

ARTIFICIAL IMMUNE SYSTEM BASED ON REAL VALUED NEGATIVE  
SELECTION ALGORITHMS FOR ANOMALY DETECTION

RIHAB SALAH KHAIRI

A dissertation submitted in Partial  
fulfillment of the requirement for the award of the  
Degree of Master of Computer Science (Soft Computing)

Faculty of Computer Science  
Universiti Tun Hussein Onn Malaysia

FEBRUARY 2015

## ABSTRACT

The Real-Valued Negative Selection Algorithms, which are the focal point of this research, generate their detector sets based on the points of self data. Self data are regarded as the normal behavioral pattern of the monitored system. In this research, the Real-Valued Negative Selection with fixed-sized detectors (RNSA) and Real-Valued Negative Selection with variable-sized detectors (V-Detector) were applied for classification and detection of anomalies. The issue of integrity and confidentiality of data have been in existence for decades. Data have been tampered and altered either by a computer user or unauthorized access via hacking. In this research, the Negative Selection Algorithms were deployed. On the contrary, the experiments with various and well-known datasets show that NSAs have great flexibility to balance between efficiency and robustness and to accommodate domain-oriented elements in the method. Classifier algorithms, namely the Support Vector Machine and K-Nearest Neighbours were used for benchmarking the performance of the Real-Valued Negative Selection Algorithms. Experimental results illustrate that RNSA and V-Detector algorithms are suitable for the detection of anomalies, with SVM and KNN producing significant efficiency rates and increase in execution time. The results shown in this study illustrate the effectiveness of the anomaly detection techniques on Iris, Balance-Scale, Lenses and Hayes-Roth datasets. On the whole, the RNSA and V-Detector outperformed SVM and KNN on all datasets by producing higher detection rates, lower false alarm rates and execution times. This shows that the Negative Selection Algorithms are equipped with the capabilities of detecting changes in data, thus appropriate for anomaly detection. With respect to all the algorithms, V-Detector proved to be superior and surpassed all other algorithms based on performance and execution time.

## ABSTRAK

Algoritma Pemilihan Nilai Nyata Negatif yang menjadi tumpuan penyelidikan ini menghasilkan set pengesannya berdasarkan kepada titik data sendiri. Data sendiri dianggap sebagai corak tingkah laku normal bagi sistem yang dipantau. Dalam penyelidikan ini, Pemilihan Nilai Nyata Negatif dengan pengesan bersaiz tetap (RNSA) dan Pemilihan Nilai Nyata Negatif dengan pengesan bersaiz boleh ubah (Penges-an-V) digunakan untuk pengelasan dan pengesanan anomali. Isu integriti dan kerahsiaan data telah wujud selama beberapa dekad. Data telah terusik dan diubah sama ada oleh pengguna komputer, atau capaian yang tidak dibenarkan melalui penggodaman. Dan dalam penyelidikan ini, algoritma pemilihan negatif telah digunakan. Sebaliknya, eksperimen dengan data yang pelbagai dan terkenal menunjukkan bahawa NSA mempunyai daya fleksibiliti untuk mengimbangi antara kecekapan dan keteguhan dan untuk menampung unsur berorientasikan domain dalam kaedahnya. Algoritma pengelasan, iaitu Mesin Vektor Sokongan dan K-Jiran Terdekat digunakan untuk penandaarasan prestasi algoritma pemilihan nilai nyata negatif. Hasil uji kaji menunjukkan bahawa algoritma RNSA dan Penges-an-V sesuai untuk pengesanan anomali, dengan SVM dan KNN yang menghasilkan kadar kecekapan ketara dan peningkatan dalam masa pelaksanaan. Keputusan yang ditunjukkan dalam kajian ini menggambarkan keberkesanan teknik pengesanan anomali pada set data Iris, Balance-Scale, Lenses dan Hayes-Roth. Pada keseluruhannya, RNSA dan Penges-an-V mengatasi SVM dan KNN pada semua set data dengan menghasilkan kadar pengesanan yang lebih tinggi, kadar isyarat palsu dan waktu pelaksanaan lebih rendah. Ini menunjukkan bahawa algoritma pemilihan negatif dilengkapi dengan keupayaan mengesan perubahan dalam data, dengan itu sesuai untuk pengesanan anomali. Berkenaan dengan semua algoritma, Penges-an-V terbukti unggul dan melepasi semua algoritma lain berdasarkan prestasi dan masa pelaksanaan.

## TABLE OF CONTENTS

<b>DECLARATION</b>	<b>ii</b>
<b>DEDICATION</b>	<b>iii</b>
<b>ACKNOWLEDGEMENT</b>	<b>iv</b>
<b>ABSTRACT</b>	<b>v</b>
<b>ABSTRAK</b>	<b>vi</b>
<b>TABLE OF CONTENTS</b>	<b>vii</b>
<b>LIST OF TABLES</b>	<b>x</b>
<b>LIST OF FIGURES</b>	<b>xi</b>
<b>LIST OF SYMBOLS AND ABBREVIATIONS</b>	<b>xiii</b>
<b>CHAPTER 1 INTRODUCTION</b>	<b>1</b>
1.1 Overview	1
1.2 Problem Statements	2
1.3 Aim of study	4
1.4 Objectives of the Study	4
1.5 Scope of Study	5
1.6 Significance of the Study	5
1.7 Thesis Organization	5
1.8 Chapter Summary	6
<b>CHAPTER 2 LITERATURE REVIEW</b>	<b>7</b>
2.1 Introduction	7

2.2	Transition from Biological Immune System to Artificial Immune System	8
2.3	Anomaly Detection	11
2.4	Anomaly Detection Technique	12
2.4.1	Negative Selection Algorithm (NSA)	13
2.4.1.1	String (or Binary)-based Negative Selection Algorithm	13
2.4.1.2	Real Value Negative Selection Algorithm (RNSA)	15
2.4.1.3	V- Detector Negative Selection Algorithm	17
2.4.2	Support Vector Machine (SVM)	19
2.4.3	K- nearest neighbor (KNN)	21
2.5	Application of AIS in Anomaly Detection	22
2.6	Chapter Summary	26

## **CHAPTER 3 RESEARCH METHODOLOGY** **27**

3.1	Introduction	27
3.2	Data Selection	28
3.2.1	Iris Dataset	28
3.2.2	Balance-Scale Dataset	29
3.2.3	Lenses Dataset	30
3.2.4	Hayes-Roth Dataset	31
3.3	Data Pre-processing	32
3.4	Data partition	33
3.5	Training and Testing Data with the Detector	34
3.6	AIS Algorithm	35
3.6.1	Real Valued Negative Selection Algorithm (RNSA)	36
3.6.2	V-Detector Real –Valued Negative Selection Algorithm	36
3.6.3	SVM Algorithm Parameter Description	37
3.6.3.1	Convergence Tolerance Value	37
3.6.3.2	Complexity Factor	37

3.6.3.3	Value Kernel Cache Size	38
3.6.4	KNN Algorithm Parameter Description	38
3.7	Performance Evaluation	39
3.8	Chapter Summary	40
<b>CHAPTER 4</b>	<b>SIMULATION RESULTS AND ANALYSIS</b>	<b>41</b>
4.1	Introduction	41
4.2	Experimental Design	41
4.3	The Performance of Real-Valued Negative Selection Algorithms	42
4.4	The Experimental Assessment	42
4.5	Experimental Assessment	47
4.6	The Experimental Assessment of SVM	47
4.7	The Experimental Assessment of KNN	49
4.8	Performance Comparison between All Algorithms	52
4.9	Chapter Summary	54
<b>CHAPTER 5</b>	<b>CONCLUSIONS AND FUTURE WORKS</b>	<b>55</b>
5.1	Introduction	55
5.2	Research Contribution	55
5.2.1	The Construction Model of AIS	56
5.2.2	Anomaly Detection by using the AIS Model with RNSA and V-Detector	56
5.2.3	Performance Evaluation of the RNSA and V-Detector Models and Benchmarked Algorithms	57
5.3	Recommendation and Future Works	58
5.4	Chapter Summary	58
	<b>REFERENCES</b>	<b>59</b>
	<b>VITAE</b>	<b>67</b>

## LIST OF TABLES

2.1	A time-line of AIS (1986 to 1999)	22
2.2	A time-line of AIS (2000 to 2003)	23
3.1	Summary of the Iris Dataset	28
3.2	Iris Dataset's Description	29
3.3	Class Information of Iris Dataset	29
3.4	Summary of the Balance-Scale Dataset	29
3.5	Balance-Scale Dataset's Description	30
3.6	Class Information of Balance-Scale Dataset	30
3.7	Summary of the Lenses Dataset	30
3.8	Lenses Dataset's Description	31
3.9	Class Information of Lenses Dataset	31
3.10	Summary of the Hayes-Roth Dataset	31
3.11	Hayes-Roth Dataset's Description	32
3.12	Class Information of Hayes-Roth Dataset	32
3.13	Parameter value for RNSA	36
3.14	Parameter value for V-Detector	37
3.15	Parameter value for SVM	38
3.16	Parameter value for KNN	39
3.17	Confusion Matrix Defines Four Possible Scenarios when Classifying Class "AB"	40
4.1	The Performance of Four Dataset on NSA on Different of Number Radius	46
4.2	The Performance of SVM with Complexity Factor on Four Dataset	49
4.3	The Performance of Four Dataset on KNN on Different number folder option $K$	51

## LIST OF FIGURES

2.1	Partitioning an Antigen and the Proliferation of the B-Cell into a Plasma Cell	8
2.2	Multi-layer Structure of the Immune System	9
2.3	Censoring Stage	14
2.4	Monitoring Stage	15
2.5	RNSA Pseudo-code	16
2.6	The Iterative Process of Real-Valued Negative Selection Algorithm	17
2.7	Comparison of Detector Coverage for Different Detector Schemes	18
2.8	Real-Valued Negative Selection V-Detector Algorithm Pseudo-code	19
2.9	Linear Support Vector Machine	20
3.1	Research Framework	27
3.2	A model How to Construct a Detector Model	34
3.3	Unknown Data is Classified Using a Pre-generated Detector Model	35
4.1	The effect of radius on the Detection Rate for V-Detector	43
4.2	The effect of radius on the Detection Rate for RNSA	43
4.3	The effect of radius on the false alarm rate for V-Detector	44
4.4	The effect of radius on the False Alarm Rate for RNSA	44
4.5	The effect of radius on the CPU time for V-Detector	45

4.6	The effect of radius on the CPU time for RNSA	45
4.7	The effect of complexity of factor on the Detection Rate for SVM algorithm	47
4.8	The effect of complexity factor on the False Alarm Rate for SVM algorithm	48
4.9	The effect of complexity factor on the CPU time for SVM algorithm	48
4.10	The effect of number folder option $K$ on the Detection Rate for KNN algorithm	50
4.11	The effect of number folder option $K$ on the False Alarm Rate (FR) for KNN algorithm	50
4.12	The Effect of number folder option $K$ on the CPU time (T) for KNN algorithm	51
4.13	The Detection Rate of Different Algorithms on Four Datasets	52
4.14	The (FR) of Different Algorithms on Four Datasets	53
4.15	The CPU time of Different Algorithms on Four Datasets	53



## LIST OF SYMBOLS AND ABBREVIATIONS

AIS	-	Artificial Immune Systems
ALC	-	Artificial lymphocytes
RNSA	-	real-valued negative selection algorithm with fixed detector
V - Detector	-	real-valued negative selection with variable detector
HIS	-	Human Immune System
SVM	-	Support Vector Machine
KNN	-	K-Nearest Neighbours
NIS	-	Natural Immune System
NSA	-	Negative Selection Algorithm
$S$	-	Self
$R_0$	-	set of competent detectors
$L$	-	length strings
$R$	-	set of competent detectors
RNS	-	Real-Valued Negative Selection
$r$	-	radius of detection
$\eta$	-	adaptation rate
$t$	-	once a detector reaches this age it will be considered to be mature
$\tau$	-	decay rate
$d$	-	Detector
$r_s$	-	self radius
$c_0$	-	estimated coverage
$D_{max}$	-	maximum number of detectors
$r_d$	-	variable radius
$M_{max}$	-	limit of the counter
$H$	-	Hyperplanes

$F(x)$	-	Margin
$y$	-	Output
$x$	-	training or test pattern
$w$	-	weight vector
$b$	-	Bias
$w_i x_i$	-	vector components
UCI	-	Machine Learning Repository
KEEL	-	Knowledge Extraction based on Evolutionary Learning
$x_i$	-	initial value of vector
$Z_i$	-	normalized data
SOM	-	Sequential Minimal Optimization
$C$	-	complexity factor
WEKA	-	Waikato Environment for Knowledge
$k$	-	Number folder MI option
NNge	-	Non--Nested generalized exemplars
DR	-	Detection Rate
FAR	-	False Alarm Rate
$A$	-	Actual normal class
TP	-	True Positives
FN	-	False Negatives
FP	-	False Positives
TN	-	True Negatives
$AB$	-	Actual anomalies class
CPU	-	Central Processing
MATLAB	-	Matrix Laboratory
IEEE	-	Institute of Electrical and Electronics Engineering

## CHAPTER 1

### INTRODUCTION

#### 1.1 Overview

A record is said to be anomalous or outlying if its behavior does not conform to the behavior of the majority of the dataset. There has been an increasing interest because its presence could indicate unauthorized usage of the system, a failure in part of the system or a diagnosis of a disease. Many domains require anomaly detection systems. One of them is anomaly detection, in which a user can try to gain extra privileges of the system or an unauthorized user can try to gain access to the system (Amer, 2011).

Additionally, anomaly detection is essentially a rare event problem that has been called and used interchangeably with various terms such as deviation detection, outlier analysis, fraud detection, exception mining, mining rare classes, and others. Similarly, a very minute fraction of the total transaction constitutes the anomalous activities executed (Tuo *et al.*, 2004).

Thus, anomaly detection can be applied in many applications, where there are different costs for false positives and false negatives that are similar to different datasets for anomaly detection. There are biased and unsatisfactory results, especially with different costs of false positives and false negatives. Under these conditions, it is important to train the classification model under a cost sensitive procedure and also to evaluate the classifier performance more accurately (Gadi *et al.*, 2008).

Over the last few years, more intelligent decision making techniques have been inspired by nature, e.g. evolutionary algorithms, ant colony optimization and simulated annealing. More recently, a novel computational intelligence technique inspired by immunology has emerged and called Artificial Immune Systems (AIS). The immune system is complex and powerful. It is characterized by special features

such as noise tolerance, robustness, diversity, reinforcement learning memory, dynamic, distributed, multilayered and adaptive. These features give the immune system a great advantage in detecting lots of different types of pathogens, both known and unknown, and also destroying pathogens. The techniques inspired by the immune system have already been useful in solving some computational problems (Dasgupta *et al.*, 2011).

AIS can be seen as a pattern recognition system and consists of artificial lymphocytes (ALC) that can classify any pattern either as part of a predetermined set of patterns or otherwise, and in training, only positive examples are required. As such, AIS lends itself to application of fraud detection problems (Graaff *et al.*, 2011). The definition of AIS by Sridevi *et al.* (2012) as adaptive systems is inspired by theoretical immunology and observed immune functions, principles and models, which are applied for problem solving, which certifies its suitability for application in this study.

AIS with two types of Negative Selection Algorithms using real-valued coordinates, namely Real-Valued Negative Selection Algorithm (RNSA) with fixed detector and Real-Valued Negative Selection with variable detector (V-Detector) are the focus of this study. The immune system metaphors relevant to AIS are described in detail in Chapter 2. Furthermore, there is a comparison with other classification algorithms such as Support Vector Machine (SVM) and K-Nearest Neighbors (KNN). The real-valued Negative Selection Algorithms and the benchmarked algorithms are tested and applied to various dataset, with the step by step algorithm walk-through is revealed.

## 1.2 Problem Statement

In a modern day life, anomaly is one of the major causes of great losses (Graaff *et al.*, 2011). Anomalies are patterns in data that do not conform to a well-defined notion of normal behavior. They may be induced in the data for various reasons, such as malicious activity, for example, credit card fraud, cyber-intrusion, terrorist activity or breakdown of a system, but all of the reasons have the common characteristics that make them interesting to the analysts. Many types of anomaly exist in different application fields, namely point anomalies, contextual anomalies, and collective anomalies. Different anomaly detection techniques have been developed and

proposed in handling issues related with keeping the integrity of data, and have been applied to fault tolerance, robotic control, network intrusion detection, and bioinformatics.

In general, the problem of anomaly detection can be seen as a two or more class classification problem. Given an element from a given problem space, the system should classify it as normal or abnormal. However, this is a very general characterization since it can correspond to very different problems depending on the specific context where it is interpreted. Therefore, from a statistical point of view, the problem can be seen as that of outlier detection, which is referred to as an observation deviating from other observations and triggering uncertainty as to how it was generated (Gonzalez & Dasgupta, 2003).

Since the 19<sup>th</sup> century, various attempts have been made to resolve the problems of anomalies. Many modern techniques exist in literature that are based on Artificial Intelligence, Neural Network, Bayesian Network, Fuzzy logic, K-Nearest Neighbor Algorithm, Support Vector Machine, Decision Tree, Fuzzy Logic Based System, Machine learning, Sequence Alignment, Genetic Programming and others, which have evolved in detecting various anomaly (Tripathi *et al.*, 2013).

The Negative Selection Algorithm is equipped with properties that make it suitable for use in detecting anomalies. The properties are as follows: (1) No prior knowledge of intrusions is necessary, (2) Detection is probabilistic but tunable, meaning a complete repertoire of detectors will not be generated (i.e. a set of detectors that covers all possible non-self strings). Instead, it is contented with matching all but a small fraction of non-self strings in exchange for a smaller set of detectors, (3) The detection scheme is inherently distributable, (4) The set of detectors at each site can be unique. This means that if one site is compromised, others would still be protected, and (5) The set of self strings and the detector set are mutually protective, meaning that the detector set protects the self set against change and vice versa (D'haeseleer *et al.*, 1997).

The field of Artificial Immune Systems which began in the early 1990s serves as alternative and efficient algorithms for detecting anomalies to the already existing methods. The immune system shows computational strength from different aspects of problem solving. In defining Artificial Immune System, there seems to be no fundamental definition, and many others are focusing on anomaly detection (Garrett, 2005).

They were inspired by the Human Immune System (HIS) which is robust, decentralized, error tolerant, and adaptive in nature. Also, the Artificial Immune System appears to be precisely tuned to the problem of detecting and eliminating infections (Tuo *et al.*, 2004). There are a number of AIS models used in pattern recognition, fault detection, computer security, and a variety of other applications in the field of science and engineering. Most existing AIS algorithms imitate one of the following mechanisms of the immune system: negative selection, dendritic cell, immune network, or clonal selection. These models emphasize on designing and applying computational algorithms and techniques using simplified models of various immunological processes and functionalities (Aziz *et al.*, 2012). Negative Selection-based Algorithm (Dasgupta *et al.*, 2011) has potential applications in various areas, in particular anomaly detection.

Considering all the above advantages of AIS, this work attempts to classify different datasets by using AIS with Negative Selection Algorithms, namely the RNSA and V-Detector, which are data-driven models to alleviate the difficulty in anomaly detection in various datasets.

### **1.3 Aim of the Study**

The aim of this study is to apply AIS with Real-Valued Negative Selection Algorithms for anomaly detection problem through classification of four datasets.

### **1.4 Objectives of the Study**

In order to achieve the above mentioned research aim, several objectives have been set, which are listed below:

- i. To apply a detector model based on Artificial Immune System (AIS) with Negative Selection Algorithms; the RNSA and V-Detector.
- ii. To classify the datasets using the algorithmic models in (i).
- iii. To compare and evaluate the performance of the AIS algorithms with other classifiers, namely the SVM and KNN based on Detection Rate, False Alarm Rate, and CPU Time.

## 1.5 Scope of the Study

This research focuses on the use of two kinds of Real-Valued Negative Selection Algorithms, namely the RNSA and V-Detector. For benchmarked purpose, two popularly known classification algorithms; Support Vector Machine and K-Nearest Neighbors have been adopted in this study. Three datasets were retrieved from the UCI Machine Learning Repository, which are the Iris data consisting of three classes of *Iris Setosa*, *Iris Versicolour* and *Iris Virginica* instances; and the Balance-Scale data constituting three classes of Left, Balance and Right; and the Lenses constituting three classes, which are the patient that should be fitted with hard contact, soft contact, and no contact lenses. Also, one dataset was retrieved from KEEL Knowledge Extraction based on Evolutionary Learning, which is Hayes-Roth consisting of three classes (1, 2, 3). As a measure of performance for comparison between the RNSA algorithms and benchmarked algorithms, three (3) evaluation metrics were considered, namely Detection Rate, False Alarm Rate, and time.

## 1.6 Significance of the Study

The importance of this research is to ascertain the accuracy rates of the Real-Valued Negative Selection Algorithms of RNSA and V-Detector when tested and applied on different datasets. The benchmark with other classification algorithms will give an insight on their performances for anomaly detection analysis. Therefore, it becomes essential to maintain the viability of the system using Negative Selection Algorithms, which significantly reduces false alarms.

## 1.7 Thesis Organization

The remaining part of this thesis is organized into the following chapters. Chapter 2 concerns with the relevant background information regarding using Real-Valued Negative Selection Algorithms for classification and detection purposes in the following order: (1) overview of RNSA and V-Detector architectures, (2) the advantages of RNSA and V-Detector, and (3) several techniques and applications that have been employed in the classification of four datasets. This chapter also

highlights the virtues and limitations of the existing algorithms, and arguments are brought forward for alternative methods that can be used for classifying datasets.

In Chapter 3, brief description on the steps of applying the RNSA algorithms of RNSA and V-Detector and benchmarked algorithms of SVM and KNN to datasets, rationale of selecting parameters for each algorithm, as well as evaluation method on the algorithms based on Detection Rate, False Alarm Rate, and CPU time are presented.

Comprehensive experimental evaluations of the algorithms used are presented in Chapter 4. The classification results of each algorithm with graphical representation constitute the fourth chapter. Lastly, Chapter 5 summarizes the work done, contributions, and several recommendations are suggested in order to improve the performance of the algorithms used.

## **1.8 Chapter Summary**

The need for ensuring integrity and confidentiality in data has prompted computer scientists and researchers in proffering ways and avenues to adequately secure information. This stems from anomalies or abnormality, and therefore detection improvement requires continuous efforts in many fields, including Artificial Immune System (AIS). For several data, AIS classifiers have proven their ability in successfully classifying those data by revealing the abnormalities therein. As such, this research concentrates on using Real-Valued Negative Selection Algorithms with the focus on fixed detector (RNSA) and variable detector (V-Detector) in classifying different datasets.

## CHAPTER 2

### LITERATURE REVIEW

#### 2.1 Introduction

This chapter introduces and discusses the idea behind anomaly detection, its definition and also different techniques in alleviating its harm. One of those techniques is Artificial Immune System (AIS) algorithms. Focus is directed at AIS algorithms that are based on the Biological Immune System of two types of negative selection, namely Real-Valued Negative Selection Algorithm (RNSA) and Variable-Detector Negative Selection Algorithm (V-Detector). Details of their working processes are elaborated upon. Also, exploration of the classifier algorithms of SVM and KNN are discussed.

AIS algorithms are machine-learning algorithms that embody some of the principles and attempts to take advantages of the benefits of natural immune systems in tackling complex problem domains (Brownlee, 2005). AIS is a branch of biologically inspired computation focusing on many aspects of immune systems. AIS development can be seen as having two target domains; the provision of solutions to engineering problems through the adoption of immune system inspired concepts and the provision of models and simulations to study immune system theories (Read *et al.*, 2012).

The motivation for building immune inspired solutions to engineering problems arises from the identification of properties within the immune system that are attractive from an engineering perspective. These include the self-organization of a huge number of immune cells, the distributed operation of the immune system throughout the body, pattern recognition and anomaly detection to enable the immune system to recognize pathogens, and optimization and memory to improve and remember immune responses (Muda & Shamsuddin, 2005).

Anomaly detection provides an alternate approach than that of traditional intrusion detection systems and suggests modelling both normal and malicious behavior (Jung *et al.*, 2004). It should be noted that not all anomalies are malicious in nature (Muda & Shamsuddin, 2005), and anomalies also have potentials to translate into significant critical and actionable information (Chandola *et al.*, 2009). Some of the uses of anomaly detection are detecting precedent attack behavior, zero day attack detection, intrusion detection, insider threat detection, situational awareness, and the validation and assisting with signature data. The targeted aim is the thoughtful process is differentiating what is normal from abnormal.

## 2.2 Transition from Biological Immune System to Artificial Immune System

The concept and theory of the Artificial Immune System will be incomplete without mentioning its source of inspiration in bringing its algorithms to the Biological Immune System. The body has different mechanisms to protect itself (self cells) from harmful foreign materials. One of these mechanisms is the natural immune system, and its main purpose is to detect and destroy any unwanted foreign cells (non-self cells) that could be harmful to the body. These non-self cells are known as antigens, and the natural immune system produces antibodies (as explained below) to bind to these antigens.

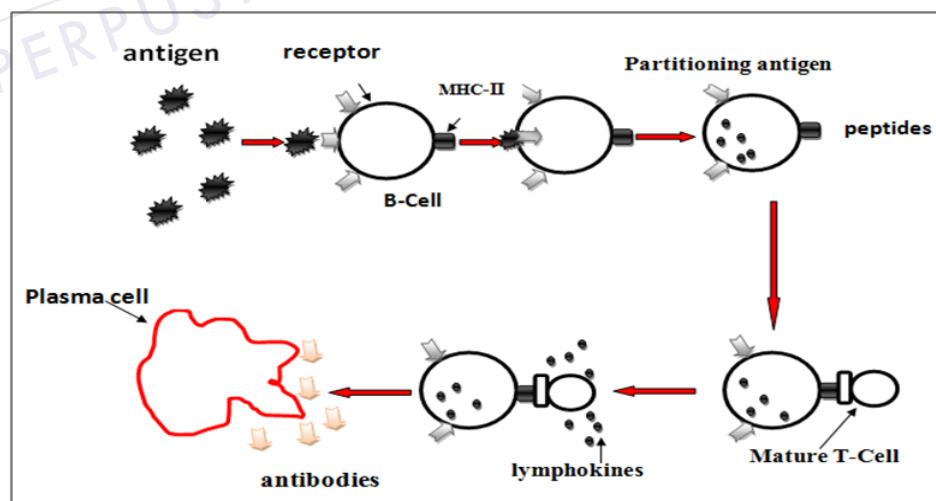


Figure 2.1: Partitioning an Antigen and Proliferation of a B-Cell into a Plasma Cell

Natural Immune System (NIS) mainly consists of lymphoid organs that create lymphocytes. The two most familiar lymphocytes are the T-Cell and B-Cell, in which both are formed in bone marrows as shown in Figure 2.1. Both T-Cell and B-Cell have receptors on their surfaces to bind with an antigen (Graaff *et al.*, 2011; Andrews, 2008). The immune system is a natural resistance to diseases using sophisticated adaptive mechanisms intended either to destroy the invaders or to neutralize their effects. As illustrated in Figure 2.2, the Biological Immune System can be classified according to functionality into four different layers of defence, which are physical, physiological, innate and adaptive. Depending on the invaders type and behavior, the immune system responds to very basic infections, and the skin serves as a physical barrier and a first line of defence. When pathogens elude the skin barrier, there are physiological barriers that provide a non-survival environment for pathogens (Elhaj *et al.*, 2013).

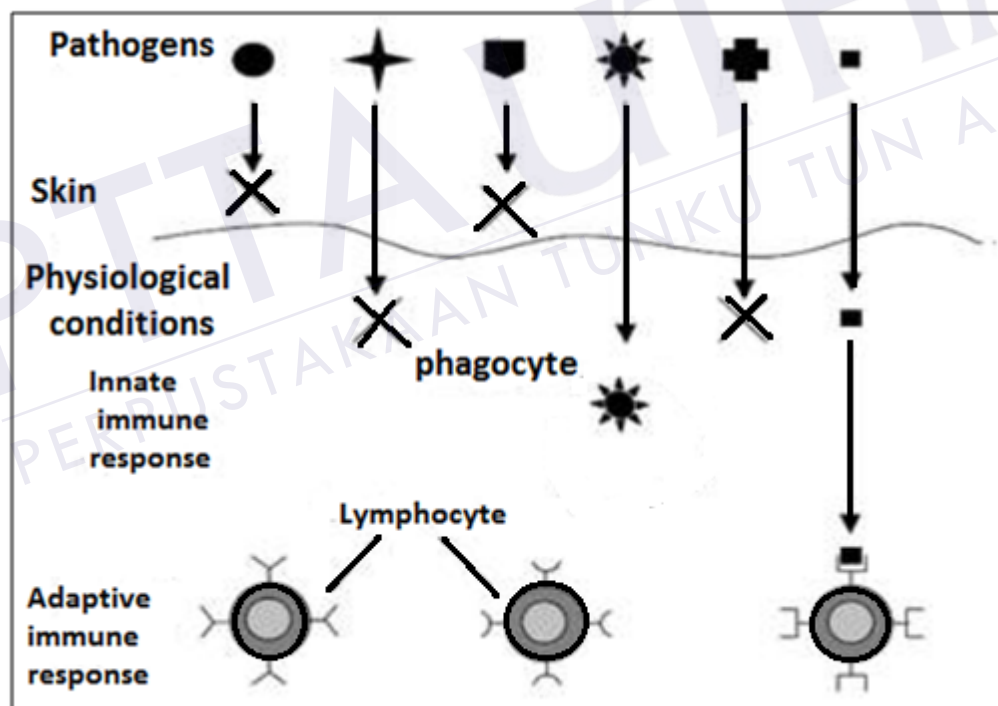


Figure 2.2: Multi-layer Structure of Immune System

The innate immune systems of these organisms are capable of self/non-self-discrimination, and also exhibit properties such as specificity, diversity and memory, which until recently have only been associated with adaptive immune systems. Low-level biological models of the mechanisms which give rise to these properties could provide important sources of inspiration for future AIS algorithms. However, if AISs

are to employ adaptive immune system mechanisms, then it is argued that they also need to incorporate innate immune system mechanisms, which control the adaptive immune system in biological organisms (Twycross & Aickelin, 2007).

Meanwhile, the artificial immune systems, techniques that are new to the scene of biologically inspired computation and artificial intelligence, are based on metaphor and abstraction from theoretical and empirical knowledge of the mammalian immune system. A robust biological process critical to the combating of disease in the body, the immune system is known to be distributed in terms of control, parallel in terms of operation, and adaptive in terms of function, all of which are features desirable for solving complex or intractable problems faced in the field of artificial intelligence (Brownlee, 2005).

There are a number of AIS models used in pattern recognition, fault detection, computer security, and a variety of other applications in the field of science and engineering (Aziz *et al.*, 2012). Most of these models emphasize on designing and applying computational algorithms and techniques using simplified models of various immunological processes and functionalities (De Castro & Timmis, 2002; Dasgupta, 2006). Also, AIS has gained increasing interest among researchers in the development of immune-based models and techniques to solve diverse complex computational or engineering problems (Al-Enezi, 2012).

Researchers have explored the main features of the AIS mechanisms and exploited them in many application areas. Based on their aspects, some AIS techniques have been found to be more suitable for certain application areas compared to other AIS approaches. It has been found that Negative Selection Models and Algorithms are widely used in fault detection and computer security applications utilizing the self/non-self-recognition aspect. Alternatively, the artificial immune network approaches have been used in clustering, classification, data analysis and data mining applications. The clonal selection models are used mostly for optimization problems (Al-Enezi *et al.*, 2009). The Danger Theory Project/Dendritic Cell Algorithm concludes the major AIS approaches that exist in literature, and they are targeted at anomaly detection and computer security applications based on the identification of danger rather than differentiating between self/non-self

as highlighted by Negative Selection Algorithm (Greensmith & Aickelin, 2007). Based on the above mentioned application areas of the AIS approaches, and in line with the objective of detecting fraud for this study, the Negative Selection Algorithm and Dendritic Cell Algorithm fall perfectly within the research confines.

Several definitions of AIS exist in literature but only three have gained popularity among AIS researchers.

- AIS is a new technique for solving combinatorial optimization problems. AIS are computational systems that explore, describe and apply different mechanisms inspired by Biological Immune Systems in order to solve problems in different domains (Guezouri & Houacine, 2012).
- AIS is a technique new to the scene of biologically inspired computation and artificial intelligence, based on metaphor and abstraction from theoretical and empirical knowledge of the mammalian immune system (Brownlee, 2005).
- The field of AIS is one of the recent biologically inspired approaches to emerge from computer science (Muda & Shamsuddin, 2005).

### 2.3 Anomaly Detection

Anomaly detection refers to the problem of finding patterns in data that do not conform to expected behavior. In understanding the principle of anomaly detection, an insight into what anomaly is all about needs to be clearly defined. An anomaly, which is also used interchangeably depending on the application area as outliers, aberrations, exceptions, or peculiarities, is defined as patterns behaving differently to the normal behavioral flow. There exist three basic types of anomalies, namely point anomalies, collective anomalies, and contextual anomalies. The point anomalies represent deviation in single data instance. When a deviation occurs with a group of data instance, the anomaly is collective. Also, deviation happening within a context depicts the contextual anomalies. Thus, anomaly detection is the process of identifying or recognizing abnormal behavioral changes in data (Lasisi *et al.*, 2014).

Anomaly detection finds extensive use in a wide variety of applications such as fraud detection for credit cards, insurance, health care, intrusion detection for cyber security, fault detection in safety critical systems, and military surveillance for enemy activities. The importance of anomaly detection is due to the fact that

anomalies in data translate into significant, and often critical, actionable information in a wide variety of application domains. For example, an anomalous traffic pattern in a computer network could mean that a hacked computer is sending out sensitive data to an unauthorized destination (Kumar, 2005). An anomalous MRI image may indicate the presence of malignant tumours (Spence *et al.*, 2001). Anomalies in credit card transaction data could indicate credit card or identity theft (Aleskerov *et al.*, 1997), or anomalous readings from a spacecraft sensor could signify a fault in some components of the spacecraft (Fujimaki *et al.*, 2005). Detecting outliers or anomalies in data has been studied in the statistics community as early as the 19<sup>th</sup> century (Edgeworth, 1887). Over time, a variety of anomaly detection techniques has been developed in several research communities. Many of these techniques have been specifically developed for certain application domains, while others are more generic.

#### **2.4 Anomaly Detection Technique**

Many modern techniques based on Artificial Intelligence, Data Mining, Neural Network, Bayesian Network, Fuzzy logic, Artificial Immune System, K- Nearest Neighbour Algorithm, Support Vector Machine, Decision Tree, Fuzzy Logic Based System, Machine Learning, Sequence Alignment, Genetic Programming and others have evolved in detecting various anomaly (Tripathi & Pavaskar, 2012; Singh & Narayan, 2012).

The various aforementioned techniques are either specific to certain application domains or more generic (Singh & Upadhyaya, 2012). The choice of which technique to use depends greatly on criteria and functionalities of each individual technique for the targeted application domain. In the following section, the AIS models of Negative Selection Algorithm using real-valued elements, namely RNSA and V-Detector for anomaly detection, are discussed in detail with their various components. Also, the selected classification algorithms in SVM and KNN are elaborated upon as well.

### 2.4.1 Negative Selection Algorithm (NSA)

Negative Selection Algorithm provides a mechanism to protect the self cells of the host, and also destroy unknown antigens (non-self cells). Within the highly-impregnable barrier of thymus, thymocytes (immature T-cells) mature by a pseudo-random genetic rearrangement of its receptors. Next (still in the thymus), these mature T-cells are exposed to self-peptides, and those that react strongly with the peptides are eliminated through a process called apoptosis. The rest of the mature T-cells that do not react with the self-peptides are subsequently released outside the thymus and into the body to fight against non-self antigens. The result of such a mechanism is that while on the one hand the (released) matured T-cells kill the non-self antigens; they are, on the other hand, non-reactive to the self (body) cells. Thus, Negative Selection Algorithm (NSA) may be viewed as a mechanism to discriminate the self from non-self cells (Dasgupta & Nino, 2008). There exist two types of NSA based on the data representation, which are the string (or binary) Negative Selection Algorithm, and the Real-Valued Negative Selection Algorithm. Detailed descriptions of the NSA types are reflected in the next section.

#### 2.4.1.1 String (or Binary)-based Negative Selection Algorithm

NSA is one of the first immune-inspired change detection algorithms that was proposed (Balachandran *et al.*, 2007). The process of keeping a computer away from intrusion can be regarded as distinguishing self and non-self. In the illumination of this idea, the research group led by Stephanie Forrest in the New Mexico University proposed the immune negative selection algorithm (Forrest *et al.*, 1994). This first implementation initially used a binary representation for the elements in the self/non-self space.

The main idea of the algorithm is to generate a set of detectors which do not harm self and distinguish the non-self (unauthorized user, virus, and others) from self (authorized users, protected data files, and others). This algorithm consists of censoring and monitoring processes as depicted in Figure 2.3 and Figure 2.4 respectively. The censoring phase caters for the generation of mature detectors. Subsequently, the system being protected is monitored for changes by the detectors

generated in the censoring stage (Forrest *et al.*, 1994). The algorithm constructs a set of competent detectors in the following steps:

- Step 1:* Define a set of self  $S$ . The data being protected is viewed as a string. The string is split into several  $L$ -length substrings. The set of self consists of several substrings.
- Step 2:* Generate a set of random candidate detectors  $R_0$ . They are also  $L$ -length strings and are generated in some probability analytical ways.
- Step 3:* Generate a set of competent detectors  $R$ . Strings from  $R_0$  that match self are eliminated. Strings that do not match any of the strings in  $S$  become members of the detector collection  $R$ . This step is called censoring.
- Step 4:* Monitor the changes of self. This is achieved by continually choosing one detector in  $R$  and testing to see if it matches with strings in  $S$ . If the self string matches one of the detector strings, a change would happen in  $S$ . Those changes are caused probably by intrusion, virus or misuse. This step is called monitoring.

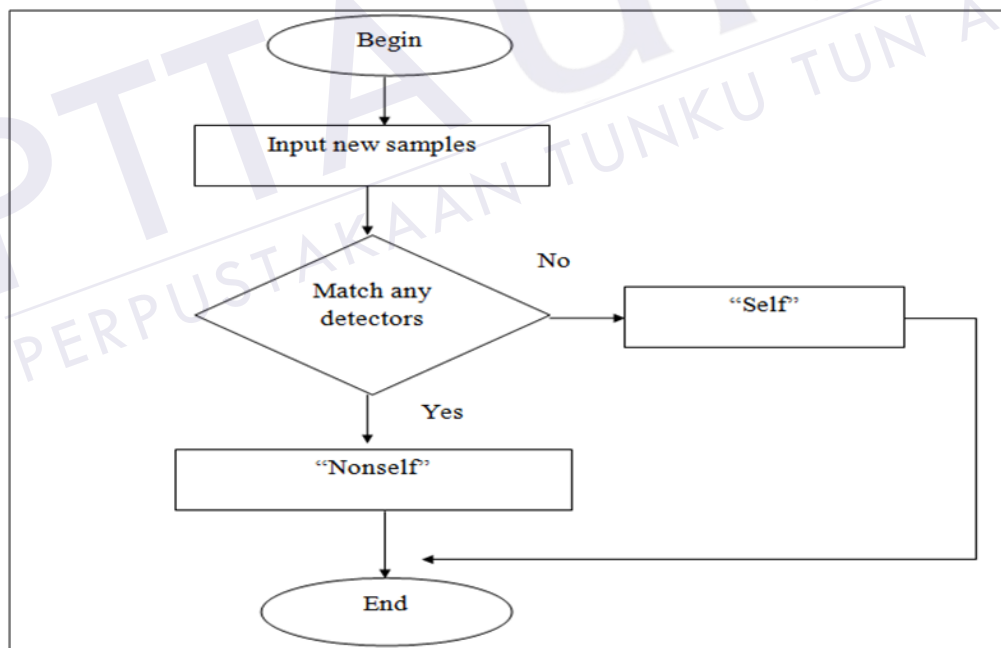


Figure 2.3: Censoring Stage

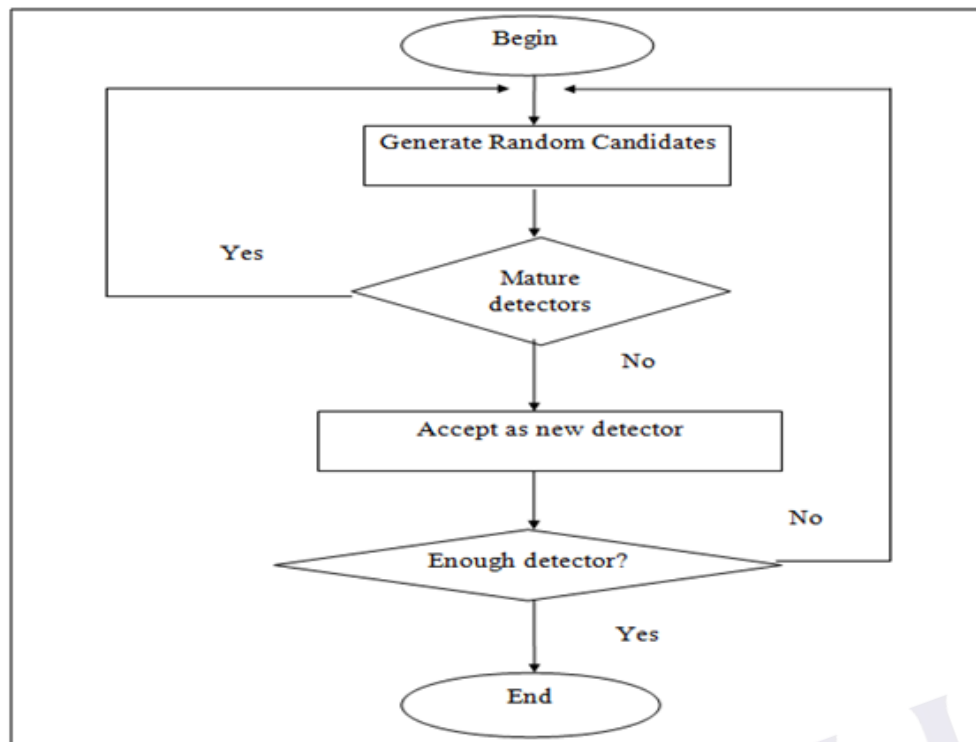


Figure 2.4: Monitoring Stage

#### 2.4.1.2 Real-Valued Negative Selection Algorithm (RNSA)

The Negative Selection Algorithm proposition as shown in Figure 2.4 (Forrest *et al.*, 1994) suffers greatly from time complexity as it is exponential to the size of the matching window (the number of bits used to compare two binary strings). In order to tackle these problems, Gonzalez *et al.* (2002) proposed a Negative Selection Algorithm that uses real-valued representation of the self/non-self space. This algorithm, called Real-Valued Negative Selection (RNSA), tries to alleviate the scaling issues of binary negative selection algorithms while it uses various schemes to speed up the detector generation process. Different RNS algorithms used different geometric shapes such as hyper-rectangles, hyper-spheres and hyper-ellipses for covering the non-self space (Ji & Dasgupta, 2006).

The Real-Valued Negative Selection Algorithm using fixed sized detectors is based on a pre-specified number of detectors. This is not the best approach, and obviously provides no guarantee that the non-self space is completely covered. However, by selecting a large enough value for the number of detectors, the algorithm is expected to provide adequate results. Figure 2.5 provides pseudo-code for the generation phase of the algorithm.

### Real-Valued Negative Selection ( $r, \eta, t, \tau$ , # of Detectors)

---

```

     $r$ : radius of detection
     $\eta$ : adaptation rate
     $t$ : once a detector reaches this, age it will be considered to
    be mature
     $\tau$ : decay rate
    Generate a random population of detectors based on # of Detectors
    While stopping criteria is not satisfied
    For each detector  $d_i$ ,
        Calculate the shortest distance to any self point,
         $dist\_min$ , and store the nearest
        point  $c_i$ 
        While ( $dist\_min < r$ )
            If age  $> t$ 
                Generate a new Detector  $d_i$ ,
            Else
                Calculate direction ( $dir$ ) using  $c_i$ ,
                Calculate  $\eta_i$ ,
                Move detector by:  $d_{(i+1)} = d_i + \eta_i * dir$ 
                Increase age + 1,
                Recalculate  $dist\_min$  and  $c_i$ 
            End If
        End While
        If (Not the first detector),
            Calculate the shortest distance to all previous
            detectors and self points,
             $dist\_min2$ , and store the nearest point  $c_i$ ,
            While ( $dist\_min2 < r$ )
                If age  $> t$ 
                    Generate a new Detector  $d_i$ ,
                Else
                    Calculate direction ( $dir$ ) using  $c_i$ ,
                    Calculate  $\eta_i$ ,
                    Move detector by:  $d_{(i+1)} = d_i + \eta_i * dir$ 
                    Increase age + 1,
                    Recalculate  $dist\_min2$  and  $c_i$ 
                End If
            End While
            Store detector as  $d_i$ 
        Else
            Store detector,
    End

```

---

Figure 2.5: RNSA Pseudo-code

The input to the algorithm is a set of self samples represented by  $n$ -dimensional points (vectors). The algorithm tries to evolve a complement set of points called antibodies or detectors that cover the non-self space. This is accomplished by an iterative process that updates the position of the detector driven by two goals:

- i. Move the detector away from the self points.

- ii. Keep the detectors separated in order to maximize the covering of the non-self space.

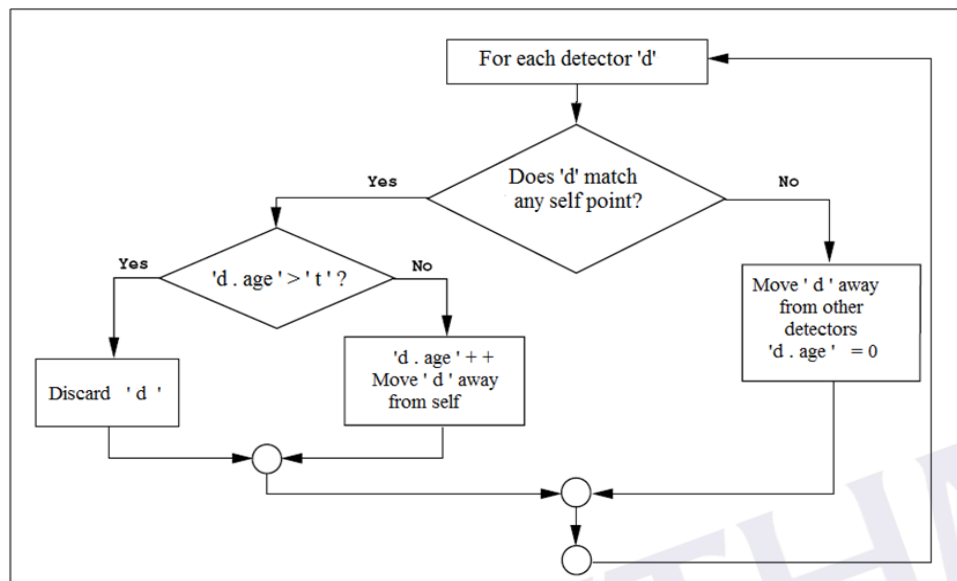
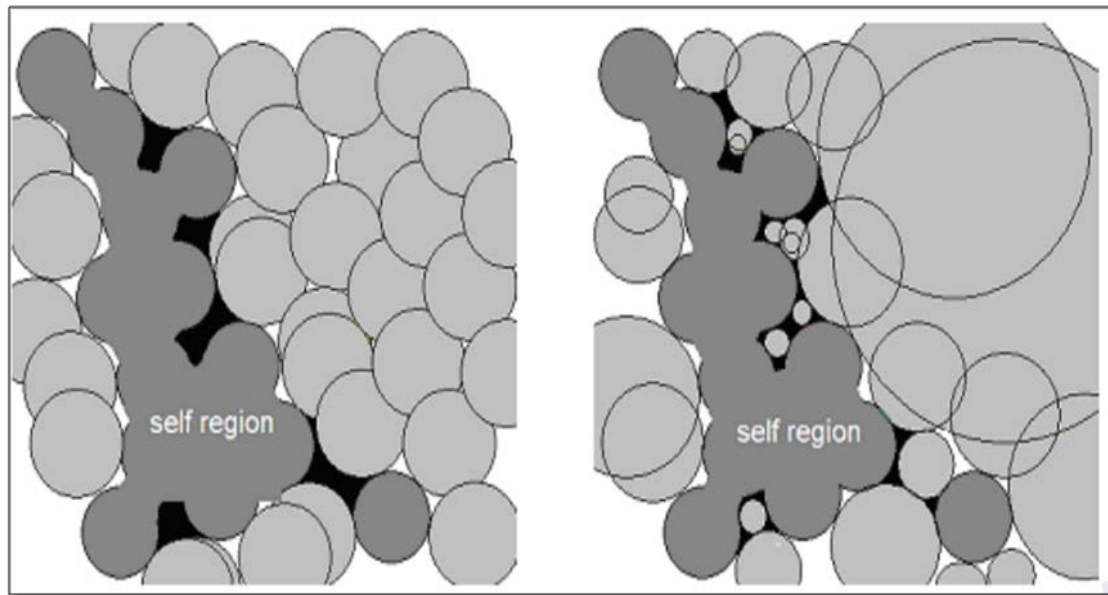


Figure 2.6: The Iterative Process of Real-Valued Negative Selection Algorithm

#### 2.4.1.3 V-Detector Negative Selection Algorithm

The first implementation of the real-valued negative selection algorithm generated detectors, in which the distance threshold (or radius) was constant throughout the entire detector set. However, the detector features can reasonably be extended to overcome this limitation. Zhou and Dasgupta proposed a new scheme of detector generation and matching mechanisms for negative selection algorithms which introduced detectors with variable properties (Ji & Dasgupta, 2007).

The proposed algorithm includes a new variable parameter, which is the radius of each detector. The threshold used by the distance matching rule defines the radius of the detectors; it is an obvious choice to make variable considering that the non-self regions covered by detectors are likely to be variable in size. The flexibility provided by the variable radius is illustrated in Figure 2.6 (Ji & Dasgupta, 2006).



a) Constant-sized Detectors

b) Variable-sized Detectors

Figure 2.7: Comparison of Detector Coverage for Different Detector Schemes

Figure 2.7 illustrates several core advantages to the method of implementing variable-sized detectors. The first apparent advantage is that a larger area of non-self space is covered with fewer detectors. The issue of “holes” is a well-known problem with Real-Valued Negative Selection Algorithms. Tiny spaces between detectors and self points cannot be filled by constant-sized detectors as illustrated in black in Figure 2.7 (a). However, by using variable-sized detectors as shown in Figure 2.7 (b), smaller detectors can be generated to cover small holes while larger detectors cover the wider non-self space.

The control parameters of the V-Detector algorithm consist of the self radius  $r_s$ , the estimated coverage  $c_0$ , and the maximum number of detectors  $D_{max}$ . The latter two are the central mechanisms for the stopping criteria; the maximum number of detectors is preset to allow the maximum allowable detectors in practice. The pseudo-code of the V-Detector is shown in Figure 2.8. The detection phase of the V-Detector algorithm is nearly similar as the fixed-sized detector algorithm. The only exception is the detector threshold utilized for the unknown data detection is based on the variable radius  $r_d$  assigned to each detector. If an unknown data instance is detected (i.e. the minimum distance to any detector is less than  $r_d$ ), it is classified as non-self; otherwise it is classified as self (Dixon, 2010).

### Real-Valued Negative Selection with Variable Detection Radius ( $r_s, M_{max}, D_{max}$ )

```

Preset Control Parameters:  $r_s, M_{max}, D_{max}$ 
While ( $m < M_{max}$ ) // ( $i < D_{max}$ )
    Generate a random Detector candidate  $di$ ,
    Calculate the shortest distance to any self points,  $dist_{min}$ ,
    If ( $dist_{min} < r_s$ )
        Return to top,
    Else
        If ( $i = 1$ )
            Store detector as  $di$  and  $dist_{min} = rdi$ ,
            Increment  $i + 1$ 
        Else
            Calculate the shortest distance for each previous
            detector,  $dist_{min2}$ ,
            If ( $dist_{min2} < rd$ )
                 $m = m + 1$ ,
            Else
                Store detector as  $di$  and  $dist_{min2} = rdi$ ,
                Increment  $i + 1$ ,
                 $m = 0$ ,
            End If
        End If
    End If
End While
End

```

Figure 2.8: Real-Valued Negative Selection V-Detector Algorithm Pseudo-code

#### 2.4.2 Support Vector Machine (SVM)

Support Vector Machine (SVM), an elegant tool for solving pattern recognition and regression problems, has been demonstrated to be valuable for several real-world applications. SVMs were introduced by Vladimir Vapnik and colleagues. The earliest mention was in 1979 by Vapnik, but the first main paper was published more than a decade later (Vapnik *et al.*, 1995). Support Vectors Machine (SVM) is a powerful classification method based on the statistical learning theory (Shevade *et al.*, 2000). SVM is also known as maximum margin classifier due to its ability to simultaneously minimize the empirical classification error and maximize the geometric margin instead of the traditional Empirical Risk Minimization principle used by other classifiers such as neural networks (de Pádúa & Nascimento, 2012). SVM was originally designed for classification and regression tasks; however, it was later expanded in other directions. The essence of SVM method is construction of optimal hyperplane, which can separate data from opposite classes using the biggest possible margin. Margin is a distance between optimal hyperplane and a vector that

lies closest to it. An example of such hyperplane is illustrated in Figure 2.9. As it can be seen in the drawing, there can be many hyperplanes that can separate two classes, but with regard to optimal choice.

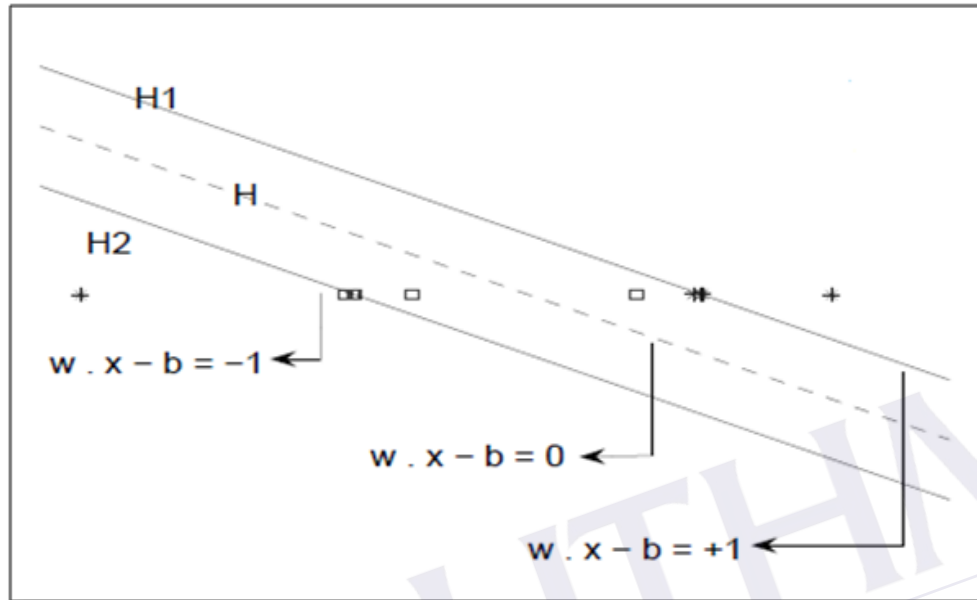


Figure 2.9: Linear Support Vector Machine

The margin  $F(x)$  and the hyperplanes as reflected in Figure 2.9 are depicted in Equations 2.2 through 2.5.  $H$  refers to hyperplanes,  $y$  refers to the output,  $x$  refers to a training or test pattern,  $w$  refers to the weight vector and the value  $b$  is the bias term. The term  $(w \cdot x)$  refers to the dot product (linear product, scalar product), which calculates the sum of the products of vector components  $w_i x_i$ .

1. Optimal hyperplane.

$$H : y = w \cdot x - b = 0 \quad (2.1)$$

2. Supporting hyperplanes - parallel and equidistant to optimal hyperplane.

$$H1 : y = w \cdot x - b = +1 \quad (2.2)$$

$$H2 : y = w \cdot x - b = -1 \quad (2.3)$$

3. Margin - distance between optimal hyperplane and a vector that lies closest to it.

$$F(x) = w \cdot x + b \quad (2.4)$$

where  $F(x)$  is a margin.

### 2.4.3 K-Nearest Neighbour (KNN)

K-Nearest Neighbour (KNN), originally proposed by Fix and Hodges, is a very simple instance-based learning algorithm (Fix & Hodges, 1951), and one of the most popular algorithms for text categorization. The idea behind K-Nearest Neighbour algorithm is quite straightforward. To classify a new dataset, the system finds the  $k$  nearest neighbours among the training dataset and uses the categories of the  $k$  nearest neighbours to weight the category candidates. One of the drawbacks of KNN algorithm is its efficiency, as it needs to compare a test dataset with all samples in the training set. In addition, the performance of this algorithm depends greatly on two factors, which are a suitable similarity function and an appropriate value for the parameter  $k$  (Li & Lu, 2003).

However, the  $k$ -nearest neighbour algorithm, which is most often used for classification, can also be used for estimation and prediction. The  $k$ -nearest neighbour is an example of instance-based learning, in which the training dataset is stored so that a classification for a new unclassified record may be found simply by comparing it to the most similar records in the training set (Marcoulides, 2005).

The traditional KNN classification has three limitations:

1. High calculation complexity: To find out the  $k$  nearest neighbour samples, all the similarities between the training samples must be calculated. When the number of training samples is less, the KNN classifier is no longer optimal, but if the training set contains a huge number of samples, the KNN classifier needs more time to calculate the similarities. This problem can be solved in 3 ways; reducing the dimensions of the feature space, using smaller datasets, or using an improved algorithm that can accelerate.

2. Dependency on the training set: The classifier is generated only with the training samples and it does not use any additional data. This makes the algorithm depend on the training set excessively; it needs recalculation even if there is a small change on the training set.
3. No weight difference between samples: All the training samples are treated equally; there is no difference between the samples with a small number of data and a huge number of data. Therefore, it does not match the actual phenomenon where the samples have uneven distribution commonly (Suguna & Thanushkodi, 2010).

## **2.5 Application of AIS in Anomaly Detection**

Anomaly detection aims to detect anomalous observations from a system. There exist some sample observations from which the normal behavior is to be learned. This anomaly detection learning problem has many important applications including the detection of anomalous jet engine vibrations (Nairac *et al.*, 1997; Hayton *et al.*, 2001; King *et al.*, 2002), abnormalities in medical data (Tarassenko *et al.*, 1995; Campbell and Bennett, 2001), unexpected conditions in engineering (Desforges *et al.*, 1998) and network intrusions (Manikopoulos and Papavassiliou, 2002; Yeung and Chow, 2002; Fan *et al.*, 2001). For more information on these and other areas of applications, as well as many methods for solving the corresponding learning problems, a timeline of Artificial Immune System is tabulated in Tables 2.1 and 2.2.

Table 2.1: A timeline of AIS (1986 to 1999)

Author (year)	Model or technique description	Type of representation used	Applications
Farmer <i>et al.</i> (1986)	An immune system as a machine learning process.	Binary strings.	NIS modeling.
Bersini and Varela (1991)	A selective evolutionary strategy based on immune recruitment.	Real-valued vectors.	Optimization.
Forrest <i>et al.</i> (1993)	Exploration of pattern recognition in NIS using genetic algorithms.	Binary strings.	NIS modeling.
Forrest <i>et al.</i> (1994)	Original Negative Selection algorithm based on the T-cell recruitment process performed by the thymus.	Strings from finite alphabet.	Change and anomaly detection.
Kephart (1994)	A computer immune system architecture to detect and repeal virus.	Byte strings (signatures).	Computer security.
Ishiguro <i>et al.</i> (1995)	A decentralized behavior arbitration mechanism to control robots inspired by the NIS.	High-level representation (robot instructions).	Robot control.
D'Haeseleer <i>et al.</i> (1996)	An efficient implementation of the negative selection algorithm for binary strings.	Binary strings.	Change and anomaly detection.
Dasgupta and Forrest (1996)	A method to detect novelties in time series based on the negative selection algorithm.	Binary string representing Real values.	Anomaly and novelty detection.
Hajela <i>et al.</i> (1997)	The use of immune networks to improve the convergence of genetic algorithms applied to design optimization.	Binary strings.	Evolutionary design optimization.
Hunt <i>et al.</i> (1999)	A machine learning system (Jisys) based on immune networks.	Mixed numerical, categorical and string data.	Fraud detection. Learning.

There has been an increase in AIS research since the middle of 1980s with a wide variety of works in different areas. Based on the survey of existing AIS literature, Tables 2.1 and 2.2 show a chronological list of some AIS models and techniques that are considered more relevant. The tables include a short description of each model or technique, along with the information about immunological mechanisms used, the type of representation, and the intended applications.

Table 2.2: A timeline of AIS (2000 to 2003)

Author (year)	Model or technique description	Type of representation used	Applications
Timmis (2000)	A resource limited artificial immune system (RAINE) for data analysis that extends the work of Cooke and Hunt	Real-valued vectors.	Data analysis. Clustering.
De Castro and Von Zuben (2000)	An immune network learning algorithm (aiNet).	Real-valued vectors.	Data analysis Clustering.
Hofmeyr <i>et al.</i> (2000)	An architecture for an artificial immune system (Lisys) for computer security.	Binary Strings.	Computer security.
Bradley and Tyrrel (2000)	A machine fault tolerance mechanism based on immune system ideas (immunotronics).	Binary strings.	Hardware fault detection and tolerance
De Castro and Von Zuben (2001)	A simulated annealing algorithm based on immune systems (SAND) applied to neural network initialization.	Real-valued vectors.	Initialization of feed-forward neural network weights.
Tarakanov and Dasgupta (2002)	An architecture to build chips that implement the immune system .	Real-valued vectors (internally represented as bits).	Pattern matching.
Coello and Cortez (2002)	An approach to handle constraints in GA based optimization.	Binary strings.	Optimization.
Nasraoui <i>etal.</i> (2003)	A scalable AIS model for dynamic unsupervised learning based on immune network theory.	Real-valued vectors.	Clustering. Dynamic learning.

One of the first works that modelled Biological Immune System (BIS) concepts in developing pattern recognition was proposed by Farmer and colleagues. Their work proposed a computational model of the BIS based on the idiotypic network theory, which explains the immune memory mechanism. This work shows that the BIS can be viewed as a learning system and suggests that it can be used as an inspiration to build machine learning techniques (Farmer *et al.*, 1986). A later work by Hajela *et al.* (1997) used immune networks to improve the convergence of genetic algorithms for design optimization. Evolutionary computation shares many elements; concepts like population, genotype-phenotype mapping, and proliferation of the most fitted are present in different AIS methods. Some of the earlier work that combined BIS ideas with evolutionary computation was developed by Bersini and Varela

## REFERENCES

- Alcalá-Fdez, J., Sánchez, L., García, S., del Jesús, M. J., Ventura, S., Garrell, J. M., Otero, J., Romero, C., Bacardit, J., Rivas, V.M., Fernández, J.C., & Herrera, F. (2009). KEEL: a software tool to assess evolutionary algorithms for data mining problems. *Soft Computing*, 13(3), pp. 307-318.
- Al-Enezi, J. (2012). Artificial immune systems based committee machine for classification application.
- Al-Enezi, J. R., Abbod, M. F., & Al-Sharhan, S. (2009). Advancement in artificial immune systems: a perspective of models, algorithms and applications. In *GCC Conference & Exhibition, 2009 5th IEEE* (pp. 1-6). IEEE.
- Aleskerov, E., Freisleben, B., & Rao, B. (1997). Cardwatch: A neural network based database mining system for credit card fraud detection. In *Computational Intelligence for Financial Engineering (CIFEr), Proceedings of the IEEE/IAFE 1997* (pp. 220-226). IEEE.
- Amer, M., & Abdennadher, S. (2011). *Comparison of unsupervised anomaly detection techniques* (Doctoral dissertation, Bachelor's Thesis 2011).
- Andrews, P. S. (2008). *An Investigation of a Methodology for the Development of Artificial Immune Systems: A Case-Study in Immune Receptor Degeneracy*. University of York, Department of Computer Science.
- Aziz, A. S. A., Salama, M. A., Hassanien, A. E., & Hanafi, S. E. O. (2012). Artificial Immune System Inspired Intrusion Detection System Using Genetic Algorithm. *Informatica (Slovenia)*, 36(4), 347-357.
- Aziz, A. S. A., Salama, M., & El-Ola Hanafi, S. (2012). Detectors generation using genetic algorithm for a negative selection inspired anomaly network intrusion detection system. In *Computer Science and Information Systems (FedCSIS), 2012 Federated Conference on* (pp. 597-602). IEEE.

- Balachandran, S., Dasgupta, D., Nino, F., & Garrett, D. (2007). A general framework for evolving multi-shaped detectors in negative selection. In *IEEE Symposium on Foundations of Computational Intelligence (FOCI'07)*, IEEE Computer Society (pp. 401-408).
- Bennett, C. C. K. P. (2001). A linear programming approach to novelty detection. In *Advances in Neural Information Processing Systems 13: Proceedings of the 2000 Conference* 13, p. 395. MIT Press.
- Bersini, H., Varela, F., Belew, R. K., & Booker, L. B. (1991). The immune recruitment mechanism: A selective evolutionary strategy. In *Proceedings of the 4th International Conference on Genetic Algorithms* (pp. 520-526). Mogan Kaufman.
- Blake, C. L., & Merz, C. J. (1998). UCI Repository of machine learning databases [<http://www.ics.uci.edu/~mlearn/MLRepository.html>]. Irvine, CA: University of California. *Department of Information and Computer Science*, 55.
- Bradley, D. W., & Tyrrell, A. M. (2000). Immunotronics: Hardware fault tolerance inspired by the immune system. In *Evolvable Systems: From Biology to Hardware* (pp. 11-20). Springer Berlin Heidelberg.
- Brownlee, J. (2005). Artificial immune recognition system (airs): a review and analysis. *Swinburne University of Technology, Melbourne, Australia, Tech. Rep*, (1-02).
- Chandola, V., Banerjee, A., & Kumar, V. (2009). Anomaly detection: A survey. *ACM Computing Surveys (CSUR)*, 41(3), 15.
- Coello, C. C., & Cortés, N. C. (2002). A parallel implementation of an artificial immune system to handle constraints in genetic algorithms: Preliminary results. In *Computational Intelligence, Proceedings of the World on Congress on* 1, pp. 819-824. IEEE.
- Cooke, D. E., & Hunt, J. E. (1995). Recognising promoter sequences using an artificial immune system. In *Proceedings of the Intelligent Systems in Molecular Biology Conference ISMB* (pp. 89-97), AAAI Press.
- D'haeseleer, P., Forrest, S., & Helman, P. (1996). An immunological approach to change detection: Algorithms, analysis and implications. In *Security and Privacy, 1996. Proceedings., 1996 IEEE Symposium on* (pp. 110-119). IEEE.

- D'haeseleer, P., Forrest, S., & Helman, P. (1997). A distributed approach to anomaly detection. Submitted to ACM Transactions on Information System Security.
- Dasgupta, D. (1999). Immunity-based intrusion detection system: a general framework. In *Proceedings of the 22nd national information systems security conference NISSC 1*, pp. 147-160.
- Dasgupta, D. (2006). Advances in artificial immune systems. *Computational Intelligence Magazine, IEEE*, 1(4), 40-49.
- Dasgupta, D., & Forrest, S. (1996). Novelty detection in time series data using ideas from immunology. In *Proceedings of the international conference on intelligent systems* (pp. 82-87).
- Dasgupta, D., & Nino, F. (2008). *Immunological computation: theory and applications*. CRC Press.
- Dasgupta, D., Yu, S., & Nino, F. (2011). Recent advances in artificial immune systems: models and applications. *Applied Soft Computing*, 11(2), 1574-1587.
- De Castro, L. N., & Von Zuben, F. J. (2000). The clonal selection algorithm with engineering applications. In *Proceedings of GECCO 2000*, pp. 36-39.
- De Castro, L. N., & Von Zuben, F. J. (2001). An immunological approach to initialize feedforward neural network weights. In *Artificial Neural Nets and Genetic Algorithms* (pp. 126-129). Springer Vienna.
- De Castro, L. N., & Timmis, J. (2002). *Artificial immune systems: a new computational intelligence approach*. Springer-Verlag, London. UK.
- De Pádua Moreira, R., & Nascimento, C. L. (2012). Prognostics of aircraft bleed valves using a SVM classification algorithm. In *Aerospace Conference, 2012 IEEE* (pp. 1-8). IEEE.
- Desforges, M. J., Jacob, P. J. and Cooper, J.E.(1998). Applications of probability density estimation to the detection of abnormal conditions in engineering. *Proceedings of the Institution of Mechanical Engineers, Part C—Mechanical engineering science*, 212:687–703.
- Dixon, S. E. (2010). *Studies on Real-Valued Negative Selection Algorithms for Self-nonsel Discrimination: A Thesis* (Master dissertation, California Polytechnic State University).
- Edgeworth, F. Y. (1887). XLI. On discordant observations. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 23(143), 364-375.

- Elhaj, M. M., Hamrawi, H., & Suliman, M. (2013). A multi-layer network defense system using artificial immune system. In *Computing, Electrical and Electronics Engineering (ICCEEE), 2013 International Conference on* (pp. 232-236). IEEE.
- Fan, W., Miller, M., Stolfo, S. J., Lee, W. and Chan, P. K. (2001). Using artificial anomalies to detect unknown and known network intrusions. In *IEEE International Conference on Data Mining (ICDM'01)*, pages 123–130. IEEE Computer Society.
- Farmer, J. D., Packard, N. H., & Perelson, A. S. (1986). The immune system, adaptation, and machine learning. *Physica D: Nonlinear Phenomena*, 22(1), 187-204.
- Fix, E., & Hodges Jr, J. L. (1951). *Discriminatory analysis-nonparametric discrimination: consistency properties*. California Univ Berkeley.
- Forrest, S., Javornik, B., Smith, R. E., & Perelson, A. S. (1993). Using genetic algorithms to explore pattern recognition in the immune system. *Evolutionary computation*, 1(3), 191-211.
- Forrest, S., Perelson, A. S., Allen, L., & Cherukuri, R. (1994). Self-nonsel discrimination in a computer. In *1994 IEEE Symposium on Security and Privacy* (pp. 202-202). IEEE Computer Society.
- Fujimaki, R., Yairi, T., & Machida, K. (2005). An approach to spacecraft anomaly detection problem using kernel feature space. In *Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining* (pp. 401-410). ACM.
- Gadi, M. F. A., Wang, X., & do Lago, A. P. (2008). Comparison with parametric optimization in credit card fraud detection. In *Machine Learning and Applications, 2008. ICMLA'08. Seventh International Conference on* (pp. 279-285). IEEE.
- Garrett, S. M. (2005). How do we evaluate artificial immune systems? *Evolutionary Computation*, 13(2), 145-177.
- Gonzalez, F., & Dasgupta, D. (2003). *A study of artificial immune systems applied to anomaly detection* (Doctoral dissertation, University of Memphis).

- Gonzalez, F., Dasgupta, D., & Kozma, R. (2002). Combining negative selection and classification techniques for anomaly detection. In *Evolutionary Computation, 2002. CEC'02. Proceedings of the 2002 Congress on* 1, pp. 705-710. IEEE.
- Graaff, A. J., & Engelbrecht, A. P. (2011). The Artificial Immune System for Fraud Detection in the Telecommunications Environment.
- Greensmith, J., & Aickelin, U. (2007). *The dendritic cell algorithm* (Doctoral dissertation, Nottingham Trent University).
- Guezouri, M., & Houacine, A. (2012). Hybrid Flow Shop Scheduling Problem Using Artificial Immune System. *International Journal of Intelligent Systems and Applications (IJISA)*, 4(10), 82.
- Hajela, P., Yoo, J., & Lee, J. (1997). GA based simulation of immune networks applications in structural optimization. *Engineering Optimization*, 29(1-4), 131-149.
- Hayton, P., Scholkopf, B., Tarassenko, L. and P.(2001). Anuzis. Support vector novelty detection applied to jet engine vibration spectra. In T. K. Leen, T. G. Dietterich, and V. Tresp, editors, *Advances in Neural Information Processing Systems 13*, pages 946–952. MIT Press.
- Hofmeyr, S. A., & Forrest, S. (2000). Architecture for an artificial immune system. *Evolutionary computation*, 8(4), 443-473.
- Hunt, J., Timmis, J., Cooke, E., Neal, M., & King, C. (1999). Jisys: The Envelopment of an Artificial Immune System for Real World Applications. In *Artificial Immune Systems and their Applications* (pp. 157-186). Springer Berlin Heidelberg.
- Ishiguro, A., Kondo, T., Watanabe, Y., & Uchikawa, Y. (1995). Dynamic behavior arbitration of autonomous mobile robots using immune networks. In *Evolutionary Computation, 1995., IEEE International Conference on* 2, pp. 722-727. IEEE.
- Ji, Z., & Dasgupta, D. (2004). Real-valued negative selection algorithm with variable-sized detectors. In *Genetic and Evolutionary Computation—GECCO 2004* (pp. 287-298). Springer Berlin Heidelberg.

- Ji, Z., & Dasgupta, D. (2006). Applicability issues of the real-valued negative selection algorithms. In *Proceedings of the 8th annual conference on Genetic and evolutionary computation* (pp. 111-118). ACM.
- Ji, Z., & Dasgupta, D. (2007). Revisiting negative selection algorithms. *Evolutionary Computation*, 15(2), 223-251.
- Jung, J., Paxson, V., Berger, A.W., and Balakrishnan, H. (2004). Fast Portscan Detection Using Sequential Hypothesis Testing, Security and Privacy. In *Proceedings 2004 IEEE Symposium* (pp. 211-225).IEEE.
- Kephart, J. O. (1994). A biologically inspired immune system for computers. In *Artificial Life IV: proceedings of the fourth international workshop on the synthesis and simulation of living systems* (pp. 130-139).
- King, S. P., King, D. M., Astley, K., Tarassenko, L., Hayton, P., & Utete, S. (2002). The use of novelty detection techniques for monitoring high-integrity plant. In *Control Applications, 2002. Proceedings of the 2002 International Conference on* 1, pp. 221-226. IEEE.
- Kumar, V. (2005). Parallel and distributed computing for cybersecurity. *IEEE Distributed Systems Online*, 6(10).
- Lasisi, A., Ghazali, R., & Herawan, T. (2014). Comparative Performance Analysis of Negative Selection Algorithm with Immune and Classification Algorithms. In *Recent Advances on Soft Computing and Data Mining* (pp. 441-452). Springer International Publishing.
- Li, B., Yu, S., & Lu, Q. (2003). An improved k-nearest neighbor algorithm for text categorization. In *Proceedings of the 20th international conference on computer processing of oriental languages (CPOL)*, Shenyang, pp. 12-19.
- Manikopoulos, C., & Papavassiliou, S. (2002). Network intrusion and fault detection: a statistical anomaly approach. *Communications Magazine, IEEE*, 40(10), 76-82.
- Marcoulides, G. A. (2005). Discovering Knowledge in Data: an Introduction to Data Mining. *Journal of the American Statistical Association*, 100(472), 1465-1465.
- Muda, A. K., & Shamsuddin, S. M. (2005). An overview of artificial immune system in pattern recognition. In *Proceedings of the Postgraduate Annual Research Seminar, In* (pp. 119-126).

- Nairac, A., Corbett-Clark, T. A., Ripley, R., Townsend, N. W., & Tarassenko, L. (1997). Choosing an appropriate model for novelty detection. In *Artificial Neural Networks, Fifth International Conference on Conf. Publ. No. 440* pp. 117-122. IET.
- Nasaroui, O., Gonzalez, F., & Dasgupta, D. (2002). The fuzzy artificial immune system: Motivations, basic concepts, and application to clustering and web profiling. In *Fuzzy Systems, 2002. FUZZ-IEEE'02. Proceedings of the 2002 IEEE International Conference on* 1, pp. 711-716. IEEE.
- Nasraoui, O., Gonzalez, F., Cardona, C., Rojas, C., & Dasgupta, D. (2003). A scalable artificial immune system model for dynamic unsupervised learning. In *Genetic and Evolutionary Computation—GECCO 2003* pp. 219-230. Springer Berlin Heidelberg.
- Read, M., Andrews, P. S., & Timmis, J. (2012). An Introduction to Artificial Immune Systems. In *Handbook of Natural Computing* (pp. 1575-1597). Springer Berlin Heidelberg.
- Shevade, S. K., Keerthi, S. S., Bhattacharyya, C., & Murthy, K. R. K. (2000). Improvements to the SMO algorithm for SVM regression. *Neural Networks, IEEE Transactions on*, 11(5), 1188-1193.
- Singh, A., & Narayan, D. (2012). A Survey on Hidden Markov Model for Credit Card Fraud Detection. *International Journal of Engineering and Advanced Technology (IJEAT)*, 1, 49-52.
- Singh, K., & Upadhyaya, S. (2012). Outlier detection: applications and techniques. *IJCSI International Journal of Computer Science Issues*, 9(1).
- Spence, C., Parra, L., & Sajda, P. (2001). Detection, synthesis and compression in mammographic image analysis with a hierarchical image probability model. In *Mathematical Methods in Biomedical Image Analysis, 2001. MMBIA 2001. IEEE Workshop on* pp. 3-10. IEEE.
- Sridevi, R., & Chattemvelli, R. (2012). Genetic algorithm and artificial immune systems: A combinational approach for network intrusion detection. In *Advances in Engineering, Science and Management (ICAESM), 2012 International Conference on* pp. 494-498. IEEE.
- Suguna, N., & Thanushkodi, K. (2010). An improved K-nearest neighbor classification using Genetic Algorithm. *International Journal of Computer Science Issues*, 7(2), 18-21.

- Tarakanov, A., & Dasgupta, D. (1999). A formal model of an artificial immune system. In *Third International Workshop on Information Processing in Cells and Tissues (IPCAT)*, 55(1), 151-158.
- Tarakanov, A., & Dasgupta, D. (2002). An immunochip architecture and its emulation. In *Evolvable Hardware, NASA/DoD Conference on* (pp. 261-261). IEEE Computer Society.
- Tarassenko, L., Hayton, P., Cerneaz, N. and Brady, M. (1995). Novelty detection for the identification of masses in mammograms. In *4th International Conference on Artificial Neural Networks*, PP 442–447.
- Timmis, J. (2000). *Artificial immune systems: a novel data analysis technique inspired by the immune network theory*. Doctoral thesis, University of Wales.
- Tripathi, K. K., & Pavaskar, M. A. (2012). Survey on Credit Card Fraud Detection Methods. *International Journal of Emerging Technology and Advanced Engineering*, 2, Issue 11, pp. 2250-2459.
- Tripathi, K. K., & Ragha, L. (2013). Hybrid Approach for Credit Card Fraud Detection. *International Journal of Soft Computing and Engineering (IJSCE)*, 3, Issue 4, pp. 2231-2307.
- Tuo, J., Ren, S., Liu, W., Li, X., Li, B., & Lei, L. (2004). Artificial immune system for fraud detection. In *Systems, Man and Cybernetics, 2004 IEEE International Conference on* 2, pp. 1407-1411. IEEE.
- Twycross, J., & Aickelin, U. (2007). Biological inspiration for artificial immune systems. In *Artificial immune systems* pp. 300-311. Springer Berlin Heidelberg.
- Vapnik, V., & Cortes, C. (1995). Support vector networks, *Machine Learning*, 20, (3). 273-297.
- Yeung, D. Y. and Chow, C. (2002.). Parzen-window network intrusion detectors. In *Proceedings of the 16th International Conference on Pattern Recognition 4*, PP 385–388. IEEE.