EFFECT OF FUZZY RESOURCE ALLOCATION METHOD ON AIRS CLASSIFIER ACCURACY

[1] SHAHRAM GOLZARI, [2] SHYAMALA DORAISAMY, [3] MD NASIR SULAIMAN, [4] NUR IZURA UDZIR

[1]PhD Candidate, Faculty of Computer Science and Information Technology, UPM , Malaysia

[2]Asstt. Prof., Faculty of Computer Science and Information Technology, UPM , Malaysia

[3]Assoc. Prof., Faculty of Computer Science and Information Technology, UPM , Malaysia

[4]Asstt. Prof., Faculty of Computer Science and Information Technology, UPM , Malaysia

E-mail: golzari@ieee.org ,{shyamala,nasir,izura@fsktm.upm.edu.my}

ABSTRACT

Artificial Immune Recognition System (AIRS) is an immune inspired classifier that is comparable to many popular classifiers. Many researches have been conducted to improve the accuracy of AIRS and to identify the significant components of AIRS that could empower it for better performance. Some of these researches have focused on the resource allocation component of AIRS. This study investigates the difference between the accuracy of AIRS using a fuzzy resource allocation approach with the accuracy of the current resource allocation technique by using statistical methods. The combination of ten-fold cross validation and t-test was used as evaluation method and algorithms tested on benchmark datasets of UCI machine learning repository. Based on the results of experiments, using the fuzzy resource allocation technique increases the accuracy of AIRS in majority of the datasets. However, the increase is significant in minority of datasets.

Keywords: *Artificial Immune System, AIRS, Fuzzy Resource Allocation, Statistical Evaluation*

1. INTRODUCTION

Natural computation is the study of computational systems that use ideas and get inspiration from natural systems, including biological, ecological and physical systems. One branch of natural computation is Artificial Immune System (AIS). AIS is a computational method inspired by the biology immune system. It is progressing slowly and steadily as a new branch of computational intelligence and soft computing [1], [2]. It has been used in several applications such as machine learning, pattern recognition, computer virus detection, anomaly detection, optimization and robotics [2]. Various algorithms have been developed based on the functions and mechanisms of immune systems. One AIS-based algorithm is the Artificial Immune Recognition System (AIRS). AIRS is a supervised immune-inspired classification system capable of assigning data items unseen during training to one of any number of classes based on previous training experience. AIRS is probably the first and best known AIS for classification, having been developed in 2001 [3].

AIRS has four main steps: Initialization, ARB generation, Competition for resources and nomination of candidate memory cell, and finally promotion of candidate memory cell into memory pool. The heart of AIRS is the resource competition phase that maintains better populations in the system [4]. This resource competition phase is based on the results of the resource allocation method. Classic AIRS uses linear method for resource allocation. Polat et al. [5] have proposed a nonlinear resource allocation method for AIRS. They have used fuzzy membership to implement nonlinearity and applied AIRS with fuzzy resource allocation to medical datasets; but we couldn't find work that has been studied and evaluated the effect of using fuzzy resource allocation method on AIRS accuracy from a statistical point of view.

In this study, we compare the accuracies of two algorithms: AIRS and AIRS with fuzzy resource allocation using statistical tests. Each algorithm runs ten times for each dataset. In each run, the ten-fold cross validation method is used to estimate the accuracy of algorithm; and after all runs, we use a two-way paired t-test to find the statistically significant difference between the accuracies of these two algorithms for related datasets. We also use twelve datasets from the UCI machine learning repository. The experiments show that using the fuzzy resource allocation method improves the accuracy of AIRS in seven datasets and for five datasets the increase in accuracy is significant.

The following section introduces the AIRS algorithm briefly. Section 3 describes the fuzzy resource allocation method and finally Section 4 illustrates evaluation method, experiments and results.

2. AIRS

Artificial Immune Recognition System (AIRS) is investigated by Watkins [3]. AIRS can be applied to classification problems, which is a very common real world data mining task. To show the capability of AIS to do the classification was the initial objective of developing AIRS, but results show that AIRS is comparable to other popular classifiers. Before AIRS, most artificial immune system research focus on unsupervised learning and clustering. One attempt to use immune systems for supervised learning was the work of Carter [6]. AIRS uses several concepts of the immune system -- including resource competition, clonal selection, affinity maturation, memory cell production and also the resource limited artificial immune system concept investigated by [7]. In fact, AIRS is a hybrid algorithm that uses concepts of different immune system theories

Feature vectors (labeled data) presented for training and test are named as antigen, and the system units are called as ARBs (Artificial Recognition Balls) or B cells. In theory, similar B cells are represented with Artificial Recognition Balls (ARBs) and these ARBs compete with each other for a fixed number of B cells. AIRS adopts these concepts. In AIRS, ARB and the B cell are the same and ARBs compete for a fixed number of resources. The algorithm generates new instances as memory cells that are used in the classification task finally. Memory cells are best ARBs. These ARBs have highest affinities to training antigens and generated based on the immune metaphors. Table 1 summarizes the mapping between the immune system and AIRS concepts.

AIRS has four stages: The first stage is performed once at the beginning of the algorithm. This stage includes normalization and initialization. Other stages are performed for each antigen in the training set. These stages are: ARB generation, resource competition and lastly, insertion candidate memory cell into memory pool.

In the first step, AIRS performs normalization for all training data. This normalization puts the distances between two data in the [0,1] interval. After normalization, algorithm initializes the memory cell pool and the ARB pool from randomly selected training data. Now, the algorithm is ready to generate memory cells. The memory cell generation mechanism for each antigen is as follows [3], [4], [8]:

1. A training antigen is compared with all the memory cells in the memory cell pool that have the same class as the antigen. The memory cell most stimulated by the antigen is selected and cloned. The memory cell and all generated clones are put into the ARB pool. The number of generated clones depends on the affinity between the memory cell and antigen. This affinity is determined by Euclidean distance between the feature vectors of the memory cell and the training antigen. The smaller Euclidean distance means the higher affinity and generating more number of clones.

2. In the next step, the training antigen is presented to all the ARBs in the ARB pool. All the ARBs are rewarded based on the affinity between the ARB and the antigen. If the ARB and antigen belong in the same class, the ARB is rewarded highly for high affinity with the antigen; otherwise, the ARB is rewarded highly for a low value of affinity measure. The rewards are in the form of number of resources (resource allocation). More rewards cause more resources. When the number of resources are calculated for all ARBs, the sum of allocated resources in the system typically exceeds the maximum number allowed for the system and the excess number of resources held by the ARBs must be removed from the system. The algorithm finds the ARB with lowest resources and removes its resources and repeats this task until the allocated resources do not exceed the number of resources allowed. Then, ARBs with zero resources are removed from the ARB pool. This procedure is named resource competition. The remaining ARBs are tested for their affinities towards the training antigen. If for any class, the ARBs do not meet a user defined stimulation threshold, then the ARBs are mutated and cloned again. This step is repeated until the affinity for all classes meet the stimulation threshold.

3. After all the classes have passed the stimulation threshold, the highest affinity ARB of the same class as the antigen is chosen as a candidate memory cell. If its affinity for the training antigen is greater than the affinity of the original memory cell selected for cloning at step 1, then the candidate memory cell is placed in the memory cell pool; and if the difference in affinity of these two memory cells is smaller than a user defined threshold, the original memory cell is removed from the memory cell pool.

These steps are repeated for each training antigen. After the completion of training, memory cells are used for classification. The *K*-nearest neighbor method is used to classify test data. The class of a test data is determined by majority voting among the *k* most stimulated memory cells.

AIRS has shown that is comparable with famous classifiers [3] and many researches have been done to improve and evaluate the performance of AIRS [4], [8], [9], [10]. The results show that AIRS is comparable with famous and powerful classifiers.

3. FUZZY RESOURCE ALLOCATION

Resource competition is one stage of AIRS. The purpose of resource competition in AIRS is improving the selection probability of high-affinity ARBs for next steps. Resource competition mechanism was described in step 2 of section 2. In this section, our focus is the resource allocation. Resource competition is done based on the ARBs' rewards. Reward of each ARB is the number of resources that could be allocated for it. The number of allocated resources for each ARB is calculated by multiplying the stimulation rate with the clonal rate as shown in (1). Clonal rate is a user-defined parameter and the stimulation rate shows the affinity between the ARB and antigen. Mervah et al [8] have used a different resource allocation mechanism. In their mechanism, the classes occurring more frequently data get more resources.

$$Rsources = StimulationRate \times ClonalRate \quad (1)$$

Polat et al. [5] have used a nonlinear method for resource allocation. Based on their hypothesis, there is a linear relation between stimulation rate and resource allocation and this linearity causes to use more resources in the system, long classification time and also more memory cells. They use nonlinear resource allocation to solve these problems and fuzzy logic to do nonlinear resource allocation. Figures 1, 2 and 3 show fuzzy memberships, rule-base for fuzzy resource allocation and Linguistic values for the input and output membership functions as used in the related researches respectively [5]. These membership functions and linguistic values have allocate less number of resources for ARBs with stimulation values between 0 and 0.50 and more for ARBs with stimulation values between 0.50 and 1. More detailed information about the proposed fuzzy membership functions could be found in [5].

There is no hypothesis about the relation between using nonlinear resource allocation method and classification accuracy in described research. What is the effect of using nonlinear resource allocation on AIRS accuracy? We study the effect of using fuzzy resource allocation methods on accuracy of AIRS. The aim of this study is applying proposed fuzzy resource allocation methods on AIRS and comparing the accuracy of AIRS and AIRS with fuzzy resource allocation statistically.

4. EXPERIMENTS AND RESULTS

Experiments were carried out in order to determine how AIRS with fuzzy resource allocation performed compared to AIRS. The WEKA version of AIRS [11] was used to incorporate fuzzy resource allocation in AIRS. Both the AIRS and AIRS with fuzzy resource allocation were run with the default parameters of code to have as fair as possible comparison between two algorithms. The values of parameters are shown in table 2.

Twelve datasets were retrieved from the well-known UCI machine learning repository [12]. UCI machine learning repository is most famous repository to test the classification algorithms. We chose datasets with varying number of attributes, instances and classes from difficult real world learning problems to cover the complete characteristics of data. Because the WEKA code of AIRS supports both continuous and discrete attributes, we could choose datasets with discrete attributes and we didn't have the limitation on dataset selection. Table 3 shows the used datasets with some characteristics of them.

The combination of ten-fold cross validation approach and the statistical t test were used to compare the mean accuracies of algorithms for each dataset. Dietterich [13] has done comprehensive study to find the most reliable statistical tests for comparing supervised classification learning algorithms; his study shows that cross-validated t test is the most powerful method to detect the differences between algorithms when differences really do exist.

N-fold cross validation is an approach to estimate the predictive accuracy of the classification algorithms. In this approach, the instances are randomly divided to N equal subsets. Each instance is put in one subset. At each iteration, N-1 subsets are merged to form the training set and the classification accuracy of the algorithm is measured on the remaining subset. This process is repeated N times, choosing a different subset as the test set each time. Therefore, all data instances have been used N-1 times for training and once for testing.

The final predictive accuracy is computed over all folds in the usual manner but dividing the number of correct classifications taken over all folds by the number of data instances in all folds. Some theoretical issue and also several tests on numerous different data sets by using different classification algorithms have shown that ten-fold cross validation gets best estimate of accuracy [14], [15]; therefore we used ten-fold cross validation method in experiments. Also as there are some randomness in AIRS and ten-fold cross validation, both algorithms were run ten times on each dataset to achieve more reliable accuracies. The average of the accuracies of ten runs was considered as final accuracy. In each run for one data set firstly dataset was divided to ten subsets, and then both algorithms were run respectively on dataset. This means that we did not do dataset dividing for each algorithm separately.

Two-way paired t test was performed to compare the mean accuracy of both algorithms. In this study, the null hypothesis (H0) is tested against the alternative hypothesis (H1).

H0: Mean-Accuracy (AIRS) = Mean-Accuracy (AIRS with fuzzy resource allocation)

H1: Mean-Accuracy (AIRS) != Mean-Accuracy (AIRS with fuzzy resource allocation)

The P-value of the test could be used the make the decision about the existing significant difference between the accuracies of two algorithms. P-value indicates the probability of obtaining the existing sample data given the null hypothesis. Low P-value leads to the rejection of the null hypothesis and acceptance of alternative hypothesis. The commonly used level of significance 0.05 is applied in this study. P-value under 0.05 rejects the null hypothesis and shows the significant difference between accuracies. Table 4 shows the accuracies and P-values are achieved by algorithms for datasets.

The results show that using fuzzy resource allocation increases the accuracy of AIRS in seven datasets; but only in four of them achieves significant improvement. On other hand, AIRS achieves higher accuracy in four datasets that two of them are significant. Using fuzzy resource allocation increases the accuracy of algorithm in all datasets that have combination of attribute types. The increase is significant in two of these three datasets, German and Credit-Crx. Both datasets are used in credit approval application and the p-value is very low for them. All attributes of balance and zoo datasets are nominal and using fuzzy resource allocation cause to decrease the accuracy in balance dataset. The accuracy decrease is significant in this dataset. Both algorithms achieved equal accuracy in zoo dataset.

For the datasets that contain only numeric attributes, using fuzzy resource allocation increases the accuracy in four of seven datasets; but increase is not significant for majority of these datasets. The increase is significant in three datasets only. Regarding to these datasets, i.e. datasets with numeric data attributes, AIRS archives higher accuracy in three datasets that one of them is significant. As main result it can be concluded that the fuzzy resource allocation increases the accuracy of AIRS in some datasets and decreases the accuracy of AIRS in other cases. In the described experiment, fuzzy resource allocation increases the accuracy of AIRS in some datasets, but puts the significant effect in the minority of datasets. It seems that the direction of changes, increase or decrease, in accuracy and also existence the significant difference between the accuracy of algorithms are depending on the type, value, distribution and nature of data. Investigate the more accurate fuzzy memberships, fuzzy rules and fuzzy values may be increase the accuracy in more datasets.

5. CONCLUSIONS

In this paper, we studied the effect of using fuzzy resource allocation method on AIRS accuracy from statistical point of view. The combination of ten-fold cross validation and statistical t test was used as evaluation method. Both algorithms, AIRS and AIRS with fuzzy resource allocation, were tested on some benchmark datasets with different characteristics. The results showed that the using fuzzy resource allocation causes significant improvement on the accuracy of AIRS in the minority of datasets.

REFRENCES

[1] de Castro, L.N., Timmis, J.: Artificial Immune Systems as a novel Soft Computing Paradigm. Soft Computing Journal 7(7) (2003)

[2] de Castro, L.N., Timmis, J.: Artificial Immune Systems: A New Computational Intelligence Approach. Springer, Heidelberg (2002)

[3] Watkins, A.: AIRS: A Resource Limited Artificial Immune Classifier. M.S. thesis,

Department of Computer Science. Mississippi State University (2001)

[4] Watkins, A., Timmis, J., Boggess, L.: Artificial Immune Recognition System (AIRS): An Immune-Inspired Supervised Learning Algorithm. Genetic Programming and Evolvable Machines 5(3), 291–317 (2004)

[5] Polat, K., Sahan, S., Kodaz, H., Gunes, S.: Breast cancer and liver disorders classification using artificial immune recognition system (AIRS) with performance evaluation by fuzzy resource allocation mechanism. Expert Systems with Applications 32, 172–183 (2007)

[6] Carter, J.H.: The immune systems as a model for pattern recognition and classification. Journal of the American Medical Informatics Association 7(1), 28–41 (2000)

[7] Timmis, J., Neal, M.: A Resource Limited Artificial Immune System. Knowledge Based Systems 14(3), 121–130 (2001)

[8] Marwah, G., Boggess, L.: Artificial immune systems for classification: Some issues. In: Proceedings of the first international conference on artificial immune systems, University of Kent Canterbury, England, pp. 149–153 (2002)

[9] Goodman, D. E., Boggess, L., Watkins, A.: Artificial immune system classification of multiple-class problems. In: Proceedings of the artificial neural networks in engineering (ANNIE'02), pp. 179–183 (2002)

[10] Goodman, D. E., Boggess, L., Watkins, A.: An investigation into the source of power for AIRS, an artificial immune classification system. In: Proceedings of the international joint conference on neural networks (IJCNN'03), Portland, Oregon, pp. 1678– 1683 (2003)

[11]Brownlee, J.: Artificial Immune Recognition System (AIRS) - A Review and Analysis. Tech Rep, Centre for Intelligent Systems and Complex Processes (CISCP), Faculty of Information and Communication Technologies (ICT). Swinburne University of Technology (2005)

[12]Newman, D. J, et al.: UCI Repository of machine learning databases. Retrieved July 2008from:http://www.ics.uci.edu/~mlearn/ML Repository.html. (1998)

[13] Dietterich, T.G.: Approximate statistical tests for comparing supervised classification learning algorithms. Neural Computation 10(7), 1895–1923 (1998)

[14] Witten, H., Frank, E.: Data Mining: Practical Machine Learning Tools and Techniques, 2nd ed. Morgan Kaufmann, San Francisco (2005)

[15] Han, j., Kamber, M.: Data Mining: Concepts and Techniques, 2nd ed. Morgan Kaufmann (2006)

Table1. Mapping between Immune System and AIRS

Immune System	**AIRS**
Antibody	Feature vector
Recognition ball	Combination of feature vector and vector class
Shape-space	Type and possible values of the data vector
Clonal expansion	Reproduction of ARBs that are well matched antigens
Antigens	Training data
Affinity Maturation	Random mutation of ARB and removal of the least stimulated ARBs
Immune memory	Memory set of mutated ARBs
Metadynamics	Continual removal and creation of ARBs and memory cells

Table2. Algorithm Parameters

Used Parameter	**Value**
Clonal rate	10
Mutation rate	0.1
ATS	0.2
Stimulation threshold	0.9
Resources	150
Hypermutation rate	2
K value in KNN classifier	3
Seed	1

Table3. Datasets

Dataset	# of Instances	# of Attributes	# of Classes	Numeric Attributes	Categorical Attributes	Missed Value
Balance-Scale	625	4	3	0	4	No
Breast-Cancer	683	10	2	10	0	Yes
Credit-Crx	690	15	2	6	9	Yes
German	1000	20	2	7	13	N/A
Glass	214	9	7	9	0	No
Hepatitis	155	19	2	6	13	Yes
Image-Segment	210	19	7	19	0	Yes
Ionosphere	351	34	2	34	0	No
Iris	150	4	3	4	0	No
Pima-Diabetes	768	8	2	8	0	No
Wine	178	13	3	13	0	No
Zoo	101	16	7	0	16	No

Table4. Comparison of classification accuracy

Dataset	AIRS (%)	Fuzzy-AIRS (%)	P-Value
Balance-Scale	**84.42704**	83.29749	**0.021014**
Breast-Cancer	**96.8376**	96.73444	0.50288
Credit-Crx	82.57971	**84.52174**	**6.13277E-06**
German	68.82	**70.76**	**0.000276**
Glass	59.44589	**61.54762**	0.093862
Hepatitis	82	**83.225**	0.052821
Image-Segment	80.09524	**83.61905**	**0.000113**
Ionosphere	**86.50714**	84.58377	**3.13663E-06**
Iris	**95.53333**	95.26667	0.566316
Pima-Diabetes	70.06169	**71.76008**	**0.003692**
Wine	95.06209	**96.24837**	**0.007181**
Zoo	94.96364	94.96364	1

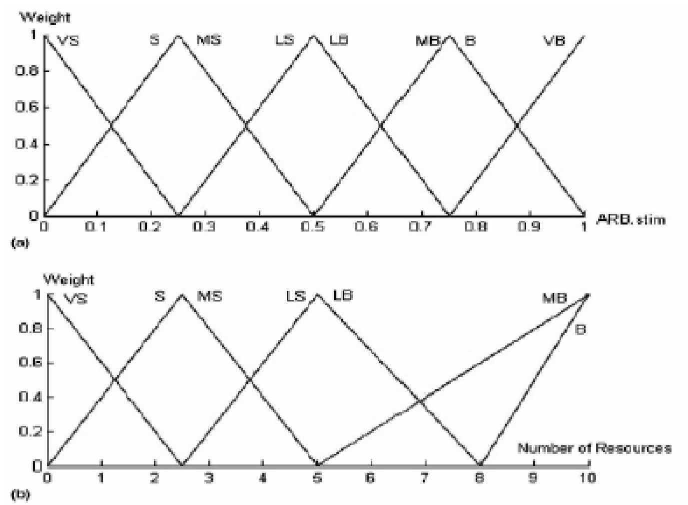

Figure1. (a) Input membership function and (b) Output membership function [5]

```
if   ARB.stim = VS and S
     then Output = (VS'+S')/2

if   ARB.stim = S and MS
     then Output = (S'+MS')/2

if   ARB.stim = MS and LS
     then Output = (MS'+LS')/2

if   ARB.stim = LS and LB
     then Output = (LS'+LB')/2

if   ARB.stim = LB and MB
     then Output = (LB'+MB)/2

if   ARB.stim = MB and B
     then Output = (MB'+B')/2

if   ARB.stim = B and VB
     then Output = (B'+VB')/2
```

Figure2. Rule base for fuzzy resource allocation [5]

Input	Output
VS—Very Small	VS'—Very Small
S—Small	S'—Small
MS—Middle Small	MS'—Middle Small
LS—Little Small	LS'—Little Small
LB—Little Big	LB'—Little Big
MB—Middle Big	MB'—Middle Big
B—Big	B'—Big
VB—Very Big	

Figure3. Linguistic values for input and output membership functions [5]