Re-visiting the Foundations of

Artificial Immune Systems for Data Mining

Alex A. Freitas
Computing Laboratory
University of Kent
Canterbury, CT2 7NF, UK
A.A.Freitas@kent.ac.uk

Jon Timmis
Dept. of Electronics and Dept. of Computer Science
University of York
Heslington, York, YO10 5DD, UK
jtimmis@cs.york.ac.uk

Abstract

This paper advocates a problem-oriented approach for the design of Artificial Immune Systems (AIS) for data mining. By problem-oriented approach we mean that, in real-world data mining applications, the design of an AIS should take into account the characteristics of the data to be mined together with the application domain: the components of the AIS – such as its representation, affinity function and immune process – should be tailored for the data and the application. This is in contrast with the majority of the literature, where a very generic AIS algorithm for data mining is developed and there is little or no concern in tailoring the components of the AIS for the data to be mined or the application domain. To support this problem-oriented approach, we provide an extensive critical review of the current literature on AIS for data mining, focusing on the data mining tasks of classification and anomaly detection. We discuss several important lessons to be taken from the natural immune system to design new AIS that are considerably more adaptive than current AIS. Finally, we conclude the paper with a summary of seven limitations of current AIS for data mining and 10 suggested research directions.

Keywords: artificial immune systems, data mining, machine learning, classification

1 Introduction

Artificial immune systems (AIS) are aimed at solving real-world problems, and therefore are mainly related to the areas of computer science and engineering. For the purposes of this paper, the following definition of AIS is appropriate [de Castro & Timmis 2002] (p. 58):

"*Artificial Immune Systems (AIS) are adaptive systems, inspired by theoretical immunology and observed immune functions, principles and models, which are applied to problem solving*."

This paper focuses on one kind of application for AIS, namely *data mining* [Fayyad et al. 1996], [Witten & Frank 2005]. More precisely, this paper focuses on two of the data mining tasks which have typically been addressed by AIS, namely classification (supervised learning) and anomaly detection.

The cornerstone of this paper is to advocate a problem-oriented approach for the design of AIS for data mining. By problem-oriented approach we mean that, in real-world data mining applications, intuitively the design of an AIS should be tailored for the data to be mined and the application. This is in contrast with the majority of the literature, where a very generic AIS algorithm for data mining is developed and there is little or no concern in tailoring the components of the AIS for the data to be mined or the application domain.

From a data mining point of view, the need for a problem-oriented approach is not only intuitive, but also strongly supported by the facts that every data mining algorithm has an inductive bias (which will be defined later), and every inductive bias is suitable for some datasets or application domains and unsuitable for others [Mitchell 1990], [Schaffer 1994], [Rao et al. 1995], [Michie et al. 1994]. Therefore, in order to maximize the performance of an AIS for data mining in real-world applications, one has to first carefully understand the nature of the data being mined and the requirements of the application domain, and then design an AIS (or choose an existing AIS) whose inductive bias is well suitable for the target data and application domain.

This current paper can be regarded as a major extension of our previous work discussed in [Freitas & Timmis 2003]. The main differences between this paper and our previous work are as follows. First, this paper discusses in much more detail some issues discussed in [Freitas & Timmis 2003]. In particular this paper discusses the inductive bias of knowledge representations and the application of clonal selection algorithms to data mining, topics which were not covered in [Freitas & Timmis 2003]. Second, this paper addresses a topic largely unexplored in [Freitas & Timmis 2003], namely a discussion of four important features of the natural immune system, namely: (a) the large diversity of antibody functional classes; (b) antibodies' ability to dynamically switch their functional classes; (c) the principle of antigen clustering; and (d) the principle of two-signal activation – the latter two principles specify requirements for cloning an immune cell. For each of these four features, we discuss why these features are important, as a metaphor, for the design of a more adaptive AIS than current AIS are at present, in the context of data mining.

There are several reviews of AIS, such as [Dasgupta 1999], [Timmis and Knight, 2002], [Tarakanov et al. 2003], [Timmis et al. 2004] but, as pointed out by [Garret 2005], most these reviews are outdated now, are not focused on evaluating the effectiveness of AIS and do not present many suggestions for improving the design of AIS. By contrast to these previous reviews, this paper focuses on evaluating the effectiveness of AIS, but in the context of data mining, and we suggest several improvements for their design and a number of research directions. We should also note that work such as [Tarakanov et al. 2003], [Tarakanov & Tarakanov 2004], [Tarakanov et al. 2004] has developed techniques inspired by immune network theory. Results published in these works are certainly competitive with standard data mining approaches. However, it should be noted this work, due to the focus on immune networks, is not the focus of this paper. For compactness, in this paper we focus on population-based AIS.

We are aware of only a few other recent works which present a *critical* review of AIS, as follows. [Garret 2005] has recently presented a comprehensive critical review of the area of AIS. There are three main differences between Garret's work and this paper. First, Garret evaluated AIS with respect to two major criteria, namely how distinct they are from other related computational intelligence paradigms and how effective they are. By contrast, the issue of to what extent AIS are distinct from other paradigms is out of the scope of this paper, which focuses only on how effective AIS are. Second, Garret's work addressed the area of AIS in general, without focusing on any particular kind of application. By contrast, this paper is more specialized: it focuses only on AIS for data mining. Third, Garret's discussion was "algorithm-oriented", whilst the critical review presented in this paper is much more problem-oriented. In particular, this paper discusses in detail the inductive biases of AIS for data mining; an issue not discussed in Garret's review, which had no focus on data mining.

[Hart & Timmis 2005] also presented a critical review of AIS from a type of problem-oriented perspective. The main differences between [Hart & Timmis 2005] and this paper are as follows. First, [Hart & Timmis 2005] discuss not only data mining applications but also other applications such as optimization, robotics and control; whilst these other applications are out of the scope of this paper. Second, the discussion of [Hart & Timmis 2005] also has a considerable focus on the issue of to what extent AIS are distinct from other computational intelligence paradigms – again, a topic out of the scope of this paper. Third, although [Hart & Timmis 2005] mention the importance of inductive biases in analyzing AIS for data mining, they do not elaborate on this issue, i.e., they do not discuss the inductive biases of AIS for data mining. By contrast, this paper presents a detailed discussion on the inductive biases of AIS for data mining.

We emphasize that the scope of this paper is a discussion of the inductive biases of AIS for data mining. For a review of the inductive biases of data mining algorithms belonging to other paradigms (i.e., not AIS) the reader is referred to [Michie et al. 1994] and [Mitchell 1997].

The remainder of this paper is organized as follows. Section 2 presents an overview of data mining tasks and the concept of inductive bias. Section 3 discusses representation issues in AIS for data mining. Section 4 discusses affinity issues, also in the context of data mining. Section 5 discusses two kinds of immune processes for AIS, namely clonal selection and negative selection, again in the context of data mining. Finally, Section 6 concludes the paper and suggests future research directions.

2 An Overview of Data Mining Tasks and Inductive Bias

2.1 The Classification Task

In the classification task we are given a data set with *N* data instances (records). Each instance consists of values for *m* + 1 attributes, where the *m* attributes are called predictor attributes and the other attribute is called the goal (or class) attribute. The value of the goal attribute for an instance is called the class of that instance. The data set being mined is divided into two mutually exclusive sets, namely the training set and the test set. The aim of a classification algorithm is to discover a relationship between the predictor attributes and the goal attribute using the training set only – i.e., without any access to the test set. The discovered relationship has to be useful to predict, as accurately as possible, the value of the goal attribute for each of the unknown-class instances in the test set, based on the values of the predictor attributes of that instance. In general, one wants to maximize a measure of predictive accuracy such as the simple classification accuracy rate in the test set or a more sophisticated measure based, e.g., on ROC curves [Flach 2004], [Flach 2003].

In addition to predictive accuracy, there are other criteria to evaluate the performance of a classification algorithm, in particular the comprehensibility of the discovered knowledge [Fayyad et al. 1996], [Witten and Frank 2005] – i.e., how comprehensible the classification model is to the user. The importance of knowledge comprehensibility depends on the application domain and the user. In general this is not an important criterion in many pattern recognition tasks, but it tends to be an important criterion when discovered knowledge will be validated and interpreted by a user wanting to get more insight about the data. Although there is no consensus about a precise definition of comprehensibility, it is usually accepted that some knowledge representations lend themselves better than others to the discovery of comprehensible knowledge. For instance, in general rule-based representations (to be reviewed later) have a tendency to represent knowledge in a more comprehensible way than, say, the low-level representation of the numerical weights of a neural network [Witten & Frank 2005].

Classification vs. Clustering – It is worth mention the main difference between the classification and clustering tasks because, although clustering is not the focus of the paper, this task is briefly referred to in some parts of this paper. The classification task is a form of supervised learning. By contrast, the clustering task involves a form of unsupervised learning, where there are no pre-defined classes assigned to instances. In general the objective of a clustering algorithm is to partition the instances into a set of clusters, where each cluster consists of similar instances. It should be stressed that in general the clustering task does not involve any prediction.

2.2 The Anomaly Detection Task and Its Relationship to the Classification Task

The anomaly detection task is described as follows by [Hart & Timmis 2005]:

"*Such techniques [for anomaly detection] are required to decide whether an unknown test sample is produced by the underlying probability distribution that corresponds to the training set of normal examples. Typically, only a single class is available on which to train the system. The goal of these immune inspired systems was to take examples from one class (usually what was considered to be normal operation data) and generate a set of detectors that was capable of identifying when the normal or known system had changed, thus indicating a possible intrusion.*"

This summarizes the way the anomaly detection task is typically described in the AIS literature. In this paper, however, we propose to examine this task from a broader perspective, by putting it in the context of a larger data mining literature. The essence of the anomaly detection task is that the training set contains instances of a single class, called the "self" (or normal) class, whilst the test set contains instances of two (or more) classes, the "self" and the "non-self" (e.g. intrusion) classes. The main difference between this description and the essence of the conventional classification task is that in the latter the training set contains instances of all classes (both self and non-self). Another important difference between the anomaly detection task and the classification task is that in the former the distribution of the two "classes" (anomaly vs. non-anomaly) is extremely unbalanced – i.e., finding anomalies is like finding "needles in a haystack" [Tan et al. 2006].

Despite these differences, there are important similarities between the anomaly detection task typically addressed in the AIS literature and the conventional classification task. First, in both tasks there is a division of the data into training and test set, and the algorithm must learn from the training data and apply the result of that learning on the test data. Second, the result of the learning in the training data has to be a *classification* model, i.e., a model that assigns, to each test instance, a value out of a small set of categorical (nominal) values (self and non-self in the AIS literature, or more generally any set of categorical classes). Thirdly, in several AIS papers addressing anomaly detection the performance of the algorithm is evaluated in the same way as one evaluates the performance of an algorithm for the classification task, i.e., reporting rates of false positives and false negatives, ROC curve or another appropriate measure of predictive accuracy. The evaluation of an anomaly detection AIS within a classification framework is clear, for instance, in the works of [Kim & Bentley 2002], [Balthrop et al.

2002], [Anchor et al. 2002], [Gonzales & Dasgupta 2002]. There are even projects where the evaluation of an AIS algorithm developed for anomaly detection is performed in public domain datasets that are well-known benchmarks for the classification task, and there is no natural notion of “intrusion” or “anomaly”. This is the case, for instance, in the works of [Kim & Bentley 2002] and [Greensmith et al. 2005], where the only dataset used in the experiments was the Wisconsin Breast Cancer dataset, a well-known classification benchmark from the UCI dataset repository.

Taken together, these similarities are strong evidence that the kind of anomaly detection task typically addressed in the AIS literature is very related to the conventional classification task.

2.3 Inductive Bias

Given a number of data instances (facts or observations about the real-world), the number of hypotheses or data models implying those instances is potentially infinite [Michalski et al. 1983]. Therefore, we must use a bias that goes beyond consistency with the observed data instances in order to choose a hypothesis or data model over another. An inductive bias can be defined as any (explicit or implicit) basis for favoring one hypothesis or data model over another, other than strict consistency with the data being mined [Mitchell 1990], [Mitchell 1997]. Note that, without inductive bias, a data mining algorithm would not be able to choose between two hypotheses or data models that are equally consistent with the data. Therefore, the algorithm would be limited, in essence, to a kind of simple rote learning. Hence, every data mining algorithm that performs some generalization – and not merely memorizes the data – must have an inductive bias. This includes virtually every useful data mining algorithm, since merely memorizing the data could hardly be called data mining.

For the purpose of the review of AIS in this paper, a very important point is that any inductive bias has an *application domain-dependent* (*more specifically, a dataset-dependent*) *effectiveness*. Since every data mining algorithm based on machine learning has an inductive bias, it follows that the performance of a data mining algorithm is very dependant on the application domain and the data being mined. The application domain/dataset-dependent effectiveness of algorithms and their corresponding inductive biases has been well established in the machine learning literature for more than a decade – both theoretically [Schaffer 1994], [Rao et al. 1995] and empirically [Michie et al. 1994], [Lim et al. 2000].

This fact strongly suggests that, in order to maximize the performance of a data mining algorithm in real-world applications, one has to first carefully understand the nature of the data being mined and the requirements of the application domain, and then design a new algorithm or choose an existing algorithm whose inductive bias is well suited for the target data and application domain. This is a problem-oriented approach, and it requires a good understanding of the inductive biases of data mining algorithms. In particular, this paper will discuss in detail the inductive biases of AIS for data mining, therefore giving a significant contribution to the design or choice of an AIS whose inductive bias is suitable for the target data and application domain.

3 Representation Issues

According to [de Castro & Timmis 2002], an antibody – representing a candidate solution to the target problem – can, in general, be represented by an L-dimensional vector $Ab = <Ab_1, \ldots, Ab_L>$, where L is the length (i.e. the number of components) of the vector. In the context of the classification task, usually each Ab_i, $i = 1,\ldots L$, essentially represents the value of the i-th attribute (feature) of the data being mined.

In [Freitas & Timmis 2003] we briefly reviewed three kinds of antibody representation with respect to attribute data types, namely binary, continuous (real-valued) and categorical (nominal) *data representations*. In this paper we propose to go considerably further, by considering *knowledge representations* [Langley et al. 1996], as discussed in the next subsection.

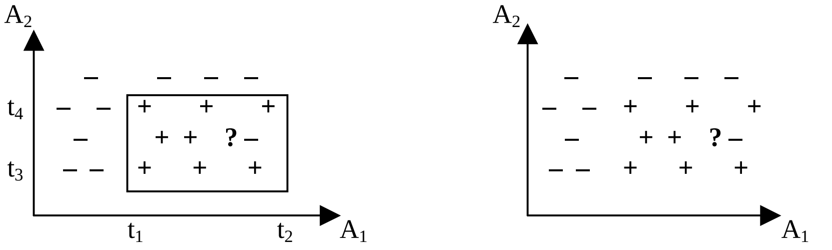
3.1 A Brief Review of Instance-Based and Rule-Based Knowledge Representations

Most AIS for classification use an instance-based representation. This includes well-known AIS such as AIRS [Watkins 2001], [Watkis & Boggess 2002b], [Watkins et al 2004] and CLONALG [de Castro & von Zuben 2000a], [de Castro & von Zuben 2002a]. Instance-based representations have a form of specificity bias, in the sense that in this representation the candidate solutions considered by the classification algorithm take the form of a subset of the original data instances, each of them with all its attribute values. This allows the representation of very specific relationships between the predictor attributes and the classes of instances.

A very different kind of representation is the rule-based one. This representation is used, for instance, in IFRAIS [Alves et al. 2004] – an AIS for discovering fuzzy classification rules. Rule-based representations have a kind of generality bias. In this representation, the candidate solutions considered by the algorithm take the form of IF-THEN classification rules, where each rule typically contains a conjunction of a few attribute values in its antecedent, namely just the attribute values that are relevant to predict the class specified in the rule consequent. A rule can be used to classify any instance satisfying the conjunction of attribute values in its antecedent, so that each rule is effectively a generalized representation for the set of instances satisfying its antecedent.

The difference between the rule-based and instance-based representations can be more easily understood from a geometrical point of view, as illustrated by Figure 1. In this figure the training instances are represented by "+" or "–", denoting that they belong to the positive or negative class, respectively, and an unknown-class test instance is denoted by "?". To keep the example simple, the figure refers to a data space with just two attributes, A_1 and A_2. The position of an instance in that space is given by its values for attributes A_1 and A_2. In Figure 1(a) there is a box covering a set of 9 training instances. This box represents the following classification rule: IF ($t_1 \leq A_1 \leq t_2$) AND ($t_3 \leq A_2 \leq t_4$) THEN (class = "+"). Given that the test instance denoted by "?" satisfies this rule, it will be classified as a positive instance. This is a result of the generalization made by the rule. By contrast, the same instance set is shown in Figure 1(b), but now an instance-based representation is used. Assuming the test instance is assigned the class of its nearest training instance (as in the 1-NN algorithm [Aha 1997]), that test instance will be assigned the negative class. This is a result of the specificity bias of the instance-based classification algorithm. Of course, we could increase the generality of the instance-based representation by assigning the test instance to the class of the majority of its *k* nearest neighbors, as in the *k*-NN algorithm [Aha 1997], but even in this case the value of *k* is typically a small integer, so that even *k*-NN algorithms still have a relatively strong specificity bias,. Further discussions contrasting the specificity bias of instance-based representations with the generality bias of rule-based representations can be found in [Ting 1994], [Carvalho & Freitas 2004].

In any case, the question of which bias, generality or specificity, leads to a higher predictive accuracy depends strongly on the data being mined – this is also true for any other kind of inductive bias. In Figure 1, it is possible that the negative training instance nearest to the test instance contains noisy data – say the wrong value of A_1 or A_2, or the wrong value of the class. If so, the rule of Figure 1(a) is correctly generalizing all the training instances inside the box and the test instance is likely to really have the positive class. On the other hand, the negative training instance nearest to the test instance can contain correct data and represent a true exception to the more generic pattern represented by the rule of Figure 1(a). If so, the test instance is likely to really have the negative class and the very specific prediction of the 1-NN algorithm would be more likely to be correct.



(a) rule-based representation (b) instance-based representation

Figure1: Difference between rule-based and instance-based representation

Regardless of predictive accuracy issues, one advantage of the rule-based representation over the instance-based one is that the former tends to be more comprehensible to the user [Witten & Frank 2005]. This is because it uses an intuitively interpretable IF-THEN structure and the antecedent usually contains a conjunction of a few conditions, rather than a vector containing values for all attributes as in a typical instance-based representation.

On the other hand, the generalized representation of the data associated with IF-THEN rules comes with a corresponding disadvantage. As pointed out by [Chao & Forrest 2003], rule induction requires that many training instances be observed before a rule generalizing those instances is created. By contrast, the instance-based representation can learn, in principle, even from a single instance. A related point is that instance-based representations are more incremental, i.e., they are more easily updated as the data being mined changes. This remark provides further support to the claim of [Hart & Timmis 2005] that continuous learning is a promising application of AIS, since most current AIS use an instance-based representation.

To conclude this section, the question: “Which knowledge representation is better, the instance-based or the rule-based one?” does not make sense per se, in isolated form. It all depends on the nature of the data being mined, the requirements of the application domain, how important knowledge comprehensibility is to the user, etc. From a data mining, problem-oriented perspective, the right question to ask is: “For a particular application domain and a particular dataset to be mined, which knowledge representation should be used?” The optimal answer might not even be the instance-based or rule-based representation, of course, since many other knowledge representations are available [Witten & Frank 2005], [Langley 1996].

3.2 A Critical Review of Representation Issues in a Number of Existing Artificial Immune Systems

Table 1 presents a summary of the antibody representation used by a number of AIS (this table is not exhaustive). Each row in that table corresponds to a given kind of AIS work. For each work the table reports: (a) the kind of application / data mining task addressed by that work; (b) the antibody representation; (c) the general geometrical shape of the antibody’s recognition area in the data space – or how an antibody recognizes an antigen, if the geometrical shape of the antibody’s recognition area is not well defined; and (d) the knowledge representation paradigm: instance-based or rule-based.

The majority of the AIS mentioned in Table 1 use one out of a couple of “standard” antibody representations, namely a real-valued vector or a binary vector, or a variation of those representations. In general these works show little or no concern in designing a representation tailored for the data or the application domain.

Table 1: A Summary of Representation Issues in a Number of Existing AIS

AIS	Kind of application or task	Antibody representation	Antibody's recognition area	Kn. Rep. Paradigm
[Dasgupta et al. 2004]	Fault detection / classification	Real-valued vector	Hyper-sphere with a variable radius	Instance-based
[Gonzales et al. 2002]	Anomaly detection / classification	Real-valued vector	Hyper-sphere with variable radius	Instance-based
[Cserey et al. 2004]	Real-time processing of image sequences for surveillance	Binary vector with "don't care" values	Hyper-sphere with fixed radius	Instance-based
[Dasgupta & Majumdar 2002]	Anomaly detection / classification	Binary vector encoding real numbers	Antibody and antigen must match in r contiguous bits	Instance-based
[Sarafijanovic & Le Boudec 2004]	Misbehaviour detection in mobile ad-hoc networks	Binary vector encoding discretised numbers	Antibody must have "1" in every position where antigen has "1"	Instance-based
[Balthrop et al. 2002]	Network intrusion detection / Classification	Binary vector	Antibody and antigen must match in r contiguous bits	Instance-based
[Anchor et al. 2002]	Network intrusion detection / Classification	Binary vector encoding ranges of numerical variables	Hyper-rectangle (each attribute has lower and upper bounds)	Instance-based
[Taylor & Corne 2003]	Fault detection / classification in time series	Discrete-number vector using two encodings – one of them tailored for time series	Hyper-sphere with fixed radius or matching r contiguous bits of antigen	Instance-based
[de Castro & von Zuben 2001 ; 2002b]	Clustering	Real-valued vector	Select *n* antibodies nearest to the current antigen	Instance-based
[de Castro & von Zuben 2000a; 2002a]	Digit recognition	Binary vector	Select *n* antibodies nearest to the current antigen	Instance-based
[White & Garret 2003]	Digit recognition	Real-valued vector	Select *n* antibodies nearest to the current antigen	Instance-based
[Watkins et al. 2004], [Watkings & Boggess 2002a]	Classification	Real-valued vector	Select memory cell nearest to the current antigen[1]	Instance-based
[Sahan et al. 2005]	Medical diagnosis / classification	Real-valued vector	Hyper-sphere with fixed radius	Instance-based
[Alves et al. 2004]	Classification	A fuzzy classification rule	Hyper-rectangle	Rule-based
[Castro et al. 2005]	Classification	A set of fuzzy classification rules	Hyper-rectangle	Rule-based
[Secker et al 2003]	Classification	Vector encoding words extracted from emails	Antibody must have *n* number of equal words to antigen	Instance based
[Ayara et al 2005]	Error detection / Classification	Vector containing discrete states of an automated teller machine	Antibody and antigen must match in *r*-contiguous bits	Instance based

[1] It should be noted AIRS has two recognition areas during training, one for deciding if an antibody is cloned, the other used to decide if a new candidate memory cell is actually added to the set of memory cells. Here we discuss the first.

The work of [Taylor & Corne 2003] is an exception. This work used an antibody representation consisting of a vector of discrete, integer numbers in the range 0-9. Two alternative encodings of the data into that representation were investigated. First, each original temperature value – in a time series of temperature values – was discretized into an integer in [0…9]. The second kind of encoding involved comparing the temperature value at time *t* with the temperature at the previous time *t* – 1 in a time series, and then encode that difference as the discrete number 0, 1, or 2, to denote an upward slope, no change or downward slope, respectively. This representation was designed in collaboration with experts in the application domain, involving fault detection in refrigeration systems. To quote from the paper:

"*In conjunction with experts in the application area, our view is that the key elements of a faulty defrost temperature curve are not the precise pattern of real-valued temperatures, but the local 'ruggedness' of the temperature curve.*"

This kind of data encoding tailored for the data being mined, designed in collaboration with experts in the application domain, is a good example of the problem-oriented approach for the design of AIS advocated in this paper. The other exception is the work by [Ayara et al 2005]. In that paper, the authors used explicit domain knowledge from engineers to establish failure criteria for ATMs (Automated Teller Machines). This knowledge is used to identify sequences of states that lead to failure: these are then used as the basis for antibodies that predict if an ATM is likely to fail or not. Results obtained showed that it was possible to identify up to 12 hours in advance of such failure.

It should also be noted that, in some AIS reported in Table 1, the corresponding paper makes it explicit the fact that the choice of the used representation was driven by an algorithm-oriented approach, rather than a problem-oriented approach. For example, quoting [Anchor et al. 2002]:

"*The binary string representation is employed to allow for easy manipulation by a genetic algorithm in the affinity maturation.*"

It is not clear that a binary representation is needed for that reason, since evolutionary algorithms can handle real-valued variables without any significant problem, and the real-valued individual representation is arguably more suitable than the binary one when the data consists of real-valued variables [Back 2000], [Freitas 2002].

As another example of work emphasizing the algorithm-oriented nature of an AIS, [Dasgupta & Majumdar 2002] used an AIS for anomaly detection in personnel data containing both numerical and categorical (nominal) attributes. That project ignored the categorical attributes and used an antibody representation containing only numerical attributes. Hence, several attributes that were potentially useful for the target anomaly detection task were ignored, possibly reducing the predictive accuracy of the system. The justification for working only with numerical attributes given by the authors was:

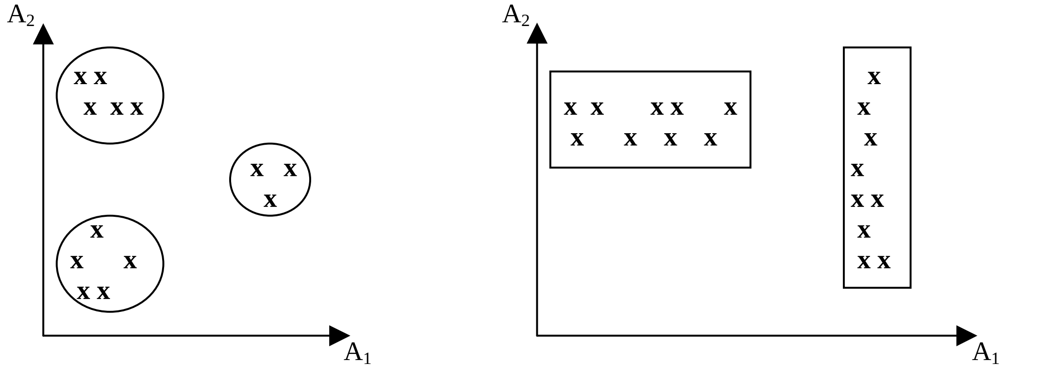
“*It is difficult to numerically represent categorical data. Any attempt to do so arbitrarily imposes an ordering in the data, which is not true in real life. To apply the negative selection algorithm we need numerical data.*”

We agree with the authors that a numerical representation of categorical data is not a good approach, introducing an arbitrary order in the data. However, it does not follow that the best solution is to ignore categorical data and use only numerical data. Rather, intuitively a better approach is to use a hybrid numerical/categorical representation. As pointed out by [Freitas & Timmis 2003], it is possible to use an affinity function that handles categorical attributes without converting them into numerical attributes. A very simple way of doing that is to define the distance between two categorical values as 0 if the two values are the same or 1 if they are different. This does not introduce any arbitrary order in the data and it returns a numerical value that can be straightforwardly used in the formula for any distance measure. A more sophisticated approach to measure the numerical distance between two categorical attribute values (again, without introducing an artificial ordering in the data) consists of using the Value Difference Metric [Stanfill & Waltz 1986], [Liao et al. 1998].

A common choice of representation in existing AIS for data mining consists of representing an antibody as a real-valued vector and, when matching an antibody and an antigen, considering that the antigen is recognized by the antibody if the distance between them is smaller than a given threshold. This defines, for each antibody, a hyper-spherical recognition region centered at the coordinates of its real-valued data vector and with radius given by the distance threshold. This kind of representation, sometimes called “artificial recognition ball”, is used in several AIS mentioned in Table 1, namely the works of [Dasgupta et al. 2004], [Gonzales et al. 2002], [Cserey et al. 2004], [Taylor & Corne 2003]**.** In early AIS the radius was fixed, but more recently some works proposed a variable radius [Dasgupta et al. 2004], [Gonzales et al. 2002]. The idea of a variable radius is an improvement, intuitively making the antibodies more adaptable to the data being mined, but it should be pointed that, even when the length of the radius is variable, existing AIS still use the fixed representation of a hyper-spherical recognition ball, which has its own representational bias and so is not suitable for all kinds of data being mined.

To see this point, consider for example the two very simple datasets shown in Figure 2(a) and 2(b), and let the target task be clustering. In Figure 2, each data instance is represented by a “x”. A hyper-spherical representation is naturally suitable for the data of Figure 2(a), but not suitable for the data of Figure 2(b), for which a hyper-rectangular representation is more suitable.

It should be noted that not all instance-based representations have a hyper-spherical recognition region. In particular, the work of [Anchor et al. 2002] proposes a hyper-rectangular representation, where the antibody encodes a lower and an upper bound for the value of each attribute.



(a) suitability of hyper-spherical representation (b) suitability of hyper-rectangular representation

Figure 2: Examples of datasets for which hyper-spherical or hyper-rectangular representations are suitable

There are also several works where an antibody's recognition region is not explicitly defined as a property of the antibody itself, but is rather implicitly defined by a competition between that antibody and other antibodies [de Castro & von Zuben 2002b], [de Castro & von Zuben 2002a], [White & Garret 2003], [Watkins et al. 2004]. In such AIS, when an antigen is presented to the system, the nearest antibody(ies) to that antigen is(are) deemed to recognize that antigen, regardless of the actual value of the distance between the antibody(ies) and the antigen. This avoids the need to specify a parameter such as the radius of a hyper-sphere, although this advantage might be cancelled out if the system requires another parameter such as the number of nearest neighbors (antibodies) that should be considered as recognizing the current antigen – and therefore are selected for cloning.

Almost all AIS mentioned in Table 1 use an instance-based representation. The only exceptions are the works of [Alves et al. 2004] and [Castro et al. 2005]. In [Alves et al. 2004] an antibody represents a fuzzy classification rule. In geometric terms, the rule-based representation corresponds to a hyper-rectangle in the data space. Interestingly, both the work of [Alves et al. 2004] and the work of [Anchor et al. 2002] use a hyper-rectangle representation. The main differences between these two works are as follows. First, the work of [Anchor et al. 2002] uses an instance-based representation, where the data vector contains a pair of values (lower and upper bounds) for every attribute of the data being mined. By contrast, the work of [Alves et al. 2004] uses a rule-based representation, where each rule typically contains values for relatively few attributes, rather than all attributes. Hence, in the latter representation an antibody tends to be considerably shorter than an antibody in the former, which makes the latter representation easier to interpret for the user. Second, the antibody representation used in [Anchor et al. 2002] handles only numerical data, whilst the antibody representation used in [Alves et al. 2004] handles both numerical and categorical data.

In [Castro et al. 2005] an antibody represents a set of fuzzy classification rules, rather than just a single rule. From a geometrical point of view, each of the rules encoded in an antibody has the same hyper-rectangular shape as the rule representation used in [Alves et al. 2004]. Encoding a set of rules (rather than a single rule) into an antibody has the advantage that the antibody's fitness evaluation directly takes into account the interactions among rules, but it has the disadvantage that the antibody representation becomes more complex and the size of the search space for the AIS becomes correspondingly larger [Freitas 2002].

Both the instance-based representation of [Anchor et al. 2002] and the rule-based representations of [Alves et al. 2004], [Castro et al. 2005] have the limitation that they represent only hyper-rectangles whose boundaries are defined by propositional-logic conditions such as "(18 ≤ Age ≤ 30)", and not first-order logic (relational) conditions such as "(Income > Expenses)". This point will be further discussed later.

Overall, considering the contents of Table 1 and the previous analysis, it should be noted that existing AIS are not very flexible nor adaptive in their choice of antibody representation. That is, in general each of these AIS uses a fixed kind of representation throughout the run of the system, implicitly assuming that that representation is suitable for the data being mined, an assumption that is not usually justified.

With this important limitation of current AIS in mind, it is now timely to turn to a discussion of "antibody representation" issues in the *natural* immune system (a much more flexible system), in order to investigate the possibility of identifying useful metaphors for the design of a more adaptive AIS. Indeed, work in [Stepney et al. 2005] advocate the re-examination of the immunological literature and to move away from simplistic views of immunological operation, and seek to capture a richer (or appropriate level of metaphor) aspect of immunology that can be used in the development of AIS.

3.3 Antibody Diversity in the Natural Immune System

There are four main kinds of antibody according to their functional class, namely IgA, IgE, IgG, and IgM. These kinds of antibody have different functional properties, evolved to function in different environments (different parts of the body) and mediate different biological responses following antigen binding [Sompayrac 2003], [Alberts et al. 2002], [Mims 2000].

IgM is a relatively large antibody, and it is the first kind of antibody secreted into the blood in the early stages of a primary immune response. As the immune response develops, IgM antibodies are replaced by other kinds of antibodies, mainly IgG ones. The logic behind the use of IgM as a “first antibody” is that in the early stage of an infection IgM antibodies are more effective than IgG ones, because IgM antibodies are considerably better at “fixing complement” – i.e., activating the complement cascade. Furthermore, IgM antibodies are very good at binding to viruses and preventing them from attaching to cells that they could infect. Due to their large size, IgM antibodies cannot easily pass through blood vessel walls, so they stay mainly in the blood.

IgG is the most abundant kind of antibody in the blood. IgG is produced in large quantities during a secondary immune response. There are several different subclasses of IgG antibodies, with different functions. For instance, IgG1 is very good at opsonizing invaders, i.e., preparing them to be ingested by phagocytes such as macrophages; whereas IgG3 is the subclass which best fixes complement. IgG antibodies are good at binding to viruses and preventing them from attaching to cells that they could infect.

IgA antibodies are very good at clumping pathogens, creating clumps that are large enough to be swept out of the body with the mucus. IgA is the most abundant kind of antibody in the body. They are not so numerous in the blood (where IgG predominate), but there are a very large number of them in the mucosal surfaces of the body. IgA is sometimes called a secretory antibody and it is the main kind of antibody in secretions such as saliva, tears, respiratory and intestinal secretions, and it is also secreted into the milk of nursing mothers. Although IgA antibodies are very good in the fight against mucosal invaders, they are useless at fixing complement.

IgE antibodies are present in relatively small amounts in the body (by comparison with other kinds of antibodies), and they are produced by B cells lying just below the respiratory and intestinal surfaces. IgE binds with unusually high affinity to special cells such as mast cells, which protect us against parasitic infections. Then, when the IgE antibody binds an antigen, it triggers the mast cell to secrete substances that kill the parasite.

Interestingly, the immune system is quite clever in using these different kinds of antibodies, because a B cell can switch the kind of antibody that it produces based “on demand”. As mentioned earlier, when B cells are first activated they secrete mainly IgM antibodies. In a later stage of the immune response, many B cells change the kind of antibody that they are producing (to IgG, IgE or IgA), based on the combination of antigens binding to the B cell and cytokines secreted by helper T cells. For instance, if a B cell detects an abundance of cytokines IL-4 and IL-5 in its environment, it tends to switch their kind of antibody from IgM to IgE – ideal for fighting parasitic worms. By contrast, if a B cell detects TGF-β it tends to change its kind of antibody from IgM to IgA – ideal for the common cold.

3.4 Lessons To Be Taken From Natural Antibody Diversity for Designing More Adaptive AIS

First of all, recall that, in AIS, an antibody is a candidate solution to the target problem. At a high level of abstraction, there are two main lessons that we can take from the antibody diversity in the natural adaptive immune system, in order to identify generic principles for designing more adaptive AIS than the AIS currently available in the literature.

The first lesson is that there is a considerable diversity of natural antibodies. They come into several functional classes and subclasses. Each of those classes and subclasses is particularly suitable for protecting us against one kind of invader. Each of the functional classes of natural antibodies can be thought of as a kind of "antibody representation", having a role conceptually analogous (as a metaphor) to a knowledge representation of an artificial antibody in an AIS. In terms of diversity of representations, AIS lag considerably behind their natural counterpart. As discussed earlier, in general AIS use only one kind of antibody representation.

The motivation for designing an AIS that considers more than one kind of knowledge representation is clear in the context of data mining, where different subsets of the data being mined may be better covered by different kinds of knowledge representation. A very simple example of this point is shown in Figure 3, which is based on the same geometrical perspective of knowledge representations (and the same notation) as used in Figure 2. Consider first the six positive training instances within the box in the lower-right part of the data space in Figure 3. These instances are naturally covered by the rule represented by that box, i.e. the rule: IF ($t_1 \leq A_1 \leq t_2$) AND ($A_2 \leq t_3$) THEN (class = "+"). Borrowing terminology from the field of Logic, this kind of rule is called a propositional rule. The majority of rule induction algorithms discover rules in this representation. This representation is not suitable, however, to cover the many positive examples above the diagonal line in Figure 3. Those examples are ideally covered by a more sophisticated kind of rule, viz. the rule: IF ($A_2 > A_1$) THEN (class = "+"). This is a first-order logic (or relational) rule representation, whose antecedent is comparing the values of two attributes, rather than just comparing the value of an attribute with a given threshold value. On the other hand, there are other training instances which apparently would not be well covered by a generic rule, and for which an instance-based representation seems more appropriate. Examples are the two negative training instances above the box and below the diagonal line in Figure 3. This figure is a very simplified example, but it illustrates the point. The need for diverse knowledge representations intuitively tends to grow stronger as larger and more complex datasets are mined. In other words, the ability to use diverse knowledge representations is a desirable characteristic of AIS for scaling them up to large and complex datasets, a research direction very open in the AIS literature.

Although the use of hybrid knowledge representations is not easy and is not the main approach in machine learning and data mining, there is evidence that well-designed systems with hybrid knowledge representations are effective. To quote just five examples, [Ting 1994] used a hybrid decision tree and instance-based learning representation; [Lopes and Jorge 2000] and [Domingos 1995] used a hybrid rule and instance-based representation; [Quinlan 1993] used a hybrid system involving instance-based learning and three different forms of model-based learning; and [Carvalho & Freitas 2004] used a hybrid decision tree and rule representation where the rules are evolved by a genetic algorithm. In general these systems have been shown to obtain a high predictive accuracy, and they often obtained better results than the individual base algorithms which they combined.

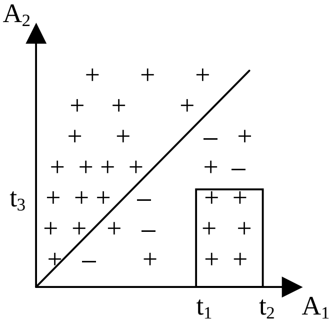


Figure 3: Example of Data Requiring Diversity of Knowledge Representations

The second lesson about representation to be taken from the natural immune system is that B cells are adaptive enough to switch the class of antibody that they produce as necessary, depending on the kind of invader that is currently attacking the body. Again, AIS lag considerably behind their natural counterpart in this kind of adaptivity, since in general they do not allow the kind of knowledge representation to be changed as new antigens are found – i.e., as a new region of the search space is explored.

4 Affinity Issues

4.1 A Review of the Importance of Affinity Functions in Artificial Immune Systems

Any kind of affinity function used to decide which antibodies will be cloned (and how many clones of them will be produced) is, in more general data mining terms, an evaluation function that guides the search for better models of the data; and any evaluation function is a source of inductive bias. Therefore, one should choose or design an affinity function whose bias is suitable for the data being mined [Freitas & Timmis 2003].

Since most AIS use an instance-based representation, as discussed in Section 3, it is natural that most AIS use an affinity function that is specified in terms of a distance function, i.e., the smaller the distance between an antibody and an antigen, the higher the affinity between them, and so the more stimulated the antibody is.

To illustrate the importance of the choice of a distance function, consider the following example, adapted from [Freitas & Timmis 2003]. The first column of Table 2 shows the coordinates – in a two-dimensional data space – of three data instances, namely antigen A and antibodies B and C. Which of the two antibodies, B or C, is nearest to antigen A? This is the sort of question that has to be continuously answered within the execution of an AIS such as CLONALG or aiNet, where the *n* nearest antibodies to an antigen are chosen to be cloned, and within the execution of AIRS, where the nearest (highest affinity) memory cell is chosen to generate new artificial recognition balls. Interestingly, different distance measures might give quite different results. This is shown in the second and third columns of Table 2, where two popular distance measures are considered: Manhattan and Euclidean distance [Liao et al. 1998]. As shown in the table, according to the Manhattan distance, the distance between A and B is 8, whilst the distance between A and C is 7, which suggests that C is the nearest antibody to A. However, according to the Euclidean distance, B is the nearest antibody to A.

This significant difference in the results happens because the Manhattan and the Euclidean distance have different inductive biases. The Euclidean distance tends to amplify the importance of a large difference between the values of a single attribute (coordinate) between two data instances. Intuitively, this makes the Euclidean distance more sensitive to noise in a single attribute than the Manhattan distance, because in the latter a large difference between two values of an attribute will have less impact than in the Euclidean distance.

The example of Table 2 is very simple, but is shows the point: the choice of a particular distance measure is important and affects the results of the AIS algorithm. There is no such thing as the “best” distance measure in general. This means that, in order to maximize the performance of an algorithm in a given dataset, one should carefully study the dataset in detail in a pre-processing phase of the knowledge discovery process, and then select the distance measure whose inductive bias is most suitable for that particular dataset. This is part of the problem-oriented approach advocated in this paper. Note that this is in contrast with the conventional “algorithm-oriented” approach of specifying the distance measure of the algorithm in a way independent of the data being mined. This pre-specification of a fixed distance measure might be appropriate for academic experiments where the algorithm is evaluated across a number of datasets, in order to show the algorithm’s robustness. However, if the goal is to maximize the performance of the algorithm in an important real-world dataset, where the results of the algorithm will be actually used for decision making in the real-world, then a more careful and justified choice of the distance function, tailored for the data being mined, should be made.

Table 2: Example of the influence of choice of distance measure in the results of an AIS

Data instances’ coordinates	Manhattan Distance to A	Euclidean distance to A
Antigen A: (0,0)	N/A	N/A
Antibody B: (4,4)	8	5.7
Antibody C: (6,1)	7	6.1

The importance of choosing a suitable affinity function (or distance measure) has also been emphasized in the work of Hart investigating different geometrical shapes of an antibody’s recognition area [Hart 2005], [Hart &

Ross 2004] – an investigation performed in the context of immune network-based AIS. One result of this investigation was that [Hart 2005] (p. 41): "*The results clearly show that the dynamics and size of the emergent networks are heavily influenced by the recognition region shape, and that the networks show varying ability to tolerate antigens over different ranges of recognition radius.*"

However, it should be noted that the results of Hart's investigation were produced by considering the performance of an immune network algorithm with randomly generated data. Hence, this investigation can be considered as an algorithm-oriented one, rather than the problem-oriented approach advocated in this paper.

4.2 A Critical Review of Affinity Issues in a Number of Existing Artificial Immune Systems

Table 3 compares the distance or affinity functions of a number of AIS. Out of all works quoted in Table 3, only the works of [Bezerra et al. 2004] and [Dasgupta et al. 1999] have proposed an affinity function particularly tailored for the data being mined. [Bezerra et al. 2004] used an affinity function based on the correlation coefficient, rather than on Euclidean distance. This choice was justified by the fact that the data being mined was a gene expression data set, consisting of expression levels measured across a number of different experimental conditions. The goal was to cluster genes according to their similarity with respect to expression levels. Note that in this case the use of Euclidean distance would be inappropriate, because it would be based on the magnitude of the differences in expression levels of two genes across different experimental conditions. Differences in magnitude are not important; what matters is the correlation. Two genes are considered to have similar expression patterns – and so should be assigned to the same cluster by the AIS clustering algorithm – to the extent that they are correlated in the sense that, the higher (lower) the expression level of the first gene in an experimental condition, the higher (lower) the expression level of the second gene on the same experimental condition.

[Dasgupta et al. 1999] used a data representation where a light spectrum was represented by a binary string where each bit was assigned a weight based on the corresponding spectroscopic band, and the affinity function was a bit-weighted one. Therefore, the affinity function exploited background knowledge about light spectra.

Hence, these two works provide good examples of how the choice of the affinity function should be dictated by the application domain and the nature of the data being mined.

The IFRAIS algorithm [Alves et al. 2004] uses an affinity function partially tailored for the target problem, as follows. An important point about this work is that it uses both an affinity function and a fitness function. The affinity function measures the degree of fuzzy matching between an antibody (fuzzy classification rule) and an antigen (training data instance). A data instance is deemed to satisfy a rule if the degree of fuzzy matching between the instance and the rule is greater than or equal to an affinity threshold.

Table 3: A Summary of Affinity Issues in a Number of Artificial Immune Systems

AIS	Kind of application or task	Distance or Affinity Function	Is the affinity function tailored for the data or application?
[Dasgupta et al. 2004]	Fault detection / classification	Euclidean distance	No
[Gonzales et al. 2002]	Anomaly detection / classification	Euclidean distance, affinity threshold based on median distance of k-neighbors	No
[Cserey et al. 2004]	Real-time processing of image sequences for surveillance	Hamming distance with “don’t care” elements	No.
[Dasgupta & Majumdar 2002]	Anomaly detection / classification	r-contiguous bits rule	No.
[Sarafijanovic & Le Boudec 2004]	Misbehaviour detection in mobile ad-hoc networks	Antibody must have “1” in every position where antigen has “1”	No
[Balthrop et al. 2002]	Network intrusion detection / classification	r-contiguous bit rule and a variant (r-chunks)	No
[Anchor et al. 2002]	Network intrusion detection / classification	At least 1 antigen must match hypervolume of antibody	No
[Hofmeyr & Forrest 1999]	Network intrusion detection / classification	r-contiguous bits rule	No
[Taylor & Corne 2003]	Fault detection/ classification in time series	Compared Euclidean and r-contiguous rules	No (but used representation tailored for time series)
[Dasgupta et al. 1999]	Spectra recognition	Hamming distance with weighted bits	Yes
[Bezerra et al. 2004]	Clustering of gene expression data	Correlation coefficient	Yes
[Timmis et al. 1999]	Clustering	Euclidean distance	No
[de Castro & von Zuben 2001 ; 2002b]	Clustering	Euclidean distance	No
[de Castro & von Zuben 2000a;2002a]	Digit recognition / classification	Hamming distance	No
[White & Garret 2003]	Digit recognition / classification	Hamming distance	No
[Watkins et al. 2004], [Watkins & Boggess 2002a]	Classification	Euclidean distance	No
[Sahan et al. 2005]	Medical diagnosis / classification	Euclidean distance with attribute weights	Partially (due to use of data-driven attribute weights)
[Alves et al. 2004]	Classification	Affinity based on fuzzy matching and fitness based specific for classification	Partially (due to data-driven adaptation of affinity threshold)
[Secker at al 2003]	Classification	Antibody must have *n* number of similar words to antigen	No
[Ayara et al 2005]	Error detection / Classification	Antibody and antigen must match in *r*-contiguous bits	Partially (takes into account time-based ordering of states)

In IFRAIS the fitness of an antibody is computed by a certain formula that measures the predictive accuracy of its rule, and that formula computes a global measure of the affinity of the antibody with respect to all antigens, rather than with respect to just one antigen. This formula is tailored for the classification task of data mining, but it is not tailored for the dataset being mined. However, the computation of the affinity (and so the fitness) is partially adapted to the dataset being mined in an automatic fashion. This is done by using an adaptive, data-driven procedure to adjust the affinity threshold. This still has the limitation that the basic structure of the affinity and fitness functions are the same for all datasets used in the experiments – different datasets might well require different affinity/fitness functions for optimal performance of the algorithm – but at least the adaptation of the affinity threshold for each dataset is a step in the direction of making the algorithm more adaptable to the data being mined.

The basic idea of an affinity threshold that varies according to the data being mined can also be found in other AIS, although in somewhat simplified forms. The AIRS algorithm [Watkins et al. 2004] also computes an affinity threshold value which is specific for the dataset being mined, since the value of that threshold is given by the average affinity value over all pairs of antigens (training data instances). In this case, however, the value of that threshold is computed once in the initialization procedure and kept fixed during the run of the algorithm, whilst in the IFRAIS algorithm the value of the affinity threshold is dynamically adapted during the run. In the AIS of [Gonzales et al. 2002] the affinity threshold is given by the median distance among the distances of the *k* nearest neighbors of an antibody, which also makes that threshold dynamically variable during the run, as in the IFRAIS algorithm. However, in IFRAIS the affinity threshold value is adapted in a way that directly maximizes the fitness of the antibody, which is not the case in the AIS of [Gonzales et al. 2002] and in AIRS.

As a brief aside, work in [Neal 2003] proposes an immune network algorithm that dynamically adjusts the size (and connectivity, thus affinity threshold) of the network in a data driven manner. However, this work was not included in Table 3 because it involves immune networks, which is out of the focus of this paper.

In any case, with the exception of the works of [Bezerra et al. 2004] and [Alves et al. 2004], the other works mentioned in Table 3 have in general used affinity functions which are popular in the AIS literature, such as the Hamming distance, the Euclidean distance or the r-contiguous bits rule. The fact that a work has used a conventional affinity function – rather than one tailored for the data being mined – is not necessarily a negative characteristic of that work. It is possible that a conventional affinity function be suitable – though not necessarily the ideal function – for the data being mined. For instance, if the data being mined consists of binary pixels in a pattern recognition application, it may be the case that a simple Hamming distance is a satisfactory affinity function, since there is no apparent higher-level structure or meaning in the data that could be exploited to design a tailored affinity function. In this sense the Hamming distance used by [De Castro & von Zuben 2000a], [De Castro & von Zuben 2002a], [White & Garret 2003] may be a satisfactory affinity function, since these works address the problem of digit recognition where the predictor attributes are binary pixels. In any case, it is also interesting to consider variations of a conventional Hamming distance, e.g. [Cserey et al. 2004] has used a modified kind of Hamming distance which also takes into account “don’t care” elements. In addition, even when working with a low-level data representation, it might be possible to exploit domain knowledge in order to design an affinity function tailored for the problem at hand. An example is the previously-mentioned work of [Dasgupta et al. 1999], where the bit weights are based on spectroscopic bands.

In the case of works using the Euclidean distance, it is important to remark that almost all the works mentioned in Table 3 are using an unweighted distance measure, assuming that all attributes have the same weight. A notable exception is the work of [Sahan et al. 2005]. Although this work is a step in the right direction, the attribute-weighting method used in this work is quite simple and it could be improved. In particular, as recognised by the authors, the method computes weights for attributes individually, ignoring attribute interactions; and it assumes that, if the standard deviation of the values of a predictor attribute within a class of instances is low, that attribute is relevant for predicting that class. This assumption seems unlikely to be true in many datasets.

The choice of an unweighted distance measure is likely to be a suboptimal choice in many application domains. This is particularly the case in the classification task, where the goal is to predict the class of a data instance. In this context, it is well-established that instance-based learning algorithms tend to be very sensitive to irrelevant attributes [Aha 1998], and it is normally the case that different attributes should have different weights, because they have different degrees of relevance for the prediction of the class of an instance. For instance, if we are trying to predict whether or not a bank customer should have a high credit, intuitively the attribute salary of the customer should have a much greater weight than the attribute gender. There is a large literature on intelligent methods for automatically computing attribute weights in the instance-based learning paradigm [Aha 1998], [Wettschereck et al. 1997], including evolutionary methods [Freitas 2002], and it is a pity that these kinds of methods are typically ignored in the AIS literature.

Another important remark is that several works mentioned in Table 3 have used an affinity function based on the r-contiguous bits rule, which has a strong positional bias [Freitas & Timmis 2003]. This kind of bias occurs due to a combination of two factors. First, when computing the degree of affinity between an antibody and an antigen according to the r-contiguous bits rule, the influence of the matching of one bit on the computed value of affinity depends on the position of that bit in the binary strings representing the antibody and the antigen. Second, although the attributes of the data being mined are encoded in a linear string representing an antibody or antigen in a certain order (from left to right), this encoding order is arbitrary and irrelevant from a data mining point of view. In general, in data mining applications the set of attributes describing the data to be mined is a set in the mathematical sense of a set, i.e., a collection of elements with no ordering and no duplication. Hence, in principle the affinity function should interpret the antibodies and antigens as sets of attributes, and compute a degree of affinity independent on the arbitrary position of the attributes in the strings representing the antibodies/antigens – unless specific characteristics of the data being mined suggest that a positional bias is desirable, such as in the work of [Ayara et al 2005], involving a kind of time series data.

As an example of positional bias, in [Hofmeyr & Forrest 1999] each antibody or antigen is a binary string representing three attributes, each of them represented by a certain number of bits, and the binary representations of those attributes is concatenated to produce a linear binary string. The r-contiguous bits rule is used in this work. The number of possible orderings of the three attributes in a linear string is 3! = 6. The choice of one particular attribute ordering to be used to compose the antibody/antigen strings is arbitrary in the target problem of network intrusion detection, where the attributes are the source IP address, the destination IP address and the service (port) by which two computers communicate. The value of affinity computed by the r-contiguous bits rule depends on the arbitrarily chosen ordering of the attributes in the antibody/antigen encoding, and so the result of the algorithm will depend on which of the six possible attribute orderings was chosen. This characterizes the positional bias of the system. Another example of an AIS using the r-contiguous bits rule, with its corresponding positional bias, is found in [Balthrop et al. 2002]. It should be pointed out, though, that in these projects the AIS also used the mechanism of permutation masks, which store different permutations of the bits representing the antibody. This mechanism helps to alleviate the positional bias of the r-contiguous bits rule [Garret 2005], since different bit permutations will cause the r-contiguous bits rule to produce different results, discovering different correlations among the bits in the antibody. In any case, if one does not want a positional bias, intuitively a simpler (and effective) solution would be to replace the r-contiguous bits rule by another affinity function which does not have a positional bias.

It should be noted that in general the choice of r-contiguous bits rule as the affinity function, with its positional bias, is not well justified in the works using this function quoted in Table 3. In other words, none of those works argued that such a strong positional bias was suitable for the data being mined or the target problem, so that it is quite possible that the strong positional bias of the r-contiguous bits rule is unsuitable for the kind of data mined in those works, again noting the possible exception of [Ayara et al 2005].

Note that the r-contiguous bits rule is based on a metaphor with a *physical* principle – the antibody-antigen matching in the natural immune system occurs in a physical 3D-space, where physical proximity is important.

Physical principles are not in general a good source of metaphors for AIS, because the latter works in a virtual space, free from physical properties. Logical principles are in general a better source of metaphors for AIS.

5 Immune Processes for AIS: Clonal Selection and Negative Selection

[de Castro & Timmis 2002] identify several major kinds of immune processes, inspired by their biological counterparts, which have been used in the design of AIS, namely: bone marrow models, negative selection, positive selection, clonal selection and immune network processes. Out of these, we focus on the clonal selection and the negative selection processes, which have been used extensively in AIS for data mining.

5.1 Clonal Selection

In the natural immune system, the basic idea of the clonal selection theory is as follows. When a B-cell's antibodies recognize an antigen with at least a certain degree of affinity, that B-cell is cloned in order to produce more antibodies with high affinity to that antigen. During its reproduction, the B-cell's clones are subject to a high rate of mutation, creating variations in the B-cell's antibodies. Due to a strong selective pressure, the new clones with higher affinity to the antigen will proliferate more than clones with lower affinity, so that this selective process usually results in B-cells having antibodies with a very high affinity to the antigen. Two important properties of the biological clonal selection process are: (a) the rate of cloning of each B-cell is proportional to the affinity to the antigen; (b) the rate of mutation of each B-cell during its reproduction is inversely proportional to the affinity to the antigen. Both properties are *logical* properties, independent of physical details, and they have been extensively used in the design of AIS.

In the context of AIS, a core issue in the use of the clonal selection principle is to decide which antibody(ies) should be cloned. This is related to the affinity issues discussed in Section 4, since in general the antibodies with highest affinity to an antigen are selected for cloning. However, in this subsection we analyze the problem of choosing which antibodies should be cloned from a broader perspective, going beyond affinity function issues. In particular, we consider important criteria to be used in order to choose the antibodies to be cloned. Regardless of how the affinity between an antibody and an antigen is computed, should the decision to clone an antibody be based on its affinity to a single antigen or based on its affinity with a number of antigens? Should that decision be based on just the antibody's affinity (to one or more antigens) or should some other additional criterion be used? These two questions are the topic of the next two Subsections.

5.1.1 The Clustering of Receptors for Activating Immune Cells and Its Significance for AIS

In the natural immune system, in general, one of the signals necessary to activate a B cell is the recognition of its cognate antigen. This recognition is performed by B cell receptors (BCRs) on the surface of the B cell. In general the activation of the B cell requires that *many* BCRs be brought close together on the B cell surface; this involves a “clustering” or “crosslinking” of BCRs [Sompayrac 2003]. This clustering of BCRs can be produced when BCRs bind to an epitope that is repeated many times on a single antigen or when BCRs bind to epitopes on antigens that are clumped together.

Although there are exceptions to this requirement of BCR clustering, the principle is important enough to deserve our attention here. In particular, it is important to abstract the logical principle behind the physical issues associated with such clustering. The lesson to be taken from this immune principle, from a logical point of view, is that the activation of an immune cell requires a substantial number of epitopes to be matched with the cell receptors, rather than a single epitope. In terms of AIS, this suggests that the activation of an artificial cell should likewise require many data items (artificial antigens or epitopes) to be matched to that cell. Intuitively, this makes sense from a statistical point of view, i.e., an artificial cell should not be activated by a matching to a single antigen or epitope, because that antigen/epitope could easily represent some noisy data or some spurious relationship in the data. By requiring that a B cell match many antigens/epitopes before it is activated, intuitively we would be making the AIS more robust, less sensitive to noisy data. It should be noted that our discussion in this paper is focused on clonal selection and negative selected-based systems. By contrast, works such as [Timmis and Neal 2001], [de Castro and Von Zuben 2002b], [Neal 2003] and [Hart 2005] have extracted various aspects of the immune network theory that have led to the development of ‘clustering type’ applications. Inherent within the immune network idea is that BCRs will interact with other BCRs that have a similar affinity (interaction is via idiotopes located on the BCR). In addition, the artificial B Cells contained within an immune network do require many interactions with different antigens to promote survival. However, our focus is not on these immune network models here, but an interested reader could consult the references provided.

5.1.2 The Two-Signal Mechanism for Activating Immune Cells and Its Significance for AIS

The natural immune system has several kinds of cells that need to be activated in order to help the fight against invader pathogens. This activation often involves a two-signal mechanism. The basic principle of this mechanism is quite generic, being used to activate several different kinds of immune cells. This generality makes the principle of two-signal activation intuitively attractive as a logical principle to be used, as a metaphor, in the design of AIS.

Let us first review, briefly, the mechanism of two-signal activation in four kinds of cells.

a) B cell activation – There are two kinds of B-cell activation, namely T-cell dependent activation and T-cell independent activation. In the former, the first signal required for activation is the recognition of cognate antigen by the B cell receptors (BCRs) on the surface of a B cell. This is called a specific signal, because it is provided specifically by the kind of antigen that the B cell can recognize – different B cells recognize different antigens. The second signal is a non-specific (independent of the antigen) co-stimulatory signal provided by a helper T cell. Typically this involves a contact between a protein called CD40L on a helper T cell and another protein called CD40 on the B cell. In T-cell independent activation, the first signal involves the recognition of special kinds of antigens – typically, microbial polysaccharides – by the BCRs. In this case the activation of the B cell does not require the co-stimulatory signal provided by a helper T cell. Once the first signal is received, the B cell proliferates. However, unlike T-cell dependent activation, after proliferation the B cell cannot secrete antibodies yet. This will happen only after the B cell receives a second signal, which is a cytokine like IFN-γ (interferon gamma) generated by the innate immune system.

b) T cell activation –The first signal required for T cell activation is the recognition of cognate antigen by the T cell receptors (TCRs) on the surface of a T cell. This is a specific signal, conceptually similarly to the first signal for B cell activation. Unlike B cells, however, T cells only recognize an antigen when it is presented by an Antigen Presenting Cell (APC). The basic idea is that the antigen is first chewed up into small pieces called peptides, which are then presented to T cells by special molecules on the surface of APCs called Major Histocompatibility Complex (MHC) molecules – "histo" means tissue. Hence, a TCR recognizes a "MHC-peptide complex", rather than just the antigen. There are three main kinds of APCs, namely dendritic cells, macrophages and activated B cells. The second signal required for T cell activation is a non-specific co-stimulatory signal provided by APCs, e.g. B7 proteins on the surface of the APC, which bind to CD28 proteins on the surface of the T cell. In addition, APCs secrete cytokine molecules that contribute to the co-stimulation of T cells. The basic principle of this two-signal activation mechanism is used for activating both killer T cells (cytotoxic lymphocytes) and helper T cells, but important details vary for those two types of T cells. Killer T cells almost always recognize class I MHC molecules. These molecules can be thought of as "billboards" that present, on the surface of the cell, fragments of proteins (peptides) that are being made inside the cell. Hence, class I MHC molecules present to T cells a "sampling" of endogenous proteins, giving T cells a chance to detect viral or other intracellular infections. By contrast, helper T cells almost always recognize class II MHC molecules. These molecules can be thought of as billboards that present extra-cellular peptides, representing a sampling of the environment around the cell. This gives T cells a chance to detect extra-cellular pathogens.

c) Macrophage activation – Macrophages can be activated in several ways. For the purposes of this subsection, it is enough to mention their activation by two signals provided by Th1, a kind of helper T cell that activates macrophages [Alberts et al. 2002]. The first signal used by a Th1 cell to activate a macrophage is IFN-γ, which binds to IFN-γ receptors on the surface of the macrophage. The second signal is a co-stimulatory protein called CD40 ligand on the Th1 cell, which binds to CD40 on the macrophage.

d) Natural killer (NK) "activation" – In order for a NK cell to kill its target cell, two signals seem to be necessary. The first one is a "kill" signal, which seems to involve interactions between proteins on the surface of the NK cell and carbohydrates on the surface of the target cell. The second signal, which is a "don't kill" signal, seems to be the expression of class I MHC molecules on the surface of the target. Hence, NK cells kill their targets only if the latter do not have class I MHC molecules on their surface.

Table 4: A Summary of Two-Signal Activation in Different Kinds of Immune Cells

Kind of cell to be activated	Signals required for activation	
	First signal	Second (co-stimulatory) signal
B cell (T cell-dependent activation)	Recognition of antigen by BCR	Binding between proteins on the B cell and on the helper T cell
B cell (T cell-independent activation)	Recognition of antigen by BCR	a cytokine such as IFN-γ, generated by the innate immune system
T cell	Recognition, by TCR, of the MHC-peptide complex presented by Antigen Presenting Cells (APCs)	Mainly binding between proteins on the T cell and on the APC, and also cytokines secreted by APCs
Macrophage	IFN-γ secreted by helper T cell Th1	Binding between proteins on the Th1 cell and on the macrophage
Natural killer	Interactions between proteins on the NK and carbohydrates on the target	Absence of class I MHC molecules on the surface of the target cell

A summary of the previous discussion is shown in Table 4. It is clear that, although the details vary depending on the kind of cell being activated, the principle of two-signal activation is generic enough to have been adopted in nature for the activation of several different kinds of cells of the immune system. Hence, it is important to understand the logic behind this principle. The two-signal activation principle is a "fail-safe" mechanism. By using this principle, in general the decision to activate an immune cell is not made by a single cell nor based on a single signal. Rather, two signals are necessary for activation, and in general at least one of the signals is provided by a kind of immune cell different from the one that is to be activated. An exception is the activation of natural killer cells, where the second signal is not provided by an immune cell, since it involves the absence of a molecule on the target cell. However, even in this case, the second signal is very different in nature from the first signal, which characterizes the "fail-safe" nature of this activation mechanism. This logical principle makes the two-signal activation mechanism attractive as a generic principle to be used in the design of AIS algorithms.

5.1.3 A Critical Review of the Use of Natural Immune Cell Activation Principles in Existing AIS

Table 5 compares several AIS with respect to whether or not they follow two logical principles used by the natural immune system for cell activation in a given clonal selection-based AIS. More precisely, for each AIS Table 5 mentions: (a) the kind of application or task addressed by the AIS; (b) the main kinds of immune processes used by the AIS; and (c) whether the AIS uses (implicitly or explicitly) the previously-discussed principles of antigen clustering or two-signal activation as a requirement for cloning an artificial immune cell.

All AIS mentioned in Table 5 use the clonal selection process. This was a pre-requisite for including an AIS in that table, since the last two columns of the table make sense particularly for clonal selection-based AIS. In Table 5 the term "clonal selection" is being used in a broad sense to refer to any kind of algorithm where the fittest artificial immune cells tend to be selected for cloning, regardless of whether or not mutation with a rate inversely proportional to fitness is applied to the clones. Mutation issues are irrelevant for our discussion in this subsection, as here we focus only on the criteria used to decide whether or not an artificial immune cell should be cloned. This has allowed us to include in the table some works which are predominantly based on negative selection, but where the fittest cells are cloned, such as the work of [Dasgupta et al. 2004].

Let us start with the analysis of how (if at all) existing AIS use the principle of two-signal activation. Recall the previously-described rationale for this principle as a "fail-safe" mechanism for the activation and cloning of an immune cell. Only four of the AIS mentioned in Table 5 – namely, the works of [Sarafijanovic & Le Boudec 2004], [Kim & Bentley 2002], [Secker et al 2003] and [Ayara et al 2005] – use the principle of two-signal activation. We will examine two of these. In [Kim & Bentley 2002] the second signal is manually provided by the user, whilst in [Sarafijanovic & Le Boudec 2004] the second signal is generated in an automatic fashion. [Sarafijanovic & Le Boudec 2004] have proposed a virtual thymus that includes not only the popular negative selection process, but also more sophisticated concepts such as the use of a danger signal and a short time of antigen presentation in the thymus. The danger theory metaphor seems very appropriate for the target application domain of misbehaviour detection in mobile ad-hoc networks. Indeed, in this work the danger signal has a spatial-temporal nature, related to a loss of a packet in the network. In any case, the fact that just a minority of the AIS mentioned in Table 5 use the principle of two-signal activation suggests that the potential of this principle has been under-explored in the AIS literature.

Table 5: A Summary of Whether or Not AIS Follow Two Immune-Inspired Principles for Deciding Whether or Not to Clone an Artificial Immune Cell

AIS	Kind of application or task	Immune Process	Antigen clustering?	Two-signal activation ?
[Sarafijanovic & Le Boudec 2004]	Misbehaviour detection in mobile ad-hoc networks	Thymus model (including negative selection), clonal selection	Yes, explicitly	Yes (danger signal)
[Kim & Bentley 2002]	Network intrusion detection / classification	Negative selection, clonal selection	Yes, implicitly	Yes (manual signal)
[Dasgupta et al. 2004]	Fault detection / classification	Negative selection, clonal selection	Yes, implicitly	No
[Sahan et al. 2005]	Medical diagnosis / classification	Clonal selection	No	No
[de Castro & von Zuben 2000a; 2002a]	Digit recognition / classification	Clonal selection	No	No
[White & Garret 2003]	Digit recognition / classification	Clonal selection	No	No
[de Castro & von Zuben 2000b; 2001; 2002b]	Clustering	Clonal selection, immune network	No	No
[Watkins et al. 2004], [Watkings & Boggess 2002a]	Classification	Clonal selection	No	No
[Alves et al. 2004]	Classification	Clonal selection	Yes, implicitly	No
[Secker et al 2003]	Classification	Clonal selection	No	Yes (manual signal)
[Ayara et al 2005]	Classification	Clonal selection	No	Yes (semi-automated)

Let us now turn to a detailed discussion on the principle of antigen clustering. An AIS based on this principle is described in [Sarafijanovic & Le Boudec 2004]. The authors explicitly emphasize the metaphor with antigen clustering in the natural immune system. The principle of antigen clustering is also used in [Kim & Bentley 2002], [Dasgupta et al. 2004], and [Alves et al. 2004] although in these works the principle was used implicitly. That is, these works did not emphasize the metaphor with antigen clustering in the natural immune system, but the cloning of antibodies does depend on the strength of matching with a number of (not just one) antigens.

It should be noted that the principle of antigen clustering is not followed in several well-known clonal selection-based AIS such as CLONALG [de Castro & von Zuben 2000a; 2002a], aiNet [de Castro & von Zuben 2000b; 2001; 2002b], and AIRS [Watkins et al. 2004], [Watkings & Boggess 2002a]. Let us analyse why this is the case. First of all, it should be noted that both CLONALG and AIRS perform a classification task, whilst AiNet performs a clustering task – here we are referring to the versions of CLONALG and aiNet performing the tasks of pattern recognition/classification and clustering, respectively, and not their versions for multi-modal optimisation [de Castro & von Zuben 2002a], [de Castro & Timmis 2002b]. AIRS was designed specifically for classification, whilst CLONALG seems to have been designed as a more generic kind of AIS – and indeed it has a variant for optimisation, unlike AIRS. Despite these important differences, it is striking that CLONALG, AIRS and aiNet share a very important design feature. The clonal selection principle is incorporated within the pseudocode of these AIS in essentially the same way, as shown by the excerpt of pseudocode in Algorithm 1, at a high level of abstraction. That excerpt of pseudocode describes particularly well the corresponding part of the pseudocodes of CLONALG and aiNet. In the case of AIRS, a more precise description would be to replace the line “Select *n* highest affinity antibodies” with the line “Select the highest affinity memory cell belonging to the same class as the antigen”. However, this is a relatively small difference for the purposes of our discussion here, and of course *n* can be set to 1, so we opted for representing that part of the pseudocode of these three AIS in a single Algorithm, for the sake of simplicity.

```
FOR EACH antigen
    FOR EACH antibody
        Compute affinity between antigen and antibody
    END FOR EACH antibody
    Select n highest affinity antibodies
    Clone the n selected antibodies
…
END FOR EACH antigen
```

Algorithm 1: Excerpt of pseudocode of CLONALG, aiNet and AIRS, at a high level of abstraction

We emphasize that the excerpt of pseudocode shown in Algorithm 1 is meant just to show that these three AIS loop over the antigens and the antibodies (or memory cells) in order to choose the highest affinity antibodies to be cloned, and not to show other aspects of the AIS which are irrelevant for the current discussion. Note that in the original description of these AIS the loop over the antibodies is not shown explicitly, but rather implicitly in an instruction like “present the antigen to all antibodies” or “find the (n) antibodies closest to the antigen”. Showing this loop explicitly, as in Algorithm 1, helps to highlight the important point that this excerpt of pseudocode performs an *external* loop over the antigens, and an *internal* loop over the antibodies. As a result, the affinity of an antibody is computed with respect to just *one* antigen.

From the viewpoint of a metaphor with the natural immune system, this means that the principle of antigen clustering is not followed. From a problem-oriented viewpoint, this means that the computation of an antibody's affinity is based on a single data instance. If the goal of computing the affinity is to decide whether or not the current antibody should be cloned, as it is the case in CLONALG, AIRS and AiNet, computing affinity based on a single antigen seems a somewhat risky strategy, since there is very little statistical support for such a decision. By contrast, if the system computed the antibody's affinity with respect to a number of antigens, then the principle of antigen clustering would be followed, and a more statistically sound decision could be made about whether or not the current antibody should be cloned. That decision would be more statistically sound because it would be based on the global value of the antibody's affinity with respect to all antigens. This global affinity value will be here referred to as simply the "fitness" of an antibody, by analogy with the common use of the term in evolutionary algorithms – where usually the fitness of an individual is computed with respect to all data instances [Freitas 2002]. Such distinction between an antibody's affinity to a single antigen and the antibody's fitness with respect to all training antigens is found in IFRAIS [Alves et al. 2004], an AIS that, like AIRS, was specifically designed for classification.

Note that one way of computing the fitness of an antibody in CLONALG, AIRS and aiNet could be obtained by simply swapping the order of the loops over antigens and antibodies in the excerpt of pseudocode shown in Algorithm 1, which would produce the new excerpt of pseudocode shown in Algorithm 2. However, this would require, of course, significant modifications in other parts of the pseudocode of these AIS. Such modifications would not guarantee an improvement in the performance of these AIS, but they seem worth trying, considering the previously-discussed rationale for the principle of antigen clustering.

Actually, there is one more motivation to design and evaluate variants of CLONALG, AIRS and aiNet following the excerpt of pseudocode shown in Algorithm 2. This is a data mining-oriented motivation, and it is the fact that, in the current version of these AIS, following the excerpt of pseudocode shown in Algorithm 1, the results produced by these AIS tend to vary according to the order in which antigens (data instances) are used in the external loop. After all, in the *i*th iteration of the external loop, when the *i*th antigen is presented to the current antibody population, the antibodies included in that population have been produced as the result of an evolution guided by the antigens presented at iterations 1, …, $i - 1$. This dependency of the results in the ordering of presentation of data instances is a kind of "instance-ordering bias", and is not necessarily a bad thing, but it does introduce one more source of non-determinism in the AIS, without any clear advantage. Note that instance-based learning or rule induction algorithms usually treat the training instances as an unordered set of instances.

```
FOR EACH antibody
    FOR EACH antigen
        Compute affinity between antigen and antibody
    Compute antibody’s fitness based on total affinity with respect to all antigens
    ….
    END FOR EACH antibody
…
END FOR EACH antigen
```

Algorithm 2: Excerpt of pseudocode modifying CLONALG, AIRS, aiNet to use the antigen clustering principle

5.2 Negative Selection

In the natural immune system, negative selection is a process that occurs in the thymus, the organ where T-cells mature. T-cells that match self are eliminated before they mature in the thymus. Hence, in general the mature T-cells leaving the thymus will not match self, and will therefore match only non-self.

The basic idea of the negative selection process as used in AIS is shown in the pseudocode of Algorithm 3. Given a set of “normal” data instances – the *self* – as input, the system performs a loop where, at each iteration, it randomly generates immature immune cells (detectors) and tries to match each cell with all the instances in the self. If that immune cell matches at least one instance in the self it is discarded, otherwise it is promoted to a mature cell and is output by the algorithm. This iterative process is repeated until almost all the non-self space has been covered by the generated immune cells, or another stopping criterion is met. From a data mining point of view, this phase is a kind of “training phase”, whose goal is to generate a set of mature immune cells that should be able to detect non-self data instances only. Next, during the “test phase”, if a given test instance matches a mature immune cell, that instance is classified as non-self (anomaly); otherwise it is classified as self (normal).

```
Input: a set of “normal” data instances, called the self (S)
Output: a set of “mature” immune cells that do not match any instance in S
REPEAT
    Randomly generate an “immature” immune cell
    Measure the affinity (similarity) between this immune cell and each instance in S
    IF the affinity between the immune cell and at least one instance in S is greater than a user-defined threshold
        THEN discard this immune cell
        ELSE output this immune cell as a “mature” immune cell
UNTIL stopping criterion
```

Algorithm 3: Pseudocode of the Negative Selection Process

Since the pioneering work of [Forrest et al. 1994], there has been an extensive research on negative selection-based AIS, most of them applied to the “anomaly detection” task, according to the terminology used in the AIS literature. Recall however, that the use of this term is somewhat misleading in the context of data mining, as explained in Section 2. In other words, several of those negative selection-based works are effectively performing a task which is similar to the conventional classification task, as explained in that section. In this context, the use of a negative selection-based AIS is problematic, presenting disadvantages such as [Freitas & Timmis 2003]:

(a) Immune cells (detectors) are randomly generated throughout the pseudocode of Algorithm 3, a method which is not adaptive and does not use any information in the set of self instances to guide the search;

(b) The lack of mechanisms to minimize overfitting;

(c) The evaluation of the negative selection-based AIS for “anomaly detection” is often based on standard benchmarks for evaluating classification algorithms. In particular, some negative selection-based AIS have been evaluated in a test set containing instances of both “normal” and “anomalous” classes, but a training set containing only instances of the normal class – in order to allow the application of negative selection process, which is trained with normal instances only – as discussed in Subsection 2.3.

The latter item is essentially a criticism of the application of a negative selection-based AIS to a scenario where both normal and anomalous classes were available in the original dataset as a whole and so both classes could be included in the training set, but only the normal class was included in order to justify the use of the negative selection algorithm. That is, an algorithm-oriented approach, rather than the problem-oriented approach has been adopted.

It should be noted that there is a counter-argument to the latter criticism. The counter-argument is that there are problems where indeed just the normal class is available in the training set, and so a standard classification algorithm – trained with more than one class – is not directly applicable. At first glance, this could seem a natural scenario for a negative selection-based AIS, given its ability in training with just one class.

There are two replies to this counter-argument. First, it is important to evaluate the negative selection-based AIS on real-world problems where indeed the original training set contains only instances of the class normal, rather than artificially remove the anomalous class from the training set. Second, even limiting the training set to have just one class, it is not clear that a negative selection-based AIS would perform well by comparison with other machine learning or data mining techniques also trained on just one class. Actually, in recent work evaluating a negative selection-based AIS [Stibor et al. 2005], the performance of that algorithm was shown to be inferior to the performance of two statistical data mining techniques in a problem where all methods were trained with a single class only. In this paper, the authors used a real-valued negative selection algorithm [Ji and Dasgupta 2004] and compared it to a one-class support vector machine and a Parzen Window algorithm. Each of the algorithms was tested on a subset of the well-known KDD Cup data from 1999. Results found that the one-class support vector machine was significantly better at identifying intrusions; in fact the negative selection approach averaged only around 3% detection rate, compared to the one-class support vector machines 98% detection rate [Stibor et al. 2005].

Concerning the performance of negative selection-based AIS for classification and related problems – such as the kind of "anomaly detection" problem addressed in the AIS literature – the general conclusion of [Hart & Timmis 2005] (p. 488) still seems to hold:

"*… at present it is not clear from the literature that the immune approach [negative selection-based AIS] offers anything. It is necessary to use two classes of data to train and tune the system, a high false positive rate seems to blight systems and the computational complexity of generating detectors seems prohibitive in large dimensional data sets*."

Does that mean that the metaphor with the natural immune system's process of negative selection is a useless metaphor for designing an AIS? No. The problem is not the metaphor per se, but the way the metaphor has been used. In the natural immune system, the process of negative selection is just one out of many processes going on at the same time in the organism. Even within the confined region of the thymus, T-cells are subject not only to negative selection but also to positive selection. After leaving the thymus, a T-cell still needs to collaborate with other immune cells in order to have a useful effect in the organism. In other words, the negative selection process per se is simply not adaptive. It is, however, an important process in the context of a bigger picture, the immune system as a whole, due to the interaction of this process with several other processes in the organism.

The realization that the negative selection process per se is not adaptive and is essentially a kind of random search should not be viewed as negating the usefulness of that process. An analogy with evolutionary algorithms is appropriate. Taken in isolation, mutation is clearly a non-adaptive, *random*-search like operator; and a conventional crossover *randomly* mixes the contents of two individuals. However evolutionary algorithms using these operators in general work fine, because these operators are not used in isolation. They are used in conjunction with a selection operator, which introduces selective pressure in the solutions randomly produced by mutation and crossover. It is the combination of selection, mutation and crossover that makes an evolutionary algorithm work well.

To summarize, negative selection, per se, is a kind of random search procedure. In order to design adaptive, intelligent AIS using the negative selection principle, that principle should not be used in stand-alone mode, but should rather be combined with more adaptive processes. Actually, this research direction is being followed in some works in the AIS literature, see e.g. [Kim & Bentley 2002], [Kim et al. 2003], [Gonzales & Dasgupta 2002], [Anchor et al. 2002], [Gonzales et al. 2002]. However, it is difficult to evaluate the results of these works, since most of them do not compare the result of the proposed AIS with the result of a conventional data mining algorithm. An exception is the aforementioned work of [Stibor et al. 2005]. In any case, much more research is required comparing the aforementioned works with conventional data mining techniques in challenging real-world datasets. A similar comment has also recently been made by [Garret 2005]: "*Comparison [of negative selection] with many more techniques is required…*".

6 Conclusions and Future Research

This paper has advocated the use of a problem-oriented approach for designing an AIS for data mining, by contrast with the kind of algorithm-oriented approach often followed in the AIS literature. This algorithm-oriented approach is perfectly reasonable in some scenarios, particularly in academic research. However, in scenarios where the goal is to develop an AIS that is at least competitive with state-of-the-art data mining techniques in an important real-world application, a problem-oriented approach is necessary. In such scenarios we need to tailor the design of the AIS for the data being mined or the application domain, because every AIS for data mining – like every other kind of data mining algorithm – has an inductive bias, and it is a well-established fact that every inductive bias is suitable for some datasets or application domains and unsuitable for others [Schaffer 1994], [Rao et al. 1995], [Michie et al. 1994], [Lim et al. 2000].

In order to design an AIS with an inductive bias suitable for the target data, it is of course crucial to understand the inductive biases of the major components of an AIS, such as its representation(s), affinity function(s) and immune process(es). This is a hard task, which was started with our preliminary work on this topic, described in [Freitas & Timmis 2003]. This paper is a major extension of that preliminary work, and hopefully it represents a significant contribution towards the challenging goal of understanding the inductive biases of AIS. Another contribution of this paper was a discussion about several important lessons that can be taken from the natural immune system to help us to design more adaptive AIS.

These two kinds of contribution can be summarized by briefly listing some important conclusions of the paper. Let us do this by listing a set of limitations found in existing AIS for data mining and the corresponding suggestions for future research in order to mitigate that limitation. These limitations and corresponding suggestions are listed in Table 6. We stress that the limitations and suggested research directions mentioned in the table refer only to AIS for data mining, and not for other kinds of application such as optimization and control.

The 10 suggested research directions in Table 6 vary considerably in the degree of difficulty, and so in the degree of rewarding too. Overall the most challenging – and also very rewarding – research direction seems to be the development of the first AIS with hybrid knowledge representations and capable of dynamically adapting the knowledge representations to the data during its run – a very open research area at the present.

Table 6: A Summary of Limitations of AIS for Data Mining and Corresponding Suggested Research Directions

Limitations in Existing AIS for Data Mining	**Suggested Research Direction**
Most AIS use a data representation consisting of numerical data only	Develop more AIS with representations for categorical data, as well as hybrid numerical/categorical data, tailoring the AIS representation for the data, rather than ignoring non-numerical data
Most AIS use the instance-based representation: a single and fixed representation	Develop more AIS with alternative knowledge representations, such as rule-based representation (including first-order logic representations) or probability-based representations (e.g. Bayesian networks)
	Develop the first AIS with hybrid knowledge representations and capable of dynamically adapting the knowledge representations to the data during their run
Most AIS use standard affinity functions, mainly for binary or numerical data	Develop more AIS with affinity function tailored for the data being mined, rather than just using a standard affinity function
	In AIS with instance-based representation for the classification task, develop affinity function considering attribute weights
	Avoid using the r-contiguous bits rule or another representation with positional bias, unless it is clear this is suitable for the data being mined
Several AIS based on the process of clonal selection do not follow the principle of antigen clustering and have an “instance-ordering” bias	Modify the corresponding AIS to follow the principle of antigen clustering; hopefully making them more robust to noisy data and removing their instance-ordering bias
Several AIS based on the process of clonal selection do not follow the principle of two-signal activation	Modify the corresponding AIS to follow the principle of two-signal activation, so implementing a fail-safe mechanism on the artificial clonal selection process
Several AIS based on the process of negative selection use that process only, which is a kind of random search	Extend the corresponding AIS to combine the negative selection process with other immune processes or with conventional data mining algorithms
Most AIS are evaluated in either artificial data or public domain datasets which are not very challenging	Evaluate AIS in many more challenging real-world datasets, e.g. datasets on which conventional data mining algorithms do not obtain a high classification accuracy, or very high dimensional datasets (with hundreds of attributes); and perform much more comparisons with conventional data mining algorithms

The motivation for hybrid representations is twofold. The first motivation is an analogy with the natural immune system, which certainly uses many kinds of immune cell (corresponding to many different kinds of representation) to achieve a high level of robustness and adaptability. Second, there are several examples of successful machine learning and data mining systems using hybrid representations (see Section 3.4). However, in some works discussed in Section 3.4, e.g. [Ting 1994] and [Carvalho & Freitas 2004], the decision about which training instances are better covered by which kind of representation (decision tree or instance-based or rule) is made in a simple way, by using the result of a run of a decision tree algorithm. The design of a hybrid representation system that decides which representation should be used to cover which training instance in a much more dynamic way, continuously making these decisions during its run, is more challenging, but there are machine learning systems with this flexibility. One example is the RISE algorithm [Domingos 1995], which dynamically decides whether to use an instance-based or rule-based representation to cover training instances.

Another suggested research direction mentioned in Table 6 that is worth highlighting here is the need for evaluating AIS in many more challenging real-world datasets and performing much more comparisons with conventional data mining algorithms. This is very important, because artificial datasets and public domain datasets have obvious limitations. Even if public domain datasets are derived from real-world data, sometimes those datasets have been already carefully preprocessed, including a selection of attributes particularly useful for the target data mining task, which makes them less challenging than true real-world applications where preparing the data for data mining purposes is a major part of the challenge of knowledge discovery. As examples of data mining application areas where such challenging datasets can be found relatively easy, one can quote bioinformatics and text mining, two very active research areas in data mining.

References

[Aha 1997] D.W. Aha (Ed.) *Artificial Intelligence Review – special issue on lazy learning, 11(1-5)*, June 1997.

[Aha 1998] D.W. Aha. Feature weighting for lazy learning algorithms. In: H. Liu and H. Motoda (Eds.) *Feature Extraction, Construction and Selection: a data mining perspective*, pp. 13-32. Kluwer, 1998.

[Alberts et al. 2002] B. Alberts, A. Johnson, J. Lewis, M. Raff, K. Roberts and P. Walter. *The Molecular Biology of the Cell*. 4th Ed. Garland Science, 2002.

[Alves et al. 2004] R.T. Alves, M.R. Delgado, H.S. Lopes and A.A. Freitas. An artificial immune system for fuzzy-rule induction in data mining. *Proc. Parallel Problem Solving from Nature (PPSN-2004), LNCS 3242*, pp. 1011-1020, Springer, 2004.

[Anchor et al. 2002] K.P. Anchor, P.D. Williams, G.H. Gunsch, G.B. Lamont. The computer defense immune system: current and future research in intrusion detection. *Proc. Congress on Evolutionary Computation (CEC)*. IEEE Press, 2002.

[Ayara et al 2005]. M. Ayara, J. Timmis, R. De Lemos, and S. Forrest. Immunising Automated Teller Machines. In: *Artificial Immune Systems: Proc. Fourth Int. Conf. (ICARIS-2005). LNCS 3627*, pp. 404-417. Springer, 2005.

[Balthrop et al. 2002] J. Balthrop, F. Esponda, S. Forrest and M. Glickman. Coverage and generalization in an artificial immune system. *Proc. Genetic and Evolutionary Computation Conf. (GECCO)*, pp. 3-10. Morgan Kaufmann, 2002.

[Back 2000] T. Back. Binary strings. In: T. Back, D.B. Fogel and T. Michalewicz (Eds.) *Evolutionary Computation 1: Basic Algorithms and Operators*, pp. 132-135. Institute of Physics Publishing, 2000.

[Bezerra et al. 2004] G.B. Bezerra, L.N. De Castro, F.J. Von Zuben. A hierarchical immune network applied to gene expression data. In: *Artificial Immune Systems: Proc. Third Int. Conf. (ICARIS-2004). LNCS 3239*, pp.14-27. Springer, 2004.

[Carvalho & Freitas 2004] D. R. Carvalho and A. A. Freitas. A hybrid decision tree/genetic algorithm method for data mining. *Special issue on Soft Computing Data Mining – Information Sciences 163(1-3)*, pp. 13-35. 14 June 2004.

[Castro et al. 2005] P.D. Castro, G.P. Coelho, M.F. Caetano and F.J. von Zuben. Designing ensembles of fuzzy classification systems: an immune-inspired approach. In: *Artificial Immune Systems: Proc. Fourth Int. Conf. (ICARIS-2005). LNCS 3627,* pp. 469-482. Springer, 2005.

[Chao & Forrest 2003] D.L. Chao and S. Forrest. Information immune systems. *Genetic Programming & Evolvable Machines, 4(4),* pp. 311-331, Dec. 2003.

[Clare & King 2002] A. Clare and R.D. King. Machine learning of functional class from phenotype data. *Bioinformatics 18(1),* 160-166. 2002.

[Cserey et al. 2004] G. Cserey, W. Porod, T. Roska. An artificial immune system based visual analysis model and its real-time terrain surveillance application. In: *Artificial Immune Systems: Proc. Third Int. Conf. (ICARIS-2004). LNCS 3239,* pp. 250-262. Springer, 2004.

[Dasgupta 1999] D. Dasgupta. *Artificial Immune Systems and their Applications*. Springer, 1999.

[Dasgupta & Majumdar 2002] D. Dasgupta and N.S. Majumdar. Anomaly detection in multidimensional data using negative selection algorithm. *Proc. Congress on Evolutionary Computation (CEC)*, pp. 1039-1044. IEEE Press, 2002.

[Dasgupta et al. 1999] D. Dasgupta, Y. Cao and C. Yang. An immunogenetic approach to spectra recognition. *Proc. 1999 Genetic and Evolutionary Computation Conference (GECCO-99),* pp. 149-155. Orlando, FL, USA, July 1999.

[Dasgupta et al. 2004] D. Dasgupta, K. KrishnaKumar, D. Wong, M. Berry. Negative selection algorithm for aircraft fault detection. In: *Artificial Immune Systems: Proc. Third Int. Conf. (ICARIS-2004). LNCS 3239,* pp. 1-13. Springer, 2004.

[de Castro & Timmis 2002] L.N. de Castro and J. Timmis. *Artificial Immune Systems: a new computational intelligence approach*. Springer, 2002.

[de Castro & Timmis 2002b] L.N. de Castro and J. Timmis. An artificial immune network for multimodal function optimization. *Proc. Congress on Evolutionary Computation (CEC-2002),* pp. 699-704. IEEE Press, 2002.

[de Castro & von Zuben 2000a] L.N. de Castro and F.J. von Zuben. The clonal selection algorithm with engineering applications. *Proc. GECCO-00's Workshop on Artificial Immune Systems and Their Applications*. 2000.

[de Castro & von Zuben 2000b] L.N. de Castro and F.J. von Zuben. An evolutionary immune network for data clustering. *Proc. IEEE Brazilian Symposium on Artificial Neural Networks (SBRN-2000),* pp. 84-89. IEEE.

[de Castro & von Zuben 2001] L.N. de Castro and F.J. von Zuben. Immune and neural network models: theoretical and empirical comparisons. *Int. Journal of Computational Intelligence and Applications, 1(3),* pp. 239-257, 2001.

[de Castro & von Zuben 2002a] L.N. de Castro and F.J. von Zuben. Learning and optimization using the clonal selection principle. *IEEE Trans. on Evolutionary Computation 6(3),* pp.239-251, June 2002.

[de Castro & von Zuben 2002b] L.N. de Castro and F.J. von Zuben. aiNet: an artificial immune network for data analysis. In: H.A. Abbass, R.A. Sarker, C.S. Newton (Eds.) *Data Mining: a Heuristic Approach*, 231-259. Idea Group, 2002.

[Domingos 1995] P. Domingos. Rule Induction and Instance-Based Learning: a unified approach. *Proc. 14th Int. Joint Conf. on Artificial Intelligence (IJCAI-95).* 1995.

[Fayyad et al. 1996] U.M. Fayyad, G. Piatetsky-Shapiro, P. Smyth. From Data Mining to Knowledge Discovery: an Overview. In: U.M. Fayyad et al (Eds.) *Advances in Knowledge Discovery and Data Mining,* 1-34. AAAI/MIT, 1996.

[Flach 2003] P. Flach. The geometry of ROC space: understanding machine learning metrics through ROC isometrics. *Proc. 20th Int. Conf. on Machine Learning (ICML-2003).* Washington DC, USA, 2003.

[Flach 2004] P. Flach. Tutorial on the many faces of ROC analysis in machine learning. *Tutorial notes of the 21st Int. Conf. on Machine Learning (ICML-2004).*

[Forrest et al. 1994] S. Forrest, A.S. Perelson, L. Allen and R. Cherukuri. Self-nonself discrimination in a computer. *Proc. IEEE Symp. On Research in Security and Privacy,* pp. 202-212. 1994.

[Freitas 2000] A.A. Freitas. Understanding the crucial differences between classification and discovery of association rules - a position paper. *ACM SIGKDD Explorations, 2(1),* pp. 65-69. ACM, 2000.

[Freitas 2002] A. A. Freitas. *Data Mining and Knowledge Discovery with Evolutionary Algorithms.* Springer-Verlag, 2002.

[Freitas & Timmis 2003] A.A. Freitas and J. Timmis. Revisiting the foundations of Artificial Immune Systems: a problem-oriented perspective. In: J. Timmis, P. Bentley, E. Hart (Eds.) *Artificial Immune Systems (Proc. ICARIS-2003), LNCS 2787,* 229-241. Springer, 2003.

[Garret 2005] S.M. Garret. How do we evaluate artificial immune systems? *Evolutionary Computation 13(2),* Summer 2005.

[Gonzales and Dasgupta 2002] F.A. Gonzales and D. Dasgupta. An immunogenetic technique to detect anomalies in network traffic. *Proc. Genetic and Evolutionary Computation Conference (GECCO-2002),* pp. 1081-1088. Morgan Kaufmann, 2002.

[Gonzales et al. 2002] F. Gonzales, D. Dasgupta and R. Kozma. Combining negative selection and classification techniques for anomaly detection. *Proc. Congress on Evolutionary Computation (CEC),* pp. 705-710. IEEE Press, 2002.

[Greensmith et al 2005]. J. Greensmith, Aickelin, U. and Cayzer, S. *Introducing Dendritic Cells as a Novel Immune-Inspired Algorithm for Anomaly Detection.* In: *Artificial Immune Systems: Proc. Fourth Int. Conf. (ICARIS-2005). LNCS 3627,* pp. 153-167. Springer, 2005.

[Hart 2005] E. Hart. Not all balls are round: an investigation of alternative recognition-region shapes. In: *Artificial Immune Systems: Proc. Fourth Int. Conf. (ICARIS-2005). LNCS 3627,* pp. 29-42. Springer, 2005.

[Hart & Ross 2004] E. Hart and P. Ross. Studies on the implications of shape-space models. In: *Artificial Immune Systems: Proc. Third Int. Conf. (ICARIS-2004). LNCS 3239*, pp. 413-426. Springer, 2004.

[Hart & Timmis 2005] E. Hart and J. Timmis. Application areas of AIS: the past, the present and the future. In: *Artificial Immune Systems: Proc. Fourth Int. Conf. (ICARIS-2005). LNCS 3627,* pp. 483-498. Springer, 2005.

[Hofmeyr & Forrest 1999] S.A. Hofmeyr and S. Forrest. Immunity by design: an artificial immune system. *Proc. Genetic and Evolutionary Computation Conference (GECCO-99)*. Morgan Kaufmann, 1999.

[Ji and Dasgupta 2004] Z. Ji and D. Dasgupta. Real Valued Negative Selection Algorithm with Variable Sized Detectors. *Proc. of Genetic and Evolutionary Computation (GECCO-2004), LNCS 3102*. pp.287-298. Springer, 2004.

[Kim & Bentley 2002] J. Kim and P.J. Bentley. Towards an artificial immune system for network intrusion detection: an investigation of dynamic clonal selection. *Proc. Congress on Evolutionary Computation (CEC-2002)*. IEEE Press, 2002.

[Kim et al. 2003] J. Kim, A. Ong and R.E. Overill. Design of an artificial immune system as a novel anomaly detector for combating financial fraud in the retail sector. *Proc. Congress on Evolutionary Computation (CEC-2003)*. IEEE Press, 2003.

[Langley 1996] P. Langley. *Elements of Machine Learning*. Morgan Kaufmann, 1996.

[Liao et al. 1998] T.W. Liao, Z. Zhang, C.R. Mount. Similarity measures for retrieval in case-based reasoning systems. *Applied Artificial Intelligence 12,* pp. 267-288, 1998.

[Lim et al. 2000] T.S. Lim, W.Y. Loh and H.S. Shih. A comparison of prediction accuracy, complexity and training time of thirty-three old and new classification algorithms. *Machine Learning 40,* pp. 203-228, 2000.

[Lopes and Jorge 2000] A.A. Lopes and A. Jorge. Integrating rules and cases in learning via case explanation and paradigm shift. *Advances in Artificial Intelligence (Proc. IBERAMIA-SBIA 2000), LNAI 1952,* pp. 33-42. Springer, 2000.

[Michalski 1983] R. W. Michalski. A theory and methodology of inductive learning. *Artificial Intelligence 20, 1983, 111-161.*

[Michie et al. 1994] D. Michie, D.J. Spiegelhalter, C.C. Taylor. *Machine Learning, Neural and Statistical Classification*. Ellis Horwood, 1994.

[Mims 2000] C. Mims. *The War Within Us: every man's guide to infection and immunity*.

[Mirtin & Ritter 2000] B. Mirkin and O. Ritter. A feature-based approach to discrimination and prediction of protein folding groups. In: S. Suhai (Ed.) *Genomics and Proteomics: functional and computational aspects*, 157-177. Kluwer/Plenum, 2000.

[Mitchell 1990] T. Mitchell. The need for biases in learning generalizations. *Rutgers Technical Report,* 1980. Also published in: J.W. Shavlik and T.G. Dietterich (Eds.) *Readings in Machine Learning*, 184-191. Morgan Kaufmann, 1990.

[Mitchell 1997] T. Mitchell. *Machine Learning*. McGraw-Hill, 1997.

[Neal 2003] M. Neal. An Artificial Immune Network for the Analysis of Time Varying Data. In: *Artificial Immune Systems: Proc. Second Int. Conf. (ICARIS-2003), LNCS 2787,* pp. 168-180. Springer, 2003.

[Perelson 1989] A. Perelson. Immune Network Theory. *Immunological Review* 110:5-36, 1989.

[Quinlan 1993] J.R. Quinlan. Combining instance-based and model-based learning. Proc. 10th Int. Conf. on Machine Learning (ML-93), pp. 236-243. Morgan Kaufmann, 1993.

[Rao et al. 1995] R.B. Rao, D. Gordon and W. Spears. For every generalization action, is there really an equal and opposite reaction? Analysis of the conservation law for generalization performance. *Proc. 12th Int. Conf. on Machine Learning,* pp. 471-479. Morgan Kaufmann, 1995.

[Sahan et al. 2005] S. Sahan, K. Polat, H. Koaz, S. Gunes. The medical applications of attribute weighted artificial immune system (AWAIS): diagnosis of heart and diabetes diseases. In: *Artificial Immune Systems: Proc. Fourth Int. Conf. (ICARIS-2005). LNCS 3627,* pp. 456-468. Springer, 2005.

[Sarafijanovic & Le Boudec 2004] An artificial immune system for misbehavior detection in mobile ad-hoc networks with virtual thymus, clustering, danger signal and memory detectors. In: *Artificial Immune Systems: Proc. Third Int. Conf. (ICARIS-2004). LNCS 3239*, pp. 342-356. Springer, 2004.

[Schaffer 1994] C. Schaffer. A conservation law for generalization performance. *Proc. 11th Int. Conf. on Machine Learning*, pp. 259-265. Morgan Kaufmann, 1994.

[Secker et al 2003] A. Secker, A. Freitas and J. Timmis. AISEC: An Artificial Immune System for Email Classification. *Proceedings of the Congress on Evolutionary Computation (CEC-2003)*, pp. 131-139. IEEE, 2003.

[Sompayrac 2003] L. Sompayrac. *How the Immune System Works*. 2nd Ed. Blackwell Publishing, 2003.

[Stanfill & Waltz 1986] G. Stanfill and D. Waltz. Towards memory-based reasoning. *Communications of the ACM, 29(12)*, pp. 1213-1228, Dec. 1986.

[Stepney et al 2005] S. Stepney, R. Smith, J. Timmis, A. Tyrrell, M. Neal and A. Hone. Conceptual Frameworks for Artificial Immune Systems. *International Journal of Unconventional Computing.* 1(3):315-338, 2005.

[Stibor et al. 2005] T. Stibor, J. Timmis and C. Eckert. A comparative study of real-valued negative selection to statistical anomaly detection techniques. In: *Artificial Immune Systems: Proc. Fourth Int. Conf. (ICARIS-2005). LNCS 3627*, pp. 262-275. Springer, 2005.

[Tarakanov et al. 2003] A. Tarakanov, V. Skormin and S. Sokolova. 2003. *Immunocomputing: Principles and Practice*. Springer, 2003.

[Tarakanov & Tarakanov 2004] A. O. Tarakanov, Y. A. Tarakanov. A comparison of immune and neural computing for two real-life tasks of pattern recognition. *LNCS 3239*, pp. 236-249. Springer, 2004.

[Tarakanov et al. 2004] A.O. Tarakanov, S.V. Kvachev, A.V. Sukhorukov. A formal immune network and its implementation for on-line intrusion detection. *LNCS 3685*, pp. 394-405. Springer, 2004.

[Tan et al. 2006] P.-N. Tan, M. Steinbach and V. Kumar. *Introduction to Data Mining*. Addison-Wesley, 2006.

[Taylor & Corne 2003] D.W. Taylor and D.W. Corne. An investigation of the negative selection algorithm for fault detection in refrigeration systems. *Proceedings of the 2nd International Conference on Artificial Immune Systems (ICARIS-2003), LNCS 2787*, pages 34-45. Springer.

[Timmis et al. 1999] J. Timmis, M. Neal and J. Hunt. An artificial immune system for data analysis. *Proc. Int. Workshop on Intelligent Processing in Cells and Tissues (IPCAT).* Indianopolis, USA, 1999.

[Timmis and Neal 2001] J. Timmis and M. Neal. A Resource Limited Artificial Immune System. *Knowledge Based Systems 3/4*:121-130, 2001.

[Timmis and Knight 2002]. J. Timmis and T. Knight. Artificial immune systems: Using the immune system as inspiration for data mining. In: H. Abbas, A. Ruhul, A. Sarker and S. Newton (Eds.) *Data Mining: A Heuristic Approach.* pp. 209-230. Idea Group, 2002.

[Timmis et al 2004] J. Timmis, T. Knight, L.N. De Castro. and E. Hart. An Overview of Artificial Immune Systems. In: R Paton, H Bolouri, M Holcombe, J H Parish, and R Tateson (Eds.) *Computation in Cells and Tissues: Perspectives and Tools for Thought*, pp. 51-86. Springer, 2004.

[Ting 1994] K. M. Ting, The Problem of Small Disjuncts: its remedy in Decision Trees, *Proc. 10th Canadian Conference on AI,* pp. 91-97. 1994.

[Watkins 2001] A. Watkins. *A resource limited artificial immune classifier*. Master's thesis, Mississippi State University, MS. USA., December 2001.

[Watkins and Boggess 2002a] A. Watkins and L. Boggess. A resource limited artificial immune classifier. *Proc. Congress on Evolutionary Computation (CEC-2002)*. IEEE Press, 2002.

[Watkins and Boggess 2002b] A. Watkins and L. Boggess. A new classifier based on resource limited artificial immune systems. *Proc. Congress on Evolutionary Computation (CEC-2002)*, pp. 1546-1551. IEEE Press, 2002.

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

[Weiss & Indurkhya 1998] S.W. Weiss and N. Indurkhya. *Predictive Data Mining*. Morgan Kaufmann, 1998.

[Wettschereck et al. 1997] D. Wettschereck, D.W. Aha and T. Mohri. A review and empirical evaluation of feature weighting methods for a class of lazy learning algorithms. *Artificial Intelligence Review, 11 (1-5)*, pp. 273-314, Feb. 1997.

[White & Garret 2003] J.A. White and S.M. Garrett. Improved pattern recognition with artificial clonal selection? In: J. Timmis, P. Bentley, E. Hart (Eds.) *Artificial Immune Systems (Proc. ICARIS-2003), LNCS 2787,* 181-193. Springer, 2003.

[Witten & Frank 2005] I. H. Witten and E. Frank. *Data Mining: Pratical Machine Learning Tools and Techniques with Java Implementation*. 2nd Ed. San Mateo: Morgan Kaufmann, 2005.