RST-DCA: A Dendritic Cell Algorithm Based on Rough Set Theory

Zeineb Chelly and Zied Elouedi

LARODEC, University of Tunis, 2000 Le Bardo, Tunisia
zeinebchelly@yahoo.fr, zied.elouedi@gmx.fr

Abstract. The Dendritic Cell Algorithm (DCA) is an immune-inspired classification algorithm based on the behavior of dendritic cells. The DCA performance depends on its data pre-processing phase including feature selection and their categorization to specific signal types. For feature selection, DCA applies the principal component analysis (PCA). Nevertheless, PCA does not guarantee that the selected first principal components will be the most adequate for classification. Furthermore, the categorization of features to their specific signal types is based on the PCA attributes' ranking in terms on variability which does not make "sense". Thus, the aim of this paper is to develop a new DCA data pre-processing method based on Rough Set Theory (RST). In this newly-proposed hybrid DCA model, the selection and the categorization of attributes are based on the RST CORE and REDUCT concepts. Results show that using RST instead of PCA for the DCA data pre-processing phase yields much better performance in terms of classification accuracy.

Keywords: Artificial immune systems, Dendritic Cells, Rough Sets, Core, Reduct.

1 Introduction

Artificial Immune Systems (AIS) are a class of computationally intelligent systems inspired by the principles of the vertebrate immune system. As AIS is being developed significantly, novel algorithms termed "2nd Generation AISs" have been created. One such 2nd Generation AIS is the Dendritic Cell Algorithm (DCA) [5] which is based on the behavior of the natural "dendritic cells" (DCs). DCA has been successfully applied to various applications. In fact, its performance depends on its data pre-processing phase which is divided into two main steps: feature selection and signal categorization. More precisely, DCA uses the principal component analysis (PCA) to automatically select features and to categorize them to their specific signal types; as danger signals (DS), as safe signals (SS) or as pathogen-associated molecular patterns (PAMP)[6]. DCA combines these signals with location markers in the form of antigen to process his classification task. For signal selection, PCA transforms a finite number of possibly correlated vectors into a smaller number of uncorrelated vectors, termed "principal components" which reveals the internal structure of the given data with the

T. Huang et al. (Eds.): ICONIP 2012, Part III, LNCS 7665, pp. 480–487, 2012.

focus on data variance [6]. However, using PCA for feature selection presents a drawback as it is not necessarily true that the first selected components will be the adequate features to retain [7]. Thus, the choice of these components for the DCA can influence its classification task by producing unreliable results. As for feature categorization, DCA uses the generated PCA ordered list of standard deviation values to assign for each selected attribute its signal type (SS, DS or PAMP). However, this categorization process which is based on high and low values of the calculated standard deviations does not make "sense" as a coherent process which can influence negatively the DCA functioning. Thus, in this paper, we develop a novel AIS hybrid model based on a new automatic data pre-processing phase for the DCA. As DCA was hybridized with various techniques to improve its classification performance such as with fuzzy set theory [2], a fuzzy clustering technique [3] and a maintenance policy [4], in this paper, our new hybrid model named "RST-DCA" is grounded on the behavior of DCs within the framework of Rough Set Theory (RST). Our RST-DCA model uses the RST REDUCT and CORE concepts to select the right features to retain and to categorize them into their right signal types. This paper is structured as follows: Section 2 of this paper introduces the DCA. Section 3 presents the RST concepts. Section 4 details our hybrid RST-DCA AIS system. The experiments and the results are outlined in Section 5 and 6.

2 The Dendritic Cell Algorithm

The first DCA step is data pre-processing which includes feature selection and signal categorization. For signal selection, DCA applies the PCA that reduces data dimension, by accumulating the vectors that can be linearly represented by each other [6]. Once features are selected, PCA is applied to assign each attribute to its specific signal type. More precisely, DCA uses the PCA calculated standard deviations and selects the highest values. As both PAMP and SS are positive indicators of an anomalous and normal signal [5], one attribute is used to form both PAMP and SS. Thus, the attribute having the lowest standard deviation out of the selected attribute set is used to form both PAMP and SS. Using one attribute for these two signals requires a threshold level to be set: values greater than this can be classed as SS otherwise as PAMP [5]. As for the DS attribute assignment and since the DS is "less than certain to be anomalous", the combination of the rest of the selected attributes are chosen to represent it [5]. After calculating the values of SS, PAMP and DS [5], DCA adheres these signals and antigen to fix the context of each DC. DCA processes its input signals to decide whether the collected DC goes to the semi-mature context, implying that the antigen data is normal, or if the DC goes to the mature context, signifying an anomalous data item. The nature of the response is determined by measuring the number of fully mature DCs and is represented by the Mature Context Antigen Value (MCAV). *MCAV* is used to assess the degree of anomaly of a given antigen. By applying thresholds at various levels, analysis can be performed to assess the anomaly detection capabilities of the algorithm. Those antigens whose

$MCAV$ are greater than the anomalous threshold are classified as anomalous else as normal. More DCA details and its pseudocode can be found in [5].

3 Rough Set Theory

In RST [8], an *information table* is defined as a tuple $T = (U, A)$ where U and A are two finite, non-empty sets, U the *universe* of primitive objects and A the set of attributes. A may be partitioned into C and D, called *condition* and *decision* attributes, respectively. Let $P \subset A$ be a subset of attributes. The indiscernibility relation, $IND(P)$, is an equivalence relation defined as: $IND(P) = \{(x, y) \in U \times U : \forall a \in P, a(x) = a(y)\}$, where $a(x)$ denotes the value of feature a of object x. The family of all equivalence classes of $IND(P)$ is denoted by $U/IND(P)$. Equivalence classes $U/IND(C)$ and $U/IND(D)$ are respectively called *condition* and *decision* classes. For any concept $X \subseteq U$ and attribute subset $R \subseteq A$, X could be approximated by the R-*lower* and R-*upper* approximations using the knowledge of R. The X lower approximation is the set of objects U that are surely in X, defined as: $\underline{R}(X) = \bigcup\{E \in U/IND(R) : E \subseteq X\}$. The X upper approximation is the set of U objects that are possibly in X, defined as: $\overline{R}(X) = \bigcup\{E \in U/IND(R) : E \cap X \neq \emptyset\}$. The boundary region is defined as: $BND_R(X) = \overline{R}(X) - \underline{R}(X)$. If $BND_R(X)$ is empty, $\overline{R}(X) = \underline{R}(X)$, X is said to be R-*definable*. Otherwise X is a rough set with respect to R. The positive region of $U/IND(D)$ with respect to C is denoted by $POS_c(D)$ where: $POS_c(D) = \bigcup \overline{R}(X)$. $POS_c(D)$ is a set of objects of U that can be classified with certainty to classes $U/IND(D)$ employing attributes of C. For feature selection, RST defines two main concepts; the CORE and the REDUCT. The CORE is equivalent to the set of strong relevant features which are *indispensable* attributes in the sense that they cannot be removed without loss of prediction accuracy of the original database. The REDUCT is a combination of all strong relevant features and some weak relevant features that can sometimes contribute to prediction accuracy. These concepts provide a good foundation upon which we can define our basics for defining the importance of each attribute. In RST, a subset $R \subseteq C$ is said to be a D-*reduct* of C if $POS_R(D) = POS_C(D)$ and there is no $R' \subset R$ such that $POS_{R'}(D) = POS_C(D)$. In other words, the REDUCT is the minimal set of attributes preserving the positive region. There may exist many reducts (a family of reducts), $RED_D^F(C)$, in T. The CORE is the set of attributes that are contained by all reducts, defined as: $CORE_D(C) = \bigcap RED_D(C)$ where $RED_D(C)$ is the D-reduct of C. In other words, the CORE is the set of attributes that cannot be removed without changing the positive region. This means that all attributes present in the CORE are indispensable.

4 RST-DCA: The Solution Approach

4.1 RST-DCA Feature Selection Process

Our learning problem is to select high discriminating features for antigen classification from the original input data set which corresponds to the antigen information database. We may formalize this problem as an information table, where universe $U = \{x_1, x_2, \ldots, x_N\}$ is a set of antigen identifiers, the conditional attribute set $C = \{c_1, c_2, \ldots, c_N\}$ contains each feature of the information table to select and the decision attribute D of our learning problem corresponds to the class label of each sample. As DCA is applied to binary classification problems, the input database has a single binary decision attribute. Hence, the decision attribute D, which corresponds to the class label, has binary values d: either the antigen is collected under safe circumstances reflecting a normal behavior (classified as normal) or the antigen is collected under dangerous circumstances reflecting an anomalous behavior (classified as anomalous). The condition attribute feature D is defined as follows: $D = \{normal, anomalous\}$. For that, RST-DCA computes, first of all, the positive region for the whole attribute set C for both label classes of D: $POS_C(\{d\})$. Based on the RST computations (seen previously in Section 3), RST-DCA computes the positive region of each feature c and the positive region of all the composed features $C - \{c\}$ (when discarding each time one feature c from C) defined respectively as $POS_c(\{d\})$ and $POS_{C-\{c\}}(\{d\})$, until finding the minimal subset of attributes R from C that preserves the positive region as the whole attribute set C does. In fact, RST-DCA removes in each computation level the unnecessary features that may affect negatively the accuracy of the RST-DCA. The result of these computations is either one reduct $R = RED_D(C)$ or a family of reducts $RED^F_D(C)$. Any reduct of $RED^F_D(C)$ can be used to replace the original antigen information table. Consequently, if the RST-DCA generates only one reduct $R = RED_D(C)$ then for the feature selection process, RST-DCA chooses this specific R which represents the most informative features that preserve nearly the same classification power of the original data set. If the RST-DCA generates a family of reducts $RED^F_D(C)$ then RST-DCA chooses randomly one reduct R among $RED^F_D(C)$ to represent the original input antigen information table. This random choice is argued by the same priority of all the reducts in $RED^F_D(C)$. In other words, any reduct R of the reducts $RED^F_D(C)$ can be used to replace the original information table. These attributes which constitute the reduct will describe all concepts in the original training data set. By using the REDUCT, our method can guarantee that the selected attributes will be the most relevant for its classification task.

4.2 RST-DCA Feature Categorization Process

RST-DCA has to assign, now, for each selected attribute, produced by the previous step, its specific signal type; either as PAMP, as DS or SS. As previously stated, both PAMP and SS have a certain final context (either an anomalous or a normal behavior) while the DS cannot specify exactly the final context to assign to the collected antigen as the DS may or may not indicate an anomalous situation. This problem can be formulated as follows: Both PAMP and SS are more informative than DS which means that both of these signals can be seen as indispensable attributes. To define this level of importance, our method uses the CORE RST concept. As for DS, it is less informative than PAMP and SS. Therefore, RST-DCA uses the rest of the REDUCT attributes (discarding the attributes of the CORE chosen to represent both SS and PAMP) to represent the DS. As stated in the previous step, our method may either produce only one reduct R or a family of reducts $RED_D^F(C)$. The process of signal categorization for both cases are described in what follows: In case where our RST-DCA generates only one reduct; it means that $CORE_D(C) = RED_D(C)$. In other words, all the features of the reduct are indispensable. In this case, RST-DCA selects randomly one attribute c from $CORE_D(C)$ and assigns it to both PAMP and SS as they are the most informative signals. Using one attribute for these two signals requires a threshold level to be set: values greater than this can be classed as SS, otherwise as a PAMP signal. The rest of the attributes $CORE_D(C) - \{c\}$ are combined and the resulting value is assigned to the DS as it is less than certain to be anomalous. In case where our RST-DCA produces a family of reducts $RED_D^F(C)$, the RST-DCA presents both concepts: the core $CORE_D(C)$ and the reduct $RED_D^F(C)$. Let us remind that $CORE_D(C) = \bigcap RED_D(C)$; which means that on one hand we have the minimal set of attributes preserving the positive region (reducts) and on the other hand we have the set of attributes that are contained in all reducts (core) which cannot be removed without changing the positive region. This means that all the attributes present in the CORE are indispensable. For signal categorization, PAMP and SS are assigned, randomly, one attribute c among the features in $CORE_D(C)$. As for the DS signal assignment, RST-DCA chooses, randomly, a reduct $RED_D(C)$ among $RED_D^F(C)$. Then, RST-DCA combines all the $RED_D(C)$ features except that c attribute already chosen and assigns the resulting value to the DS. Once signal categorization is achieved, RST-DCA processes its next steps as the DCA does [5].

5 Experimental Setup

To test the validity of our RST-DCA hybrid model, our experiments are performed using binary databases from [1] described in Table 1.

For data pre-processing, DCA and RST-DCA uses PCA and RST, respectively. Each data item is mapped as an antigen, with the value of the antigen equal to the data ID of the item. To perform anomaly detection, a threshold which is automatically generated from the data is applied to the MCAVs. The MCAV threshold is derived from the proportion of anomalous data instances of

Table 1. Description of Databases

Database	Ref	♯ Instances	♯ Attributes
Spambase	SP	4601	58
SPECTF Heart	SPECTF	267	45
Cylinder Bands	CylB	540	40
Chess	Ch	3196	37
Ionosphere	IONO	351	35
Mushroom	Mash	8124	23
Congressional Voting Records	CVT	435	17
Tic-Tac-Toe Endgame	TicTac	958	10

the whole data set. Items below the threshold are classified as class 1 and above as class 2. The resulting classified antigens are compared to the labels given in the original data sets. The results presented are based on mean MCAV values generated across 10 runs. We evaluate the performance of RST-DCA in terms of number of extracted features, sensitivity, specificity and accuracy which are defined as: $Sensitivity = TP/(TP + FN); Specificity = TN/(TN + FP); Accuracy = (TP + TN)/(TP + TN + FN + FP)$; where TP, FP, TN, and FN refer respectively to: true positive, false positive, true negative and false negative. We will also compare the classification performance of our RST-DCA method to well known classifiers which are the Support Vector Machine (SVM), Artificial Neural Network (ANN) and to the Decision Tree (DT).

6 Results and Discussion

In this Section, we show that using RST instead of PCA is much convenient for the DCA data pre-processing phase as it improves its classification performance which is confirmed by the results given in Table 2. Let us remind that for signal selection, DCA applies PCA where it selects the highest standard deviation values. As the highest values have to be selected, this needs either to keep only the eigenvalues larger than 1 [7] or involving the user to decide which features to keep for the algorithm. However, the fact of using eigenvalues can either lead to overestimate the number of factors to keep or to underestimate it leading to ignore important information. In addition, involving users to determine a priori the number of attributes to retain may result to preserve more or less features than necessary. In this Section, we will show that these problems are solved by our RST-DCA.

From Table 2, it is clearly seen that the number of features selected by our RST-DCA is less than the one generated by DCA when applying PCA (PCA-DCA). This can be explained by the appropriate use of RST for feature selection. In fact, RST-DCA keeps only the most informative features which constitute the REDUCT. For instance, by applying our RST-DCA method to the CylB data set, the number of selected features is only 7 attributes. However, when applying

Table 2. DCA and RST-DCA Comparison Results

Database	Sensitivity (%)		Specificity (%)		Accuracy (%)		♯ Attributes	
	DCA		DCA		DCA		DCA	
	PCA	RST	PCA	RST	PCA	RST	PCA	RST
SP	86.76	94.53	87.58	94.47	87.26	94.5	14	8
SPECTF	72.16	84.43	67.27	74.54	71.16	82.4	11	4
CylB	91.50	96.50	92.94	96.79	92.38	96.67	16	7
Ch	94.06	97.84	93.64	98.23	93.86	98.02	14	11
IONO	93.65	95.23	94.22	96.88	94.58	96.29	24	19
Mash	99.41	99.82	99.28	99.73	99.34	99.77	7	6
CVT	91.07	95.83	92.13	97	91.72	96.55	14	8
TicTac	91.37	93.45	89.15	93.67	90.6	93.52	7	6

the PCA-DCA to the same database (CylB), the number of the retained features is 16. We can notice that PCA preserves additional features which are the result of the PCA overestimation of the number of factors to retain. This overestimation affects the DCA classification task by producing unreliable results. On the other hand, RST-DCA based on the REDUCT concept, selects the minimal set of features from the original database and can guarantee that the reduct attributes will be the most relevant for its classification task. In fact, by reducing more the number of features while preserving the classification power of the original data set, our RST-DCA has the advantages to decrease the cost of acquiring data and to make the classification model easier to understand unlike when applying the PCA. In addition, RST-DCA has sufficient advantages over the PCA-DCA, as it does not require any additional information about data a priori such as thresholds or expert knowledge on a particular domain. Thus, RST-DCA results will not be influenced by any external information. As for the classification accuracy, from Table 2, we can easily remark that the RST-DCA accuracy is notably better than the one given by the PCA-DCA. For example, when applying the RST-DCA to the CylB database, the RST-DCA accuracy is set to 96.67%. Nevertheless, when applying the PCA-DCA to the same database, the accuracy is 92.38%. Same remark is noticed for both the sensitivity and the specificity criteria. These encouraging RST-DCA results are explained by the appropriate set of features selected and their categorization to their right and specific signal types. As stated previously, the classification results of the DCA depends on its data pre-processing phase which is crucial to obtain reliable results. RST-DCA uses the REDUCT RST fundamental concept to select only the essential part of the original database. This pertinent set of minimal features can guarantee a solid base for the signal categorization step. The RST-DCA good classification results are also explained by the appropriate categorization of each selected signal to its right signal type by using both the REDUCT and the CORE concepts. As for DCA, by applying the PCA, it produces less accuracy in comparison to our RST-DCA method which is explained by the inappropriate use of the PCA for data pre-processing. In fact, the first components selected are not necessarily the right set of features to retain since this set still contains extra features that do not add anything new to the target concept while increasing the cost of acquiring data. The set may also contain misleading features which have a negative effect on classification accuracy. Furthermore, the DCA categorization step does not make "sense" as a coherent categorization procedure.

The performance of our RST-DCA is, also, compared to SVM, ANN and to DT in terms of the average of accuracies on the 8 data sets. The parameters of SVM, ANN and DT are set to the most adequate parameters to these algorithms using the Weka software. Figure 1 shows that PCA-DCA has nearly the same classification performance as SVM and ANN and a better one than DT. It also shows that our RST-DCA outperforms all the mentioned classifiers including the PCA-DCA in terms of overall accuracy. These encouraging RST-DCA results are explained by the appropriate application of RST to the DCA data pre-processing phase making the DCA a better classifier by generating pertinent and more reliable results.

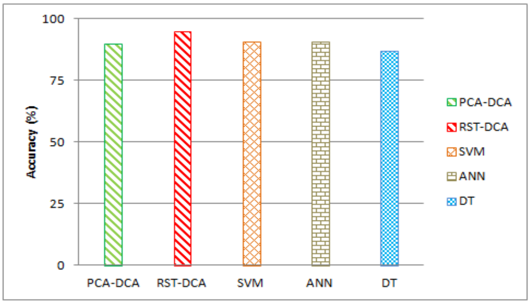

Fig. 1. Comparison of Classifiers' Average Accuracies on the 8 Binary Datasets

7 Conclusion and Further Works

In this paper, we have introduced a new hybrid computational biological model for the DCA based on RST. Our model aims to select the convenient set of features from the initial database and to perform their signal categorization using the REDUCT and the CORE RST concepts. The experimentation results show that our RST-DCA is capable of performing better its classification task than DCA and other classifiers. Future works will include the use of fuzzy rough set theory for the DCA and the application of RST-DCA to real world problems.

References

1. Asuncion, A., Newman, D.J.: UCI Machine Learning Repository (2007)
2. Chelly, Z., Elouedi, Z.: FDCM: A Fuzzy Dendritic Cell Method. In: Hart, E., McEwan, C., Timmis, J., Hone, A. (eds.) ICARIS 2010. LNCS, vol. 6209, pp. 102–115. Springer, Heidelberg (2010)
3. Chelly, Z., Elouedi, Z.: Further Exploration of the Fuzzy Dendritic Cell Method. In: Liò, P., Nicosia, G., Stibor, T. (eds.) ICARIS 2011. LNCS, vol. 6825, pp. 419–432. Springer, Heidelberg (2011)
4. Chelly, Z., Smiti, A., Elouedi, Z.: COID-FDCM: The Fuzzy Maintained Dendritic Cell Classification Method. In: Rutkowski, L., Korytkowski, M., Scherer, R., Tadeusiewicz, R., Zadeh, L.A., Zurada, J.M. (eds.) ICAISC 2012, Part II. LNCS, vol. 7268, pp. 233–241. Springer, Heidelberg (2012)
5. Greensmith, J., Aickelin, U., Cayzer, S.: Introducing Dendritic Cells as a Novel Immune-Inspired Algorithm for Anomaly Detection. In: Jacob, C., Pilat, M.L., Bentley, P.J., Timmis, J.I. (eds.) ICARIS 2005. LNCS, vol. 3627, pp. 153–167. Springer, Heidelberg (2005)
6. Gu, F., Greensmith, J., Oates, R., Aickelin, U.: Pca 4 dca: The application of principal component analysis to the dendritic cell algorithm. CoRR (2010)
7. Kaiser, H.: A note on guttmans lower bound for the number of common factors. British Journal of Mathematical and Statistical Psychology 14, 1–2 (1961)
8. Pawlak, Z., Rough, S.: International Journal of Computer and Information Science 11, 341–356 (1982)