Reducing the Memory Size of a Fuzzy Case-Based Reasoning System Applying Rough Set Techniques — Source link ↗

Florentino Fernández-Riverola, Fernando Díaz, Juan M. Corchado

Institutions: University of Vigo, University of Valladolid

Related papers:

- A polarity analysis framework for Twitter messages

- A comparative performance study of feature selection methods for the anti-spam filtering domain

- Hybrid artificial intelligence methods in oceanographic forecast models

- Development of CBR-BDI Agents: A Tourist Guide Application

- Maximum likelihood hebbian learning based Retrieval method for CBR systems

Reducing the Memory Size of a Fuzzy Case-Based Reasoning System Applying Rough Set Techniques

F. Fdez-Riverola, F. Díaz, and J. M. Corchado

Abstract— **Early work on Case Based Reasoning reported in the literature shows the importance of soft computing techniques applied to different stages of the classical 4-step CBR life cycle. This paper proposes a reduction technique based on Rough Sets Theory capable of minimizing the case memory by analyzing the contribution of each case feature. Inspired by the application of the minimum description length principle, the method uses the granularity of the original data to compute the relevance of each attribute. The rough feature weighting and selection method is applied as a pre-processing step prior to the generation of a fuzzy rule system which is employed in the revision phase of the proposed CBR system. Experiments using real oceanographic data show that the rough sets reduction method maintains the accuracy of the employed fuzzy rules, while reducing the computational effort needed in its generation and increasing the explanatory strength of the fuzzy rules.**

Index Terms—**Artificial Intelligence, Biological system modeling, Case based reasoning, Fuzzy systems, Reduced order systems**

I. INTRODUCTION AND MOTIVATION

CASE–Based Reasoning (CBR) systems solve problems by reusing the solutions to similar problems stored as cases in a case base. These systems have been successfully used in several domains such as diagnosis, prediction, control and planning [1]–[2]. However, a major shortcoming in these systems is the difficulty they have to evaluate the proposed solution and, where necessary, to repair it using domain-specific knowledge [3]. This is usually carried out through interaction with a human expert and is highly dependent on the problem domain. Furthermore, there are very few standard techniques for completely automating their construction, since each problem may be represented by a different data set and requires a customized solution [4]. This current weakness of CBR systems presents a major challenge.

A CBR system analyses a new problem situation and, by indexing algorithms, retrieves previously stored cases together with their solution by matching them against the new problem situation. The CBR system then provides a solution to the new problem by retrieving, adapting and reusing knowledge stored in the form of cases, in the case base. All of these actions are self-contained and may be represented by a cyclic sequence of processes. A typical CBR system is composed of four sequential steps that are called into action each time a new problem needs to be solved [5].

Over the past few years, a lot of work has been carried out using *soft computing* methods to improve CBR systems as a way of automating their life cycle and enhancing their accuracy [2]–[6]. These soft computing techniques (fuzzy logic, artificial neural networks, genetic algorithms and rough sets, mainly) work in parallel and enhance each other's problem-solving ability. Due to the successful results obtained, several real CBR applications have been successfully developed [2]–[6].

We have been working on the identification of techniques to automate the reasoning cycle of CBR systems used to solve dynamic problems [7]–[8]. Although in general, each specific problem and domain requires a particular solution, we have centered our efforts on forecasting the evolution of complex problems, with large case memories, for which there is a lack of knowledge, and for which an adaptive learning system is required. In these situations, in which the CBR systems need to deal with very large case bases, the soft computing methods embedded in their reasoning stages have difficulty in managing increasingly dynamic large memories.

While a CBR approach, has shown good results in forecasting the evolution of complex problems [9], the technologies embedded in the methodological framework proposed by the CBR methodology requires the use of reduction techniques capable of minimizing the case memory. This paper proposes a memory reduction technique that is able to select relevant features which can give a boost to the revision stage. This paper is also concerned with the definition of explanation methods able to justify the proposed solution

F. Fdez-Riverola is with the Department of Computer Science, University of Vigo, Edificio Politécnico, Campus Universitario As Lagoas s/n, 32004, Ourense, Spain (phone: +34 988 387015; fax +34 988 387001; e-mail: riverola@ uvigo.es).

F. Díaz is with the Science Faculty, University of Vigo, Edificio Politécnico, Campus Universitario As Lagoas s/n, 32004, Ourense, Spain (e-mail: fdiaz@uvigo.es).

J.M. Corchado is with the Computer Science Department, University of Salamanca, Plaza de la Merced s/n, 37008, Salamanca, Spain (e-mail: corchado@usal.es).

[10].

The reduction technique proposed by this paper is based on Rough Sets theory which is able to minimize the case base by analyzing the contribution of each feature. The rough feature weighting and selection method is applied as a pre-processing step previous to the generation of a fuzzy rule system employed in our case-based forecasting platform called CEFS (*Changing Environment Forecasting System*). Presently, the CEFS platform is able to combine several soft computing techniques at the retrieval and reuse stage whereas the system employs a set of Sugeno-Takagi (TSK) fuzzy systems [11] in order to validate the initial solution at the revision stage. Previous experiments have shown the effectiveness of the proposed revision subsystem and its superiority over other techniques [10], but, even so, the revision subsystem suffers from two major drawbacks: the large computational effort needed to carry out the revision process and the explanatory complexity of the fuzzy rules used for the final solution proposed by the system.

In order to evaluate the benefits of the proposed method, the CEFS platform is applied to a forecasting problem consisting in the prediction of the concentration of diatoms (a type of single-celled algae) in different water masses. Specifically, the available data is given by a biological database composed of several physical variables (temperature, PH, oxygen, PH, etc.) measured at different depths and belonging to several monitoring points along the north western coast of the Iberian Peninsula. The count of diatoms (cell/liter) at these points at different moments in time is also stored. These data values are complemented with data derived from satellite images stored separately

The paper is structured as follows: section II covers relevant work on case base maintenance and attribute selection for CBR systems while section III describes the Case-Based Reasoning platform used in this study; section IV introduces the Rough Set theory grounding and section V details the proposed Rough Set reduction technique; section VI describes the test bed of the experiments and the results obtained; and finally, section VII presents the conclusions and further work.

II. Review of Existing Memory Reduction/Maintenance Techniques and Attribute Selection Algorithms

As already mentioned, CBR systems solve problems by reusing the solutions to similar problems stored as cases in a case memory. However, these systems are sensitive to the cases present in the case memory and often their accuracy rate depends on the significance of the cases stored. Therefore, in CBR systems it is important to maintain a memory with an adequate number of cases, to eliminate noise and redundant cases and to maximize the levels of efficiency and generalization.

Case base maintenance, which refers to the task of indexing, adding, deleting and updating cases, is vital for guaranteeing the ongoing efficient performance of CBR systems. Case maintenance techniques have been categorised [12] as *competence preservation* and *competence enhancement* techniques. The first corresponds to redundancy reduction, which aims to remove internal cases in a cluster of cases of the same class and can help to preserve noisy cases as exceptions or border cases. The latter is effectively noise reduction, which aims to remove noisy or corrupt cases but can remove exceptional or border cases that may not be distinguishable from true noise. For this reason, a balance of both can be useful. Later editing techniques can be classified as hybrid techniques incorporating both competence preservation and competence enhancement stages.

Competence enhancement and hybrid techniques are not suitable for systems that have to make accurate predictions due to the errors that can be generated. Recent approaches on competence preservation are based on a competence model of the training data and use the competence properties of the cases to determine which cases to include in the edited set. Case competence was first introduced by [13] and developed by [14]. In the work of [15] two important competence properties were introduced - *coverage* and *reachability* sets for a case in a case base. These properties are used as the basis for a great number of editing techniques.

In the work of [16], a family of competence-guided editing methods for case bases is presented which combine both incremental and decremental strategies. [17] also use the coverage and reachability properties of cases in the Iterative Case Filtering (ICF) algorithm. In the work of [18], a series of Reduction Technique (RT) algorithms are presented. More recently, the work [19] presents an enhanced competence model which uses a Blame-Based Noise Reduction (BBNR) and Competence-based Redundancy Reduction (CRR) algorithms based on the previous ideas of [20] and [16] respectively.

However, the problem here is slightly different. In the domain of modeling changing environments in general, and making accurate predictions in particular, all the information is needed in order to achieve accurate results. As such, one cannot discard any variable in advance, invalidating previous research on maintaining the competence/preservation techniques that works over the whole case base.

Attribute selection or feature subset selection is also an attractive area of research, especially in the context of CBR-based analysis [21]. Generally, it implies the reduction of the number of attributes or features used to characterize a data set in order to enhance the performance of an algorithm for a given task. The memory reduction reduces the computational effort needed to carry out the revision process and to improve the explanatory power of the fuzzy rules obtained and used in the CEFS system.

Several knowledge-intensive algorithms have been used to perform feature selection [22] in CBR systems. However, domain specific knowledge is not always available for the selected domain. This prevents us from using explanation-based approaches for indexing and retrieving appropriate features [23]–[24]. Furthermore, the same set of features are

always used to describe each case in the case base, their values have been pre-computed, and no further processing is required to access the values. Therefore, we do not address the cost of evaluating features [25] and this study is restricted to using knowledge-poor feature selection approaches.

Three main approaches can be distinguished in the feature selection literature [26]–[27]: wrapper approaches, filter approaches, and embedded approaches.

As reported in [28], when the goal is the maximization of the accuracy of a given feature subset, the features selected should depend not only on the relevance of the data with respect to the target concept, but also on the learning algorithm. This defines the so-called wrapper approach and it implies that the selection algorithm searches for a good subset of features using the induction algorithm itself as part of the evaluation function. These approaches usually exert a high computational cost and before applying them, an enumeration of the available resources is quite critical; two main factors can accentuate the problem of selection [29]: the number of features and the number of instances.

When the learning algorithm is not used in the evaluation function, the "goodness" of a feature subset can be assessed referring only to the intrinsic properties of the data. This type of feature selection approach, which ignores the induction algorithm is known as the filter approach. Many algorithms were designed in the 90s under this approach, such as FOCUS[30], RELIEF [31] or its extension RELIEFF [32]–[33], Cardie's algorithm[34], the incremental feature selection method [35] or Bell and Wang's approach [36].

Finally, another type of feature subset selection has been identified in [37]: the embedded approach. In this case the feature selection process is carried out within the induction algorithm itself. Classical induction algorithms like ID3, C4.5 or CART are included in this category.

The proposed feature selection algorithm, which is based on the Rough Set Theory, follows the filter approach as shown in Section V. Moreover, the filter approach is the only feasible approach because, in the first place, its computational effort is tolerable, and secondly, because when it is used there is no evaluation metric for the attributes selected. As detailed in the following sections, the feature selection algorithm is used as a previous step for the generation of a TSK system in the revision stage and, at this point, we do not yet know the accuracy of the prediction given by the CBR.

III. DESCRIPTION OF THE CEFS PLATFORM

The study described in this paper was carried out in the context of the CEFS platform. CEFS is a structured hybrid system that can employ several soft computing techniques in order to accomplish the 4-steps of the classical CBR life cycle [38]. This section covers two main topics: (i) details of the architecture of the CEFS platform and (ii) presentation of the central points of the fuzzy revision method employed by the system.

In order to define in detail the group of complex problems

that our system is going to deal with, we use the term *changing environment* characterised by the following situations:

--The real process being forecast is stochastic, and come to us characterised by a high group of variables that evolve in a complex and irregular way with the time.

--In the domain there exists a lack of knowledge about the rules that define the deterministic behavior of the system, and its formalisation requires mechanisms able to manage uncertainty.

--The data about the process being forecast is expressed numerically, although on occasions they can be incomplete, imprecise or present inconsistencies that hinder their treatment.

--The variables of the systems can be charactersied by their tendency to involve problems related to heterocedasticity (different variance) and multicolinearity (lineal relationship among the different variables that explain a model) which need to be solved.

--The forecasting of a certain variable should take place with a given level of precision, and the presence of some mechanism that provides an explanation of the decision adopted by the system may well be necessary.

A. CEFS Platform Architecture

The CEFS platform is an extension of a previous successful system able to make predictions of red tides (discolourations caused by dense concentrations of microscopic sea plants, known as phytoplankton) [39]. The CEFS platform allows us to combine several soft computing techniques in order to test their suitability working together to solve complex problems. The core and the interfaces of CEFS have been coded in the Java language and new capabilities are being developed. The general idea is to have different programmed techniques able to work separately and independently in cooperation with the rest. The main goal is to obtain a general structure that could change dynamically depending on the type of problem. Fig. 1
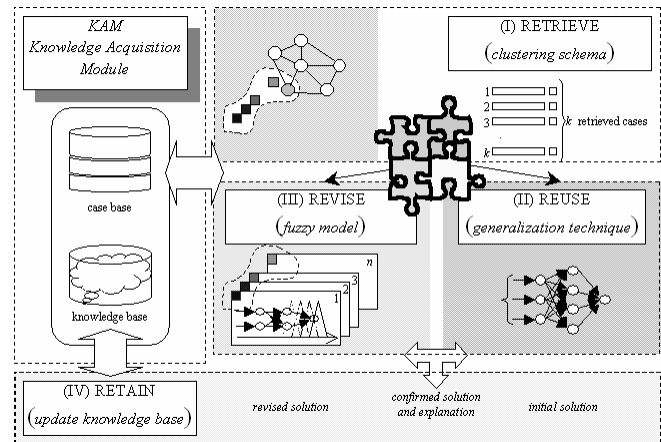

Fig. 1. CEFS platform architecture.

shows a schematic view of the system.

The left of Fig. 1 depicts the core of the platform, consisting of a KAM (Knowledge Acquisition Module). The

KAM is able to store all the information needed by the different techniques employed in the construction of a final CBR system. In the retrieve and reuse stages, several soft computing techniques can be used [2]–[6], while in the revise stage, our platform employs a set of TSK fuzzy systems in order to perform the validation of the initial solution proposed by the system [10].

Our aim in this work is to perform a feature subset selection step in order to reduce the original set of attributes used by the fuzzy revision subsystem (stage III on Fig. 1). The purpose is twofold: on the one hand, to decrease the computational effort needed for the generation of the n different fuzzy models, on the other, to simplify the complexity of the fuzzy rules that compose the explanation for the proposed solution.

B. Fuzzy Revision Module Generation

The first step in the generation of the TSK fuzzy model is the construction of an initial fuzzy system able to model the knowledge represented by the case base of the CBR [40]. This can be done following the advice of human experts, learning symbolic rules from artificial neural networks [41], using evolutionary strategies [42], applying fuzzy clustering to the data or using a hybrid approach as proposed here.

A novel method of fuzzy clustering able to extract interpretable fuzzy rules from a Radial Basis Function (RBF) neural network [43] is proposed in [44], and applied successfully in the work of [45]. Starting from the TSK fuzzy rule base obtained in the initial step, a measure of similarity is applied in order to reduce the number of fuzzy sets describing each variable. We use a similarity measure for identifying similar fuzzy sets and replace these with a common fuzzy set representative. If the redundancy in the model is high, merging similar fuzzy sets for each variable might result in equal rules that can also be merged, thereby reducing the number of rules as well [10]. As a result, the new fuzzy rule base increments the capacity of generalisation of the original TSK fuzzy system.

In order to generate several fuzzy rule bases with different generalization degrees, it is necessary to set up a λ-limit from which two membership functions can be considered analogous and therefore can be joined [46]. In our revision method, the parameter λ goes from 0.9 to 0.6 with decrements of 0.1 [47], generating four fuzzy rule bases corresponding to the TSK

R_j : If x_1 is A_{1j} and x_2 is A_{2j} and ... and x_M is A_{Mj}, Then $y = g_j(x_1, x_2, ..., x_M)$

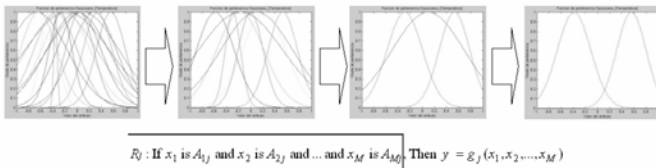Fig. 2. Simplification of the membership functions that describe each attribute of the case base. Figures from left to right stand for a different λ-value.

fuzzy systems (see Fig. 2).

The algorithm starts in an iterative way grouping membership functions attribute by attribute. In each iteration, the similarity S between all the membership functions for a given attribute is calculated, selecting the pair of functions that holds a higher degree of similarity providing that $S > \lambda$. The selected pair of functions is joined and the rule base is brought up to date with the new membership function. The algorithm continues until the maximum similarity between two membership functions belonging to any attribute is less or equal to λ. Finally, the fuzzy rules with similar antecedent part are merged (fuzzy rule of Fig. 2), and the consequent of the new rule is recalculated by means of following expression:

$$C_r = \frac{1}{k} \sum_{i=1}^{k} C_i \qquad (1)$$

where C_r is the consequent of the new generated rule and k represents the number of rules with similar antecedent.

IV. ROUGH SET THEORY

Rough Set theory, proposed by Pawlak, is an attempt to provide a formal framework for the automated transformation of data into knowledge [48]–[49]. It is based on the idea that any inexact concept (for example, a class label) can be approximated from below and from above using an indiscernibility relationship. Pawlak points out that one of the most important and fundamental notions to the Rough Set philosophy is the need to discover redundancy and dependencies between features [50].

The main advantages of Rough Set theory are that it: (i) provides efficient algorithms for discovering hidden patterns in data; (ii) identifies relationships that would not be found using statistical methods; (iii) allows the use of both qualitative and quantitative data; (iv) finds the minimal sets of data that can be used for classification tasks; (v) evaluates the significance of data and (vi) generates sets of decision rules from data.

A. Basic Concepts and Definitions

Briefly, the relevant Rough Set terminology is stated below. An information system is a pair $S = \langle U, A \rangle$, where U is a non-empty and finite set, called the universe, and A is a non-empty, finite set of attributes (or features). An equivalence relation, referred to as an indiscernibility relation, is associated with every subset of attributes $P \subseteq A$. This relation is defined as:

$$IND(P) = \{(x, y) \in U \times U : \text{for every } a \in P, a(x) = a(y)\} \qquad (2)$$

Given any subset of features P, any concept $X \subseteq U$ can be defined approximately by the employment of two sets, called lower and upper approximations. The lower approximation, denoted by $\underline{P}X$, is the set of objects in U which can be classified with certainty as elements in the concept X using the set of attributes P, and is defined as follows:

$$\underline{P}X = \cup\{Y \in U / IND(P) : Y \subseteq X\} \qquad (3)$$

The upper approximation, denoted by $\overline{P}X$, is the set of elements in U that can be possibly classified as elements in X, formally:

$$\overline{P}X = \cup\{Y \in U / IND(P) : Y \cap X \neq \varnothing\} \qquad (4)$$

The degree of dependency of a set of features P on a set of

features R is denoted by $\gamma_R(P)$, $0 \leq \gamma_R(P) \leq 1$, and is defined as:

$$\gamma_R(P) = \frac{card(POS_R(P))}{card(U)} \tag{5}$$

where

$$POS_R(P) = \bigcup_{X \in U / IND(P)} \underline{R}X \tag{6}$$

$POS_R(P)$ contains the objects of U which can be classified as belonging to one of the equivalence classes of $IND(P)$, using only features from the set R. If $\gamma_R(P) = 1$, then R functionally determines P.

Various extensions have been defined from the basic model proposed by Pawlak. Among these extensions the most outstanding is the Variable Precision Rough Set model (VPRS) which is a generalisation that introduces a controlled degree of uncertainty within its formalism [51]. This degree is established by an additional parameter ϕ.

B. Rough Sets as Reduction Technique

A major feature of the Rough Set theory is to find the minimal sets of data that can be used for classification tasks. In this sense, the notions of core and reduct of knowledge are fundamental for reducing knowledge preserving information. After stating the formal definitions of these concepts, the reduction process proposed by the methodology is outlined.

P is an independent set of features if there does not exist a strict subset P' of P such that $IND(P) = IND(P')$. A set $R \subseteq P$ is a reduct of P if it is independent and $IND(R) = IND(P)$. Each reduct has the property that a feature cannot be removed from it without changing the indiscernibility relation. Many reducts for a given set of features P may exist. The set of attributes belonging to the intersection of all reducts of P is called the core of P:

$$core(P) = \bigcap_{R \in Reduct(P)} R \tag{7}$$

An attribute $a \in P$ is indispensable if $IND(P) \neq IND(P \setminus \{a\})$. The core of P is the union of all the indispensable features in P.

The reduction technique stated by the methodology is especially suitable for reducing decision tables. A decision table is an information system of the form $S = \langle U, A \cup \{d\} \rangle$, where $d \notin A$ is a distinguished attribute called the decision attribute or class attribute. The elements of the set A are referred to as condition attributes. A decision table is a classifier that has as its internal structure a table of labelled instances. Given a novel instance, the classification process is based on the search of all matching instances in the table. If no matching instances are found, unknown is returned; otherwise, the majority class of the matching instances is returned (there may be multiple matching instances with conflicting labels). The indispensable attributes, reducts, and core can be similarly defined relative to a decision attribute or output feature. The precise definitions of these concepts can be fount in Pawlak's book on Rough Sets [49].

At this point, it is very important to use the classification rules (given by a decision table) with the minimal effort, and therefore, the simplification of decision tables is of primary importance. The simplification process comprises two fundamental tasks. On the one hand, reduction of attributes consists of removing redundant or irrelevant attributes, without losing any essential classification information. This goal is achieved by computing the reducts for the condition attributes relative to the decision attribute. On the other hand, reduction of attribute values is related to the elimination of the greatest number of condition attribute values, at the same time maintaining the classificatory power.

V. INTEGRATING THE ROUGH SET REDUCTION TECHNIQUE INTO THE CBR-SYSTEM

This section details the reduction technique based on the Rough Set theory, which is used in the CBR system to diminish the computational effort at the revision stage. This decrement of the computational load is due to the reduction of relevant features, which are passed to the revision stage. Since the underlying principle of the feature subset selection algorithm is the Minimum Description Length principle (a decision criterion which attempts to decrease the model complexity at the same time as preserving the model accuracy), the explanatory power of the fuzzy rules that endorse the final solution of the CBR system is also augmented [52]. The second subsection describes how to integrate the Rough Set technique into the CBR system.

A. The Feature Subset Selection Algorithm

The computation of the reducts and the core of the condition attributes from a decision table is a way of selecting relevant features. It is a global method in the sense that the resultant reduct represents the minimal set of features that are necessary to maintain the same classificatory power given by the original and complete set of attributes. A straighter method for selecting relevant features is to assign a measure of relevance to each attribute and choose the attributes with higher values.

In the Rough Set framework, the natural way to measure the prediction success is the degree of dependency defined above. However, this measure has been shown to be weak in assessing an estimation of the predictive accuracy of a set of condition attributes Q with regard to a class attribute d [53]. To overcome this deficiency, Düntsch and Gediga define the notion of rough entropy [54]. Based on this notion and its adaptation to the VPRS model (in order to exploit the knowledge that is provided by the observations in the boundary region or the uncertain area of the universe more efficiently), we have defined a coefficient that allows us to assess the significance of an attribute within a set of attributes [55]. The significance of an attribute $a \in Q$ is defined in a way that its value is greater when the removal of this attribute leads to a greater diminution of the complexity of the hypothesis $Q \setminus \{a\}$, and simultaneously, to a lesser loss of accuracy in the hypothesis. Implicitly, the underlying principle used to evaluate the relevance of an attribute in this way is the

Minimum Description Length Principle (MDLP) [52].

The associated complexity of a given set of condition attributes Q can be evaluated through the entropy of the partition $U / IND(Q)$, which will be denoted by $H(Q)$. On the other hand, the conditional rough entropy $H_\phi (d \mid Q)$ can be used to evaluate the accuracy that is achieved when the condition attributes Q are used to predict the value of the condition attribute d. Therefore, the formal definition of the ϕ-rough entropy, denoted by $RH_\phi (d \mid Q)$, is given by the following expression:

$$RH_\phi(Q,d) = H(Q) + H_\phi(d \mid Q) = H(Q) +$$

$$\left\{ \{1 - \gamma_{Q,\phi}(d)\} \log_2 |U| + \sum_{x_i \subseteq POS_{Q,\phi}(d)} \frac{|X_i|}{|U|} \log_2 \frac{|X_i|}{|U|} \right\} \quad (8)$$

$$= \{1 - \gamma_{Q,\phi}(d)\} \log_2 |U| - \sum_{x_i \subseteq POS_{Q,\phi}(d)} \frac{|X_i|}{|U|} \log_2 \frac{|X_i|}{|U|}$$

where X_i represents each one of the classes of the partition $U / IND(Q)$, the set $POS_{Q,\phi}(d)$ is the positive region of Q with regard to the decision attribute d, and $\gamma_{Q,\phi}(d)$ is the degree of dependence of attribute d on the set of attributes Q.

Then, the ϕ-significance of a condition attribute, $a \in Q$, with regard to the decision attribute d, denoted by $\sigma_{a,\phi}(Q, d)$, is defined as the variation that the ϕ-rough entropy suffers when the considered attribute is dismissed from Q. Namely, it is computed the term $\Delta_a RH_\phi(Q, d)$, given by the difference between $RH_\phi(Q, d)$ and $RH_\phi(Q \setminus \{a\}, d)$. Formally,

```
Function FEATURE_SUBSET_SELECTION (input: decisionTable, φ; output: outputFeatures)
{
00   begin
01      outputFeatures ← Features(decisionTable) /* the starting point is the complete set of input features */
02      γinitial ← CoefDependence(decisionTable, outputFeatures, TargetFeature(decisionTable), φ)
03      step ← 0; haltCriterion ← FALSE
04      while ¬haltCriterion do /* while it is not satisfied criterion of halt, remove features */
05         haltCriterion ← TRUE; step ← step + 1
06         σmax ← 1.0;
07         for each feature f ∈ outputFeatures do /* select the most irrelevant feature */
08            σf ← CoefSignificance(decisionTable, f, outputFeatures, TargetFeature(decisionTable), φ)
09            γ ← CoefDependence(decisionTable, outputFeatures \ {f}, TargetFeature(decisionTable), φ)
10            if ( σf < σmax ) then
11               γcurrent ← γ
12               feature_to_remove ← f
13         γthreshold ← γinitial*(1.0 − DecrementFactor(step));
14         if (γcurrent >= γthreshold ) then
15            outputFeatures ← outputFeatures \ {feature_to_remove} /* remove feature */
16            haltCriterion ← FALSE
17      end.
}
```
Fig. 3. Algorithm for feature subset selection.

$$\sigma_{a,\phi}(Q,d) = \Delta_a RH_\phi(Q,d) = RH_\phi(Q,d) - RH_\phi(Q \setminus \{a\}, d)$$
$$= \{H(Q) - H(Q \setminus \{a\})\} - \{H_\phi(d \mid Q \setminus \{a\}) - H_\phi(d \mid Q)\} \quad (9)$$

Fig. 3 provides a concise description of the algorithm that selects a subset of relevant features using the significant ϕ-rough coefficient to evaluate the relevance of a feature. The proposed algorithm for selecting relevant features is described according to the view proposed by [56]. These authors state that a convenient paradigm for viewing feature selection methods is that of heuristic search, with each state in the search space specifying a subset of the possible features. Following Blum and Langley's viewpoint, the four basic issues that characterise this method are:

-- The starting point in the space, which in turn influences the direction of search and the operators used to generate successor states. The proposed algorithm starts with all attributes and successively removes them (lines 1 and 15, respectively). This approach is known as backward elimination.

-- The organisation of the search. The feature selection algorithm is based on a greedy method to traverse the space in the event that an exhaustive search is impractical. At each point in the search, the proposed algorithm considers all local changes, namely, it evaluates the significance of each attribute of the current set of attributes (loop for).

-- The strategy used to evaluate alternative subsets of attributes. In this paper, the variation of the normalised ϕ-rough entropy has been chosen for this purpose. Specifically, at each decision point the next selected state is that one which results from removing the attribute with the least significant ϕ-rough coefficient (line 10).

-- A criterion for halting the search. In the algorithm, the criterion for halting is that the difference between the degree of dependency at initial state and the current one (both with respect to the decision) do not exceed a predefined threshold (line 14).
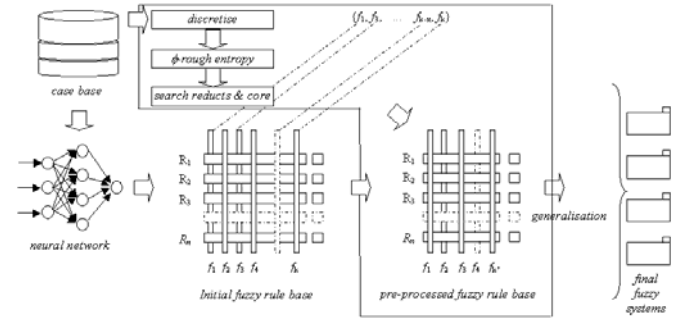
B. Rough Sets inside the CEFS Platform

Fig. 4. Rough Set pre-processing step.

Fig. 4 shows the meta-level process when incorporating the Rough Sets as a pre-processing step before the generation of the fuzzy revision subsystem.

For details related to the construction of the fuzzy systems starting from a Radial Basis Function neural network see [10]. The Rough Set process described here generates the initial fuzzy system and it is divided into three phases:

The first one discretises the cases stored in the case base. It is necessary in order to find the most relevant information using the Rough Set theory. The second one uses the significant ϕ-rough coefficient to select a subset of relevant features as described in subsection V.A (see Fig. 3). Finally, the last phase searches for reducts and the core of knowledge from the features selected in the previous phase, as explained in section IV.

The motivation for including the second phase is that the computation of reducts is a blind technique, where several combinations of a sufficient number of irrelevant features can become a reduct. The pre-selection of features leads to reducts with a lesser complexity and a higher predictive accuracy.

VI. CASE STUDY

In order to evaluate the proposed method, we use a biological database composed of several physical variables (temperature, PH, oxygen, PH, etc.) measured at distinct depths and belonging to different monitoring points along the north western coast of the Iberian Peninsula. These data values are complemented with data derived from satellite images stored separately. The satellite image data values are used to generate cloud and superficial temperature indices. Table I shows the original attributes taken into account for the generation of the initial fuzzy rule base versus the final selected attributes that constitute each optimized fuzzy rule. This table presents the sampling intervals in the third column. The fourth column presents the number of the variables used initially to construct the case, which is composed of 46 attributes and the fifth column indicates variables that are used
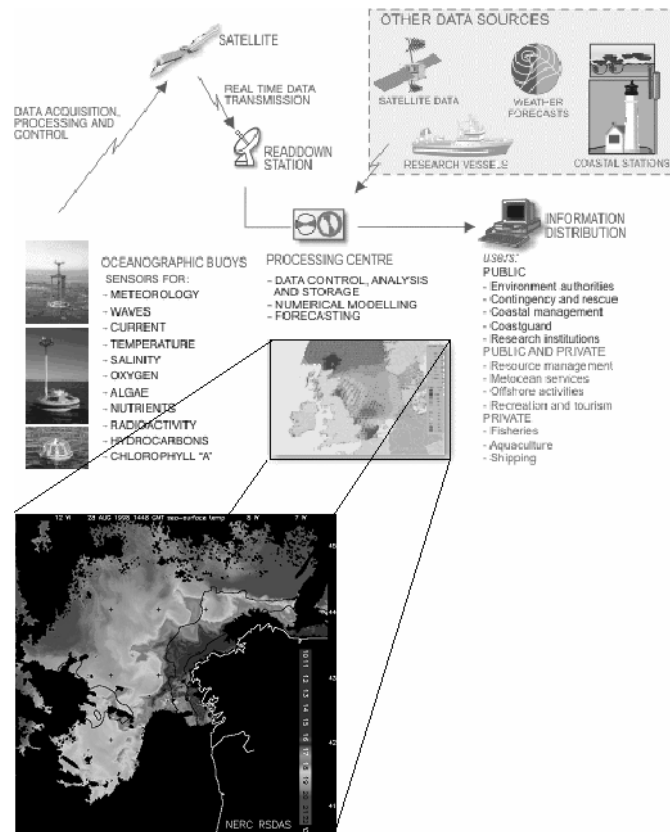

Fig. 5. Different sources of information.

to construct the cases after applying the proposed reduction technique.

Fig. 5 shows a schematic view of the whole data manipulated by the CEFS platform. The whole memory of the system consists of approximately 6300 cases, each one represented as a feature vector that holds 46 attributes.

The CEFS platform was configured to use the same techniques as in our previous work [10], where the fuzzy

revision method was successfully tested: (i) a Growing Cell Structure (GCS) neural network as retrieval method, (ii) a Radial Basis Function (RBF) neural network for the reuse step and (iii) the aforementioned set of TSK fuzzy systems working as the revision mechanism (see Table II). Specific

TABLE I
FEATURE SUBSET SELECTION CARRIED OUT BY THE
PROPOSED ROUGH SET ALGORITHM

Variable	Unit	Sampling interval	Number of initial measurements	Number of selected measurements
Identifier	numeric	-	(1)	(1)
Date	dd-mm-yyyy	weekly	2	1
Temperature	ºC	weekly	6	2
Oxygen	ml/l	weekly	6	1
PH	acid/based	weekly	6	1
Transmitance	%	weekly	6	1
Fluorescence	%	weekly	6	1
Cloud Index	[-2, 2]	diary every 12 hours	4	1
Bloom index	%	diary every 8 hours	4	1
Temperature index	[-2, 2]	diary every 12 hours	4	2
Recount of diatoms	cell/l	weekly	2	1

TABLE II
CONFIGURATION OF THE CEFS PLATFORM WITH ENHANCED
ROUGH SET SUPPORT

CBR STAGE	Technology	Input	Output	Process
Retrieve	GCS network	Problem descriptor	*k* similar cases	All the cases that belong to the same class to which the GCS associates the problem case are retrieved
Reuse	RBF network	Problem descriptor *k* similar cases	Initial solution	The RBF network is retrained with the *k* retrieved cases
feature subset selection	Rough Sets	Case base features	Subset of relevant features	Calculate the ϕ-rough coefficient, search for reducts and core
Revise	TSK Fuzzy system	Problem descriptor Initial solution	Confirmed solution	Different TSK fuzzy systems are created using the RBF network configuration with different degrees of generalization
Retain	GCS network RBF network Fuzzy systems	Problem descriptor Forecasting error	Configuration parameters of the GCS network, RBF network and Fuzzy systems	The configurations of the GCS network, the RBF network and the Fuzzy subsystems are updated according to the accuracy of the forecast

information about these techniques and its integration inside the CBR life cycle can be found in [47].

The main goal of the previous work was to develop a biological forecasting system capable of predicting the concentration of diatoms (a type of single-celled algae) in different water masses. The possibility of forecasting the concentration of diatoms is very important for obtaining a

valuable freshwater bioindicator, eliminating the need for a single group of organisms that can continually register the health of water masses. Diatoms are stationary and are therefore less able to avoid harmful conditions [57]. Indices based on diatom composition give more accurate and valid predictions than benthic macroinvertebrates, as they react directly to pollutants. Moreover, diatoms are better documented universally, they are sensitive to water quality changes and more importantly, this sensitivity is measurable by well-developed indices especially for community structure [58].

Although the experiments carried out in [10] showed the effectiveness and the straightforward improvement of the proposed fuzzy revision method over other approaches, some issues remained unsolved in order to deploy the application for real use. Concisely, the main drawbacks of the tested method were: (i) the time needed for generating each one of the optimal TSK fuzzy models and (ii) the explanatory complexity of the fuzzy rules used for the final solution

TABLE III
NUMBER OF SELECTED FEATURES *VS.* PARAMETER ϕ

parameter ϕ	number of selected features
0.0	16
0.01	12
0.025	10
0.05	8
0.1	8

proposed by the system (46 features in the antecedent of each fuzzy selected rule).

In order to solve these problems while maintaining the level of accuracy, in this paper we have proposed a feature subset selection algorithm based on Rough Set theory. As we can see in Table III, several ϕ values have been tested in order to obtain the most accurate set of representative features defining each problem case. For the current domain of diatom forecasting, the optimal number of features was 12 ($\phi = 0.01$), corresponding to the physical magnitudes measured with a

Fig. 6. Mann-Withney test carried out between each pair of models.

smaller level of depth and those generated from satellite images.

A crucial aspect in this experiment is the accuracy level of the Rough Set based revision subsystem and its comparison with the original. Starting from the error series generated by the different models, the Kruskall-Wallis test has been carried out. Since the P-value is less than 0.01, there is a statistically significant difference among the models at the 99.0% confidence level. Fig. 6 shows a multiple comparison

procedure (Mann-Withney) used to determine which models are significantly different from the others. The experiments were made with a data set of 448 cases randomly taken from the case base. It can be seen that the CBR with TSK fuzzy revision subsystem (CBR TSK) presents statistically significant differences with the rest of the models, whilst it is as accurate as the simplified method presented here (CBR ϕ (TSK)).

Therefore, the selected value of parameter ϕ leads to a simplified TSK subsystem at the revision stage (with a reduction of 74% in the number of attributes) and a loss of accuracy in the CBR ϕ (TSK) of about 12% in the mean squared error. Moreover, the statistical tests show that this difference between the two models is not significant. Fig. 7
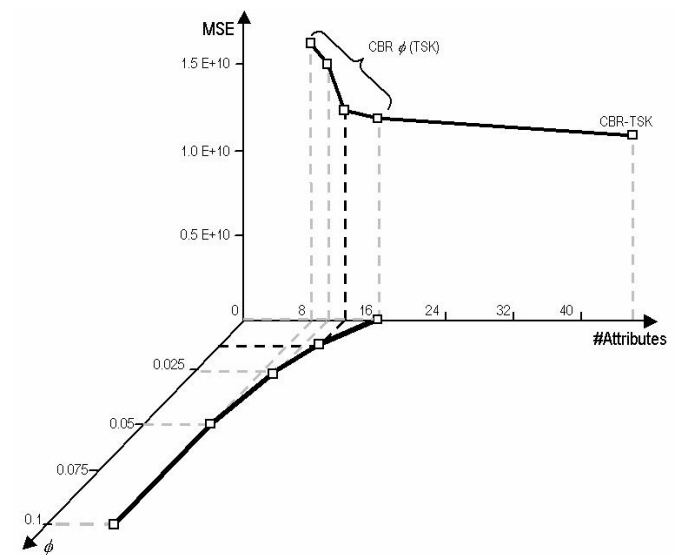

Fig. 7. Mean squared error (MSE) of several CBR ϕ (TSK) systems and the CBR-TSK system.

shows the mean squared errors of several CBR ϕ (TSK) systems and the CBR-TSK system.

The time spent in the execution of the pre-processing step plus the whole generation of the TSK fuzzy systems (about 2
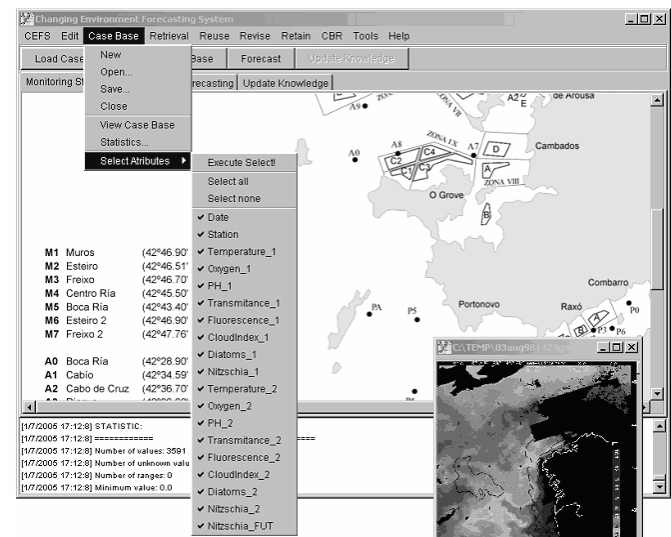

Fig. 8. CEFS platform look and feel for biological forecasting.

hours in a Pentium IV processor) was 80% less than the time required for generating the original fuzzy revision subsystem. This timesaving operation is motivated by the simplified fuzzy rule base employed by the greedy algorithm used to generate each one of the TSK fuzzy systems. The benefits obtained from the enhanced method allow us to deploy applications such as CEFS for real use (see Fig. 8).

Another relevant circumstance derived from the adoption of the proposed system was an increase in the explanatory strength of the justification generated by the final CBR system. Initially the feature vector describing a problem was composed of 46 attributes, the same as the fuzzy rule antecedents, now the system is able to produce an explanation based on only 12 main features with the same level of accuracy.

VII. CONCLUSIONS AND FURTHER WORK

This paper introduces a new reduction technique based on Rough Set theory that can be applied for improving a previous successful method that automates the revision stage of CBR systems.

Empirical studies show that this reduction technique allows us to obtain a more general knowledge of the model and gain a deeper insight into the logical structure of the system to be approximated. Employing the simplified fuzzy rule base as the starting point to generate the fuzzy revision subsystem proposed in [10], leads to a dramatic decrease in the time needed for this task while maintaining an equivalent generalized accuracy.

These benefits are augmented with the simplicity of the new fuzzy rules used by the CBR system as an explanation for the final adopted solution. In this way, it is interesting to define a formal measure in order to rate and compare the explanation strength of these fuzzy rules. Related to this last point, we are working on the representation of each variable as an overlapping linguistic property set 'low', 'medium' and 'high' based on the idea of [59].

Due to the suitability showed by the Rough Set theory working together with other soft computing techniques, we are also interested in the development of new ways to put together this formalism with the existing techniques coded in the CEFS platform.

REFERENCES

[1] Watson, I.: Applying Case-based Reasoning: Techniques for Enterprise Systems, Morgan Kaufmann, San Mateo, CA (1997)

[2] Pal, S.K., Dilon, T.S., Yeung, D.S.: Soft Computing in Case Based Reasoning, Springer Verlag, London (2000)

[3] Lenz, M., Bartsch-Sprl, B., Burkhard, H.D., Wess, S. (eds.): Case Based Reasoning Technology - From Foundations to Applications. Springer Verlag, LNAI 1400 (1998)

[4] Bergmann, R., Breen, S., Göker, M., Manago, M., Wess, S.: Developing Industrial Case-Based Reasoning Applications - The INRECA Methodology. Springer Verlag, LNAI 1612 (1998)

[5] Aamodt, A., Plaza, E.: Case-Based Reasoning: foundational Issues, Methodological Variations, and System Approaches. AI Communications, Vol. 7 (1). (1994) 39–59

[6] Sankar, K.P., Simon, C.K.S: Foundations of Soft Case-Based Reasoning, Wiley-Interscience, Hoboken, New Jersey (2004)

[7] Fyfe, C., Corchado, J.M.: Automating the construction of CBR Systems using Kernel Methods. International Journal of Intelligent Systems, Vol. 16 (4). (2001) 574–586

[8] Corchado, J.M., Aiken, J., Corchado, E.S., Fyfe, C., Fdez-Riverola, F., González, M.: Maximum Likelihood Hebbian Learning Based Retrieval Method for CBR Systems. Proc. of the Fifth International Conference on Case-Based Reasoning, (2003) 107–121

[9] Corchado, J.M., Aiken, J.: Hybrid Artificial Intelligence Methods in Oceanographic Forecasting Models. IEEE SMC Transac. Part C, Vol. 32 (4). (2002) 307-313

[10] Fdez-Riverola, F., Corchado, J.M.: Employing TSK Fuzzy models to automate the revision stage of a CBR system. Current Topics in Artificial Intelligence. Springer Verlag, LNAI 3040 (2004) 302–311

[11] Takagi, T., Sugeno, M.: Fuzzy identification of systems and its applications to modeling and control. IEEE Transactions on Systems, Man and Cybernetics, Vol. 15. (1985) 116–132

[12] Brighton, H., Mellish. C.: Advances in Instance Selection for Instance-Based Learning Algorithms. Data Mining and Knowledge Discovery, Vol. 6. (2002) 153–172

[13] Smyth, B., McKenna, E.: Remembering to Forget: A Competence Preserving Case Deletion Policy for CBR Systems. Proc. of the Fourteenth International Joint Conference on Artificial Intelligence, (1995) 337–382

[14] Zhu, J., Yang, Q.: Remembering to Add: Competence Preserving Case-Addition Policies for Case-Base Maintenance. Proc. of the Sixteenth International Joint Conference on Artificial Intelligence, (1997) 234–239

[15] Smyth, B., McKenna, E.: Modelling the Competence of Case-Bases. Proc. of the Fourth European Workshop on Case-Based Reasoning, (1998) 208–220

[16] McKenna, E., Smyth, B.: Competence-guided Editing Methods for Lazy Learning. Proc. of the Fourteenth European Conference on Artificial Intelligence, (2000) 60–64

[17] Brighton, H., Mellish, C.: Advances in Instance Selection for Instance-Based Learning Algorithms. Data Mining and Knowledge Discovery, Vol. 6. (2002) 153–172

[18] Wilson, D.R., Martínez, T.R.: Instance Pruning Techniques. Proc. of the Fourteenth International Conference on Machine Learning, (1997) 404–411

[19] Delany, S.J., Cunninghan, P.: An Analysis of Case-Base Editing in a Spam Filtering System. Proc. of the Seventh European Conference on Case-Based Reasoning, (2004) 128–141

[20] Aha, D.W., Kibler, D., Albert, M.K.: Instance-Based Learning Algorithms. Machine Learning, Vol. 6. (1001) 37–66

[21] Aha, D.W., Bankert, R.L.: Feature Selection for Case-based Classification of Cloud Types: An Experimental Comparison. Proc. of the AAAI-94 Workshop on Case-Based Reasoning, (1994) 106–112

[22] Ashley, K.D., Rissland, E.L.: Waiting on weighting: A symbolic least commitment approach. Proc. of the Seventh National Conference on Artificial Intelligence, (1988) 239–244

[23] Cain, T, Pazzani, M.J., Silverstein, G.: Using domain knowledge to influence similarity judgement. Proc. of the Case-Based Reasoning Workshop, (1991) 191–202

[24] Ram, A.: Indexing, elaboration, and refinement: Incremental learning of explanatory cases. Machine Learning, Vol. 10. (1994) 201–248

[25] Owens, C.: Integrating feature extraction and memory search. Machine Learning, Vol. 10. (1993) 311–340

[26] Kohavi, R, John, G. Wrappers for feature subset selection. Artificial Intelligence, 97(1-2). (1997) 273-324

[27] Das. S. Filters, wrappers and a boosting-based hybrid for feature selection. In Proc. 18th. Intl. Conf. On Machine Learning (2001)

[28] Kohavi, R, John, G. and Pfleger, K. Irrelevant features and the subset selection problem. In Proc. 11th Intl. Conf. on Machine Learning, (1994) 121-129

[29] Liu. H., Sedtiono, R. Incremental feature selection. Applied Intelligence, 9(3). (1998) 217-230

[30] Almuallin, H, Dietterich, T. G. Learning with many irrelevant features. In Proc. AAAI-91, (1991) 547-552

[31] Kira, K., Rendell, L.A. The feature selection problem: traditional methods and a new algorithm. In Proc. AAAI-92, (1992) 122-126

[32] Kononenko, I. Estimating attributes: analysis and extension of RELIEF. In Proc. European Conf.. on Machine Learning, (1994) 171-182

[33] Witten, I., Frank, E. Data Mining: Practical Machine Learning Toools and Techniques with Java Imprementations. Morgan Kaufman (2000)

[34] Cardie, C. Using decision trees to improve case-based learning. In Proc. 10th Intl. Conf. On Machine Learning, (1993) 25-32

[35] Liu, H., Motoda, H. Feature Selection for Knowledge Discovery and Data Mining. Kluwer Academic Press (1998)

[36] Bell, D. A., Wang, H. A formalism for relevance and its application in feature subset selection. Machine Learning, 41 (2000) 175-195

[37] Blum, A. I., Langley, P. Selection of relevant features and examples in machine learning. Artificial Intelligence, 97 (1997) 245-271

[38] Riesbeck, C.K., Schank, R.C.: Inside Case-Based Reasoning, Lawrence Erlbaum Associates, Hillsdale, NJ, US (1999)

[39] Fdez-Riverola, F., Corchado, J.M.: FSfRT, Forecasting System for Red Tides. Applied Intelligence, Vol. 21 (3). (2004) 251–264

[40] Takagi, T., Sugeno, M.: Fuzzy identification of systems and its applications to modeling and control. IEEE Transac. on Systems, Man and Cybernetics, Vol. 15. (1985) 116–132

[41] Towell, G., Shavlik, J.: Extracting refined fuzzy rules from knowledge-based neural net-works. Machine Learning, Vol. 13. (1993) 71–101

[42] Jin, Y., von Seelen, W., Sendhoff, B.: On generating FC3 fuzzy rule systems from data using evolution strategies. IEEE Transac. on Systems, Man and Cybernetics, Vol. 29 (6). (1999) 829–845

[43] Fritzke, B.: Fast learning with incremental RBF Networks. Neural Processing Letters, Vol. 1 (1). (1994) 2–5

[44] Jin, Y., von Seelen, W., Sendhoff, B.: Extracting Interpretable Fuzzy Rules from RBF Neural Networks, Technical Institut für Neuroinformatik, Ruhr-Universität Bochum, January (2000)

[45] Fdez-Riverola, F., Corchado, J.M.: CBR based system for forecasting red tides. Knowledge-Based Systems, Vol. 16 (5–6). (2003) 321–328

[46] Setnes, M., Babuška, R., Kaymak, U., Lemke, R.: Similarity measures in fuzzy rule base simplification. IEEE Transac. on Systems, Man and Cybernetics, Vol. 28. (1998) 376–386

[47] Fdez-Riverola, F.: Neuro-symbolic model for unsupervised forecasting of changing environments. Ph.D. diss., Dept. of Computer Science, Vigo University, Spain (2002)

[48] Pawlak, Z.: Rough Sets. International Journal of Computer and Information Sciences, Vol. 11. (1982) 341–356

[49] Pawlak, Z.: Rough Sets: Theoretical Aspects of Reasoning about Data. Kluwer Academic Publishers, Dordrecht (1991)

[50] Pawlak, Z.: Rough sets: present state and the future. Foundations of Computing and Decision Sciences, Vol. 11 (3-4). (1993) 157–166

[51] Ziarko, W.: Variable Precision Rough Set Model. Journal of Computer and System Sciences, Vol. 46. (1993) 39–59

[52] Rissanen, J.: Minimum description length principle. In Kotz, S. and Johnson, N.L. (eds.). Encyclopedia of Statistical Sciences. John Wiley and Sons, New York (1985) 523–527

[53] Düntsch, I., Gediga, G.: Statistical evaluation of rough set dependency analysis. International Journal of Human-Computer Studies, Vol. 46. (1997) 589–604

[54] Düntsch, I., Gediga, G.: Uncertainty measures of rough set prediction. Artificial Intelligence, Vol. 106. (1998) 77–107

[55] Díaz, F., Corchado, J.M.: A method based on the Rough Set theory and the MDL principle to select relevant features. Proc. of the X CAEPIA - V TTIA, Vol. 1. (2003) 101–104

[56] Blum, A.L., Langley, P.: Selection of relevant features and examples in machine learning. Artificial Intelligence, Vol. 97. (1997) 245–271

[57] Jan, R-Q., Daiand, C-F., Chang, K-H.: Monitoring of Hard Substrate Communities. In Calow P. and Petts G.E. (eds.). The Rivers Handbook - Hydrological and Ecological Principals II, Blackwell Scientific Publications, London (1994)

[58] Dixit, S.S., Smol, J.P., Kingston, J.C., Charles, D.F.: Diatoms: powerful indicators of environmental change. Environmental Science and Technology, Vol. 26 (1). (1992) 23–33

[59] Pal, S.K., Mitra, P.: Case Generation: A rough-fuzzy approach. Proc. of International Conference on Case Based Reasoning, (2001) 236–242