research, 16, 321-357.

[4] Dasgupta, D. and Gonzalez, F. "An immunity-based Technique to Characterize Intrusions in Computer Networks", IEEE transaction on evolutionary computation 6(3),pp 1081-1088 June 2002.

[5] Paul, K. harmer, Paul, D. Williams, Gregg H.Gunch and Gary B.Lamont, "An Artificial Immune System Architure for Computer Security Applications" IEEE transaction on evolutionary computation, vol.6. No.3 June 2002.

[6] Fawcett, T. and Provost, F. Adaptive fraud detection, Data mining and knowledge discovery, 1-28(1997).

[7] Fan, W. Miller, M and Stolfo, S. Using Artificial Anomalies to detect unknown network intrusions.

[8] González, F., Dasgupta, D. and Kozma, R. Combining Negative Selection and Classification Techniques for Anomaly Detection. In Proceedings of the Congress on Evolutionary Computation , pp 705-710, Honolulu, HI, May 2002.

[9] Kim, J, Ong, A. and Overill, R. Design of an artificial immune system as novel anomaly detector for combating financial fraud in the retail sector.

[10] Wei-Chou Chen, *et al*, A novel manufacturing defect detection method using data mining approach. In the proceeding of innovation in applied artificial intelligence, IEA/AIE,2004.

[11] Kubat, M, Hole, R.C. and Matwin,S. Machine learning for the detection of oil spills in satellite radar images. Machine Learning, 30(20):195-215,1998.

[12] Potter, M. A. and Kenneth A. De Jong (1998). The Coevolution of Antibodies for Concept Learning. In *Proceedings of the Fifth International Conference on Parallel Problem Solving from Nature*, pp 530-539.

[13] Hang, X. and Dai, H: Constructing Detectors in Schema Complementary Space for Anomaly Detection. GECCO (1) 2004: 275-286.

[14] Poli, R. and Langdon, W. B. *Schema theory for genetic programming with onepoint crossover and point mutation*. Evolutionary Computation, 6(3):231-252, 1998.

[15] Potter, M. A., Jong, K. A. and Grefenstette, J. J., *A coevolutionary approach to learning sequential decision rules*. In Larry J. Eshelman, editor, Proceedings of the 6th International Conference on Genetic Algorithms (ICGA95), pages 366--372. Morgan Kaufmann Publishers, 1995.