Artificial Immune Systems for Diagnostic Classification Problems

Jeremy Mange[1], David Daniszewski[2], and Andrew Dunn[2]

[1] *Western Michigan University, Kalamazoo, MI, 49008*
jbmange@cs.wmich.edu

[2] *US Army RDECOM-TARDEC, Warren, MI, 48397*
david.g.daniszewski.civ@mail.mil
andrew.g.dunn.civ@mail.mil

ABSTRACT

Artificial Immune Systems (AISs), a class of artificial intelligence algorithms, have been an area of growing research and development in recent years. AISs, along with other well-known algorithms such as neural nets or particle swarm optimization, are biologically inspired, with AISs in particular designed to exhibit many of the behaviors of biological immune systems. In this paper, we explore the application of AISs to classification problems, particularly in the context of diagnostics, where the goal is generally to classify data into "nominal" or "error" classes. In particular, we present a formal definition of feature space as a multi-dimensional space constructed by a set of real-valued functions, define the process of feature selection, and explain and demonstrate its importance. We provide an overview of an AIS-based program developed for the International Diagnostic Competition, with particular focus on feature selection and AIS detector generation. Finally, we present experimental results, conclusions, and areas for future research.

1 BACKGROUND OF AIS

Artificial Immune Systems (AISs) form a class of artificial intelligence algorithms designed to mimic certain behaviors of biological immune systems. These algorithms exhibit many of the important features of the natural immune systems which inspire them, including adaptation, automated learning, and memory. Generally, AISs are used for variants of classification problems, in which data needs to be separated into one of two or more classes, often corresponding intuitively to "self" and "non-self" (Forrest *et. al.*, 1994), terms used in the context of biological immune systems to differentiate between healthy cells and the potential dangerous cells that trigger an immune system response. This basic immunological function has been applied in a wide number of contexts, including pattern recognition, anomaly detection, optimization, control problems, and many others. A number of journal papers, articles, and books describe the general function of AISs in more detail (Hofmeyer and Forrest, 2000; De Castro and Timmis, 2002) .

1.1 Feature Space

A representation of the problem space within which AISs operate is often referred to as "feature space" or "shape space". More formally, given a real value x, a "feature" of that data point is any real-valued function $f(x): \Re \rightarrow \Re$. If upper and lower limits for the domain of the function can be safely estimated, the function is commonly scaled to produce $f'(x): \Re \rightarrow [0,1]$. The feature space, then, is a set of n features $\{f_1, f_2, ..., f_n\}$, which collectively form an n-dimensional space in which to locate the data point. Given this feature space, the value x can be represented as a data point by the vector $\langle f_1(x), f_2(x), ..., f_n(x) \rangle$.

The general function, then, of most AISs is to establish a series of "detectors", with a specified volume within the feature space that does not include any of the "self", or healthy data points. Several methods for obtaining these detectors have been proposed and implemented; the method selected will be discussed in greater detail below. During the operation of the system, if any new data point falls within the region specified by one of these detectors, it is considered a "non-self" data point, and appropriate action is taken to address the potential fault.

To illustrate this idea visually, we will show a tightly packed training data set with a very simple division between the "self" nominal data (corresponding to the circles in the graph) and "non-self" error data (corresponding to the Xs in the graph), in figure 1. We plot the data using two arbitrary features, f_1 and f_2, which together will define the two-dimensional feature space for this example.

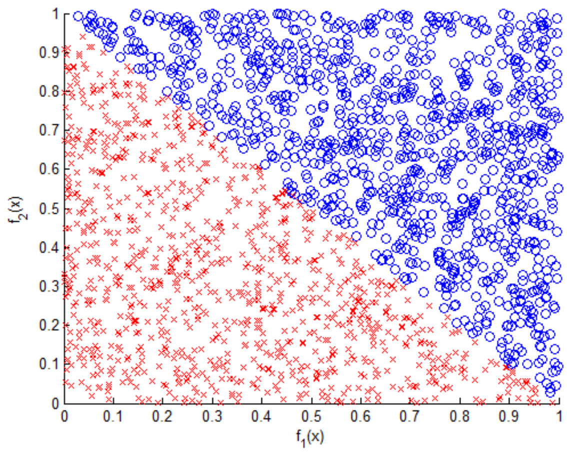

Figure 1: Simple Data Set

In this example, any data point x where $f_1(x)+f_2(x)>1$ is part of the "self", and any other data point is part of the "non-self". The goal, then, is to generate a series of detectors with a volume in the two-dimensional feature space that covers as much of the non-self as possible. For a two-dimensional feature space, these detectors are generally represented as circles (more discussion of detector shape in subsequent sections). Any new test data point which falls within one of these detectors circles would be treated as a non-self data point, and an appropriate error response would occur in the system.

Figure 2 shows a plot of the same self data with a set of randomly generated detectors using the negative selection algorithm (discussed further in section 1.3). Note that this does not include any of the common techniques for filtering out detectors completely contained within others, or optimizing detector generation and efficiency.

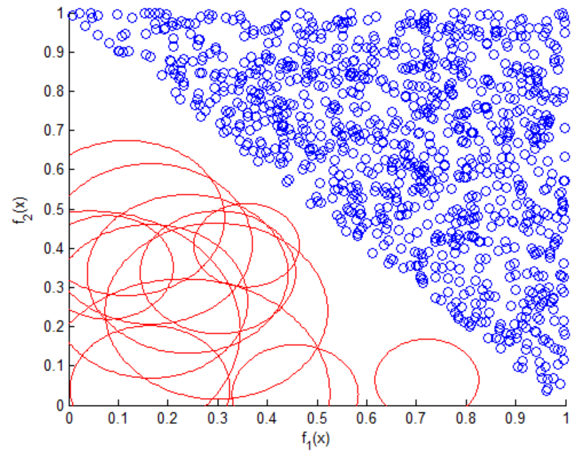

Figure 2: Simple Data Set with Random Detectors

This illustrates a simplified version of the basic process involved with detector generation for an AIS. The detectors shown cover some portion of the non-self feature space and will "react" with any test data point which lies in that portion. Since each test data point must be checked for its distance from the center of each detector (often referred to as the data point's "affinity" to the detector), the processing time increases at least linearly with the number of detectors. Depending on the application, the number of needed detectors can be estimated based on the desired trade-off between processing speed and detection accuracy.

1.2 Feature Selection

Since a feature can be any real-valued function, for any given problem, there are infinitely many combinations of features available to be chosen to comprise the feature space for an AIS. The question of how to appropriately select features for a problem is one that seems to have received relatively little attention in literature, and one that is of particular interest to the authors. Feature selection can greatly affect not only the performance, but also the results of an AIS, which we shall illustrate using a simplified example.

Suppose we are given a problem for which we design an AIS, and we consider three arbitrary features for representing the feature space, $f_1:\Re\rightarrow[0,1]$, $f_2:\Re\rightarrow[0,1]$, $f_3:\Re\rightarrow[0,1]$. Furthermore, suppose the actual data for our classification problem is such that each value x is part of the "self" if and only if $f_1(x)^2-f_2(x)<-0.2$. A plot of the data using the first two features, then, might look like figure 3:

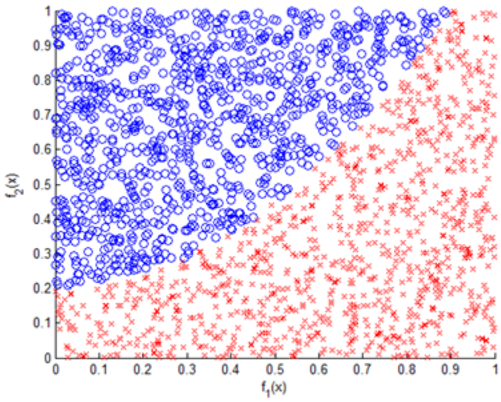

Figure 3: Features 1 and 2

Note that because f_1 and f_2 are precisely the two features which are used to define the separation of the data into two classes, there is a very clean difference between the spaces representing each on the plot. Even with ideal feature selection, it is unlikely that the separation in a real-world data set would be as clear.

However, since by using the standard representation for AIS detectors, each detector is a circle in two-dimensional space (or a sphere or hyper-sphere in higher-dimensional space), it is easy to see visually that it will take many detectors to accurately cover the non-self portions of the data, even in two-dimensional space. So then, if we chose to represent the feature space as two-dimensional, consisting of f_1 and f_2, it would be possible to form a detector set to accurately classify the data, but it might take many detectors.

If instead we chose to represent the feature space using f_1 and f_3, the situation is much worse. Figure 4 shows a plot of the same data in two-dimensional space using those two features:

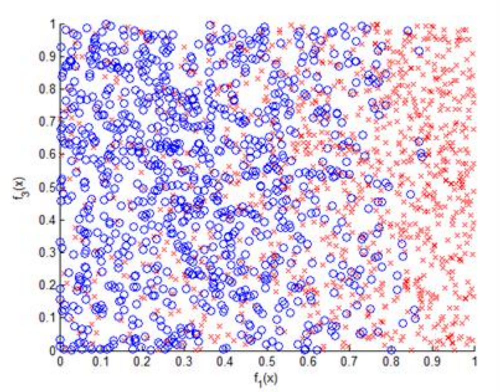

Figure 4: Features 1 and 3

Since the actual classification of the data does not require f_3, its inclusion together with the exclusion of the important feature f_2 makes the generation of accurate detectors impossible.

On the other hand, if we define a new feature $f_4(x) = f_1(x)^2 - f_2(x)$ (appropriately scaled), we could accurately classify all the data using only a one-dimensional feature space and a single detector. It is difficult to visually demonstrate a one-dimensional space, but even if we include f_1 and plot it against the newly defined feature f_4, the simplicity of detection is plain, as shown in figure 5:

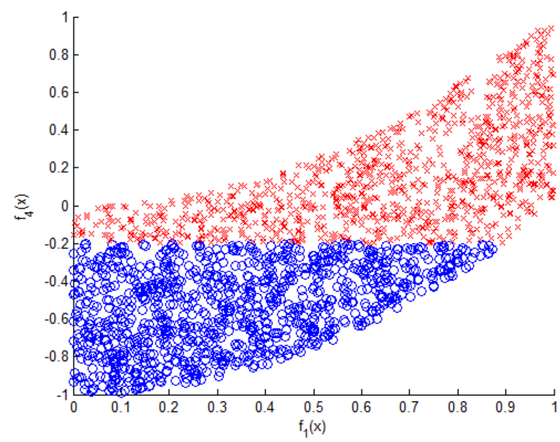

Figure 5: Features 1 and 4

Obviously, this is an ideal situation in which we know ahead of time how the data is classified, but it illustrates the point that good feature selection not only can make the difference between a difficult task and an easy one, but can in fact make the difference between an impossible task and a possible one.

1.3 Negative Selection Algorithm

Several methods of obtaining a detector set have been proposed and implemented. The one most commonly used for problems similar to data classification for diagnostics is the Negative Selection algorithm. In its most general form, the Negative Selection algorithm estimates the number of detectors needed to cover the "non-self" portions of the feature space, then randomly generates detectors, discarding those that include any "self" data points, until that number is reached.

Although a detector can be any shape with volume within the feature space, for the sake of compact representation they are generally represented as a hyper-sphere, thus necessitating the storage of only a center and a radius. Using this representation, several refinements to the basic Negative Selection algorithm have been proposed, including moving the center for detectors which include any "self" data points, or increasing or decreasing the radius of a detector under certain conditions (D'haeseleer *et. al.*, 1996). The method we chose is outlined in the "Our Implementation" section.

2 AIS FOR DIAGNOSTICS

The basic problem for diagnostics is one of classification -- separating nominal data from error data in order to determine when an error occurs. For this reason, AISs have a natural application to diagnostic systems, and indeed some research has already been done along these lines (for example, Dasgupta *et. al.*, 2004).

A typical approach to a diagnostic task using an AIS would be to first gather training data, either including both nominal and failure data, or simply nominal data. Then, using this training data, a set of detectors would be generated. Finally, the system would be run using new test data, with the detector set alerting the system of the possibility of a fault if any test data point fell within the volume of the feature space that they occupy.

2.1 DX Competition

The International Diagnostic Competition (Feldman *et. al.*, 2010) is an annual competition designed to provide a method for testing, evaluating, and comparing approaches to automated diagnostics. As such, it provided an ideal platform for implementing and testing an AIS for the purposes of diagnostics, using real-world electrical system data collected at the NASA Ames Research Center. The authors of this paper implemented such a system and submitted it for the Third International Diagnostic Competition (DXC 2011) under the ADAPT-Lite track under the name "AntigenDX".

3 OUR IMPLEMENTATION

For the competition, we implemented a basic AIS using a Negative Selection algorithm to generate detectors. The competition organizers provided training data in the form of a sequence of "scenarios", some of which were nominal operations of the modeled electrical system, and some of which included known failures. Since in many applications (including the ground vehicle context by which the authors are primarily motivated) obtaining failure data is not feasible, we chose to use only the nominal training data for the generation of the detectors, and then used the failure data for testing.

The scenario files for the competition include time series data for a number of sensors within the electrical system. We used a separate feature space for each sensor in order to detect errors separately on a per-sensor basis, and then aggregated any detected faults to provide an overall diagnosis.

3.1 Feature Selection

As stated previously, of major interest to the authors is feature selection within AISs. One of our primary goals with this system, therefore, was to select a small number of features which would nevertheless provide a sufficiently specific feature space to allow for accurate classification.

For every sensor, we were able to obtain extremely accurate (>95% accuracy) results using only two features, that is, a two-dimensional feature space. For all of the sensors except one, those features were:

$$f_1(x) = (m - x)^2 \tag{1}$$

where *m* is the mean of the sensor data for the first 20 seconds of the scenario

$$f_2(x) = c \tag{2}$$

where *c* is the number of consecutive data points with the value *x*

For one sensor (sensor "ST516", see Feldman *et. al.*, 2010), f_2 above was used in conjunction with the most basic feature, $f_3(x) = x$. This was because that particular sensor had a fairly wide variation for its values of $f_1(x)$ defined above, but only a small set of actual data values in nominal operation. Again, this illustrates that within an overarching system, different features can be selected for separate data sets (sensors, in this case) based on the efficacy of each feature in partitioning the data set into separable spaces based on the classification classes.

All of these features were obtained using extensive testing, mostly through visual inspection. After examining the data through a variety of tools, various functions were proposed, and then their feasibility as features was tested using the test data provided. The "Extensions and Further Work" section contains more information.

3.2 Detector Generation

After defining the two-dimensional feature space for each sensor, we generated detectors using the Negative

Selection algorithm outlined above. Our basic approach was as follows:

- randomly generate detector center and radius
- if detector includes any test data point, discard
- if not, increase radius until it touches a test data point
    - decrease radius by a factor α (to provide a "cushion" around test data)

This is described using the standard hyper-sphere detector shape for detectors. In actuality, we implemented the system using rectangular detectors. However, this does not differ conceptually, but only in implementation details. The "Extensions and Further Work" section provides further notes on different shapes for detectors within the feature space.

3.3 Integration with DXC Framework

After implementing the AIS components of the diagnostic system, the major remaining challenge was aggregating error detection by individual sensors into an accurate diagnosis of faults within the electrical system as a whole. This was done through a rule-based system derived from the provided documentation on the electrical system components and interconnections.

When the AIS subsystem first detects any error data point, a countdown is begun for a window of time before a diagnosis is attempted. This is to help ensure that if multiple components have errors, sufficient time elapses to allow the AIS subsystem to detect as many as possible.

As an intermediate step, the fault type for each faulty sensor is estimated ("abrupt", "intermittent", or "drift" faults, as per Feldman *et. al.*, 2010), using elementary statistics from the time series data before and after the time of the fault detection. At that point, the set of components with detected errors is submitted to the rule-based diagnosis subsystem. This subsystem consists of a series of approximately 25 if-else rules which, based on the electrical system description, work from the power source of the system outwards to attempt fault diagnosis. Finally, based on the electrical properties of the components in question and the type of fault diagnosed, the significant parameters of the fault are estimated.

All of these components were integrated with the Java framework for the competition, in order to receive time series data and report diagnoses for test data.

3.4 Testing / Results

As mentioned previously, the AIS detectors were generated using only the nominal data from the training data sets. Once the feature space was defined and the detectors in place, we tested the system against all of the training data, both nominal and error. Our final system achieved a classification accuracy of 99.67% on the provided training data, in terms of the AIS classification of data into the "nominal" or "error" classes; that is $c/(sn) = .9967$, where c is the total number of correctly classified sensors, s is the number of scenarios, and n is the number of sensors in each scenario. Given that we were able to use only two-dimensional feature space for each feature, we were very pleased with these results, and believe that they provide validation for the use of AISs for diagnostic problems of this type and justification for further research and development.

The ADAPT-Lite system is a fairly steady-state system, and many of the errors in the testing data were fairly easy to detect compared to many real-world problems, particularly in the area of ground vehicle diagnostics. Therefore, while we believe this application provides a baseline validation for use of AISs in diagnostic problems, further extensions would be necessary to apply these techniques to other problems. Some research has already been performed in this area, and a number of extensions have been proposed which apply directly to the problems under consideration.

For longer-running or more dynamic systems, instead of simply producing detectors based on training data, detectors can be continuously produced throughout the life of the system. These detectors can have memory mechanisms and mature and degenerate over time in order to deal with changing characteristics of both the "host" system and the "pathogen" anomalous data it might encounter (see Hofmeyr and Forrest, 1999).

The official results of the Third International Diagnostic Competition, which beyond diagnosis accuracy will contain several additional metrics including detection speed and resource usage, are not yet available as of the writing of this paper. However, based on the success of the approach in the classification of all training data, using detectors generated only from the nominal scenario subset of that data, we emphasize that the AIS approach already shows promise for similar classification problems.

4 EXTENSIONS AND FURTHER WORK

An immediately apparent area for further research is automating the process of feature selection. This paper has demonstrated the importance of feature selection within AISs and highlighted the power that appropriate feature selection can produce. We believe that certain machine learning techniques such as genetic approaches may have promise when applied at a meta-level to feature selection within AISs, and have begun some work to investigate this possibility further.

A potential limitation of most current approaches to AISs is the use of a hypersphere shape for all detectors within the system. Although this provides for a compact representation of the detector, it sacrifices some generality and limits the actual shape of the space that can be covered with a given number of detectors. We have begun some work investigating the possibility of linearly transforming a hyperspherical detector in order to facilitate far more general shapes. More general shapes would, in theory, allow for more arbitrary coverage of the "non-self" portions of the feature space for an AIS, but would also add a level of complexity to the process of generating useful detectors. We hope to further explore these trade-offs.

5 CONCLUSION

Artificial Immune Systems can be useful artificial intelligence tools, particularly for classification problems, including diagnostics. As part of the International Diagnostic Competition, we have developed an AIS which should further demonstrate the validity of the approach within the context of diagnostics. Since ground vehicle applications are of particular interest to the authors, we hope that this research will help to promote the development of AIS-based diagnostic algorithms and equipment for use with ground vehicle and other electrical and mechanical systems.

REFERENCES

(Daspugta *et. al.*, 2004) Dasgupta, D., K. KrishnaKumar, D. Wong, and M. Berry. "Negative Selection Algorithm for Aircraft Fault Detection." *Lecture Notes in Computer Science* 3239 (2004): 1-13.

(De Castro and Timmis, 2002) De Castro, Leandro N., and Jonathan Timmis. *Artificial Immune Systems: a New Computational Intelligence Approach*. London: Springer, 2002.

(D'haeseleer *et. al.*, 1996) D'haeseleer, Patrik, Stephanie Forrest, and Paul Helman. "An Immunological Approach to Change Detection: Algorithms, Analysis and Implications." *IEEE Symposium on Security and Privacy* (1996).

(Feldman *et. al.*, 2010) Feldman, Alexander, Tolga Kurtoglu, Sriram Narasimhan, Scott Poll, David Garcia, Johan De Kleer, Lukas Kuhn, and Arjan Van Gemund. "Empirical Evaluation of Diagnostic Algorithm Performance Using a Generic Framework." *International Journal of Prognostics and Health Managemen* (2010).

(Forrest *et. al.*, 1994) Forrest, Stephanie, Alan S. Perelson, Lawrence Allen, and Rajesh Cherukuri. "Self-Nonself Discrimination in a Computer." *IEEE Symposium on Research in Security and Privac* (1994).

(Hofmeyr *et. al.*, 2000) Hofmeyr, Steven A., and Stephanie Forrest. "Architecture for an Artificial Immune System." *Evolutionary Computation* 8.4 (2000): 443-73.

(Hofmeyr and Forrest, 1999) Hofmeyr, S. A. and Forrest, S., "Immunity by Design: An Artificial Immune System." *Proceedings of GECCO* (1999): 1289-1296.

(Ji and Dasgupta, 2004) Ji, Zhou, and Dipankar Dasgupta. "Real-Valued Negative Selection Algorithm with Variable-Sized Detectors." *Proceedings of GECCO* 3102 (2004): 287-98.

(Kephart, 1994) Kephart, Jeffrey O. "A Biologically Inspired Immune System for Computers." *Artificial Life IV* (1994).