International Journal of Information, Control and Computer Sciences
ISSN: 2517-9942
Vol:2, No:3, 2008

Automated Knowledge Engineering

Sandeep Chandana, Rene V. Mayorga, and Christine W. Chan

***Abstract*—This article outlines conceptualization and implementation of an intelligent system capable of extracting knowledge from databases. Use of hybridized features of both the Rough and Fuzzy Set theory render the developed system flexibility in dealing with discreet as well as continuous datasets. A raw data set provided to the system, is initially transformed in a computer legible format followed by pruning of the data set. The refined data set is then processed through various Rough Set operators which enable discovery of parameter relationships and interdependencies. The discovered knowledge is automatically transformed into a rule base expressed in Fuzzy terms. Two exemplary cancer repository datasets (for Breast and Lung Cancer) have been used to test and implement the proposed framework.

***Keywords*—**Knowledge Extraction, Fuzzy Sets, Rough Sets, Neuro–Fuzzy Systems, Databases

I. INTRODUCTION

ORIGINALLY when AI was conceptualized in the early 1950's, the importance of knowledge in a consumable form was not recognized. AI researchers in the past two decades have come to appreciate the fact that Domain Specific Knowledge is of great value to solve significant problems. And the fact that despite of the powerful computing capabilities of a computer, Humans are still considered the best problem solvers due to the ignorance of the said machines, has been accepted. The aspect of Knowledge Engineering has thus stemmed from the thirst for machine knowledge. Since knowledge systems lack the capability to tap into the unsaid potential of knowledge. Knowledge engineering (KE) comprises of detailed and meticulous articulation and organization of knowledge, the way a house is built, in order to achieve maximum efficiency from the system. By now, it has become more than obvious that KE's prime function is to extract knowledge from the said source and integrate the same into knowledge systems in an electronic / computer usable format.

Knowledge (Acquisition): Knowledge is an entity which does not exist in free form in the universe, rather is in abundance in the form of tacit knowledge. Tacit knowledge is explicit but is embodied in an experts' experience, work skills and intuition or simply in their knowledge of '*what really works and how to make it work*' [1]. The most important issue thus becomes being able to deal with construction of knowledge based systems and assimilating such a *variety* of knowledge (sources). Off the various sources / forms of knowledge, databases are considered to be an important location due to extensive use of databases across all fields and areas of application.

Knowledge Discovery in Databases (KDD): Database Systems originally conceived to act as a storage device, have now grown out of that perception to support many other novice new applications. Database mining is one such application. It primarily is concerned with the aspect of *discovering* valuable information (or knowledge) from very large databases, whose existence is unknown to the user or the expert. The humongous growth of databases and their size has made it literally impossible to analyze the data by any one individuals' intellect, but the need to understand the data has also increased manifold. The central aim is to transform information present inside the database into knowledge or some kind of relationships or any domain specific characteristics.

These relationships or characteristics in a way bring to light the interdependencies and the functional dependencies of the various objects found in the database. And in most cases, these relationships may be determined (in a scalable database) without the need for any sophisticated techniques. Having stated, that, one should also bring to notice, the growing need of statistical and mathematical tools to analyze the data and bring about facts which are relatively more reliable than human inference. As stated by Ohsuga [2], data concerning a physical process has a *latent structure*, and knowledge discovery entails determining the causal relationships embedded within such a structure. Such knowledge should then be useful in identifying system response with corresponding stimuli. By the virtue of this definition, KDD encompasses data mining, pattern recognition and such other information (knowledge) retrieval (resp. extraction) processes.

Classification: In general, knowledge discovery is aimed to identifying statistically signification patterns in data based on which individual patterns are located within the test or unseen data [4]. Abstract level classification forms the basis for reasoning, learning and decision making. Thus knowledge has a definitive relationship with classification patterns which may be used in a way to derive secondary knowledge from primary knowledge / information. Database Mining and KDD have been primarily directed towards classification. Classification

Sandeep Chandana in 2006 obtained his M.A.Sc. in Industrial Systems Engineering, from the University of Regina, Canada,. He is currently pursuing a Ph. D. at the University of Calgary, email: schandan@ucalgary.ca.
Dr. Rene V. Mayorga is Associate Professor and Head of the Wise & Intelligent Systems & Entities Lab, Industrial Systems Engineering, University of Regina, Canada, email: Rene.Mayorga@uregina.ca.
Dr. Christine W. Chan is a Canada Research Chair (Energy and Environmental Informatics), Professor and Head of the Energy Informatics Lab, Software Systems Engineering, University of Regina, Canada, email: Christine.Chan@uregina.ca.

in principle aims to segregate the given objects in various classes in an effort to allocate a known solution to the elements belonging to the respective classes i.e. classification identifies classes with solutions. The domain knowledge does not only help in mapping the input-output space but also guides KDD by asserting value to the discovered knowledge in relation to the problem or process. [5]. Thus problem solving based on classification reduces down to, initially classifying and then processing as per the solution match. Classes are defined as per a set of given conditions, which are in turn constructed based on certain problem or domain specific knowledge. A simple knowledge system based on principles of classification essentially involve the following processes,

a) data abstraction
b) knowledge mapping to predefine solutions
c) refinement of solution space

Data abstraction and solution space refining, in principle involve pruning of the respective search space (based on certain parameters). And taking into account the size of the present day databases, emphasis should be laid more on such pruning tasks than anything else.

II. Preliminaries

This section aims to present some of the philosophy and rationale behind the presented work. Discussed in certain detail are, problem identification, scope of the work.

Problem Definition: The work aims to address the various issues which have been touched in the previous sections. Information Retrieval and Knowledge Acquisition have become tasks which require extra attention, in terms of processing capabilities and importance. With the growing size of databases and the ever increasing need for information processing, we have come to such a stage that human participation only slows down the process. Despite of the intelligence and problem solving skills, due to the inferior computing capabilities, human data interpreters have become obsolete.

a.) Manual SQL queries to obtain information about any prominent changes in the database trends are no longer a feasibility, given the enormity of the database structures. None of the existing learning algorithms posses the capacity to deal with large data sets effectively; *Data reduction / pruning is one approach of addressing large datasets*.

b.) Now assuming that we have the power to deal with large datasets, the next issue that needs attention is of efficiently discovering relationships and patterns within the given data. This is otherwise the issue of knowledge acquisition and discovering previously unknown information. Conventional pattern recognition systems fail when they encounter discrepancies in the data, and in practical applications data is not always in the best of the conditions, nevertheless the need to process such unruly data doesn't diminish; *Knowledge Discovery in large Databases is one way of approaching this issue.*

c.) Given all the requisite features, one tends to run into data with missing values or with a tinge of uncertainty. Conventional techniques of classical set theoretical classification tend to collapse given a situation wherein an object belongs to more than one set with certain discrete levels of membership with each. *Rough Sets, Multi Sets and Fuzzy Sets address this issue to a certain extent.*

e.) Rough Sets are not efficient in dealing with data sets made up of continuous attribute values. Various quantization or Discretization techniques exist to address this issue, but no global technique applicable to a variety of data sets exists.

f.) Rough Sets are purely rule based and due to the limited number of rules that such a system has, it fails to efficiently evaluate new cases or unseen situations. A Rough system can thus be said to be very fragile at its boundaries. Further details about Rough Set Theory shall be discussed in the following sections.

g.) Adaptive Neuro – Fuzzy Inference Systems inherently are disadvantaged in modeling systems which have a large number of inputs or outputs. Though time efficient, the amount of pre – processing required before an ANFIS model can be generated, can be enormous depending upon the given problem or situation.

Scope: In the following sections, we have described the details the concept of automatic knowledge extraction from a database. The system is modular in nature and adopts concepts from various machine learning and data processing paradigms. In particular we have relied upon the likes of Rough Sets, Fuzzy Logic and Neural Networks. Following is a brief rundown of the tasks performed by the integral parts of the automated knowledge extractor;

Preprocessing of Data

1. Preprocessed the data in order to transform it into usable format.
2. Conducted consistency and indiscernibility checks on the data.
3. Determined the interdependencies of the involved object parameters or the attributes through both the Discernibility Matrix and by calculation of the Degree of Dependency.
4. Developed a reduct finding program based on QuickReduct Algorithm
5. Developed another reduct finding program based on Greedy Algorithm

Processing of Data

6. Developed a rule base from the respective Reducts of the data
7. Developed an ANFIS based on the preprocessed data.
8. Developed an ANFIS based on the un – processed data.

System Evaluation

9. Presented a comparison between the internal structures of the three techniques.

The authors have used two data sets to test and validate the proposed intelligent knowledge extractor viz,

a) Breast Cancer (Orange) Data set and,
b) Lung Cancer (Orange) Data set.

The classification system aims to determine if the patient should be tested further for the given ailment or not. Thus it is expected to perform at an assistive level and aide in decision making. The three components of the individual cancer classification system have been implemented in order to work complimentary to each other. Further, each of system have been implemented using the following techniques,

a) Rough Sets (Reducts & Rules)
b) Adaptive Neuro – Fuzzy Inference Systems (standalone Model)
c) Rough – ANFIS (Rough Preprocessed ANFIS model)

III. Theory

Some background information about various concepts has been provided for a better understanding of the readers.

Rough Set Theory [6]**:** was developed in Poland in the early 1980s by Z. Pawlak. It deals with the classificatory analysis of imprecise, uncertain or incomplete information. And this information is in the form of data obtained from practical or real time applications. Basically it is the approximation of sets that are difficult to describe, given the available information. Each of the objects or cases is described by the set of attributes. The attribute set may be divided into two disjoint subsets i.e. *Conditional* attributes and *Decisional* attributes. Conditional attributes express descriptive information about the objects while the Decisional attributes express the respective conclusions or results. An ordered representation of these subsets in the form of a table is called a *Decision Table.*

Indiscernibility Relation: Indiscernibility is one of the primary properties of rough set. Informally, two objects under consideration are said to be indiscernible if we cannot differentiate between the two based on the given attributes / information set. Thus indiscernibility relation is always local to a set of attributes and an indiscernibility relation consists of all objects which stand indiscernible under the influence of a set of attributes. By the virtue of this definition, we can state that an indiscernibility relation partitions the *universe of discourse* (a super set of all elements) into many disjoint subsets or equivalence classes. An equivalence class of an object is thus a collection of objects that are indiscernible to the one under consideration**.**

Approximations: The indiscernible and discernible classes are the building blocks in order to put together all similar classes. And typically objects which have the same decisional attribute are clubbed together. Following these guidelines, one may place the classes into various regions as per the following.

- The lower approximation with respect to a given set of attributes is the collection of cases whose equivalence classes are fully contained in the set of cases we want to approximate. [7]

- The upper approximation with respect to a given set of attributes is the collection of cases whose equivalence classes are at least partially contained in the set of cases we want to approximate. [7]

Therefore lower approximation is a subset of the upper approximation and when these two are equivalent to each other, rough sets transform into conventional classical sets. Further, there is the following set of regions which pertain to class classification, viz,

- The Positive Region comprises of all cases which definitely belong to the concept or set that is being approximated.

- The Positive Region comprises of all cases which definitely do not belong to the concept or set that is being approximated and,

- Boundary region is the difference between the upper and lower approximations.

Rules: A statement of the form "if *Condition* then *Decision*" is called a decision rule or a rule in general. Relating the same to the aforementioned to concepts of Rough Sets, an If - then rule would have more than one conditional parts match the same Decisional attribute in the boundary region of the set which is being approximated. Further a distinct decision is representative of a case which belongs to either the positive or the negative regions of the set approximation. A case where not all attributes are known leads us to the phenomenon of upper approximation. The main challenge in inducing rules is that of determining the attributes which are prime to the quality of the given set approximation and then articulating with the data set based on the same. Not only do reducts help in fine tuning the search space but also help generalize the rule base. An over fitted rule base tends to incorporate the numerous unnecessary features that are possessed by the data i.e. noise and any data specific but domain independent trends in the data. Thereby, the rough set theory generated rules are more reliable since they are based on approximations of the sets and not alone on descriptions or definitions of the concept or set.

Adaptive Neuro - Fuzzy Inference Systems (ANFIS): The ANFIS proposed by Jang [8] can be described as adaptive networks which are similar to Fuzzy Inference Systems functionally. They also have outlined a methodology to help decompose the parameter set (of the adaptive network nodes), so as to help implement the Hybrid Learning algorithm within the said systems. A successful attempt to represent the Sugeno and Tsukamoto Fuzzy models through ANFIS has been undertaken and further, it is also observed that the Radial Basis Function Network when subjected to certain constraints is equivalent to the ANFIS functionality. The idea of such an approach towards modeling systems is to interpret fuzzy rules in terms of a neural network. The fuzzy sets can be representative of the weights and the input – output functions along with the rules are representative of the neurons of a network. The (hybrid) learning as well is implemented as in a connectionist system i.e. the system learns by continuously modifying and adapting the neural structure and the parameters. The advantage of such a system is that the learning can be interpreted from perspectives of both neural and fuzzy systems. And more importantly such a system enables viewing the problem solution in a linguistic fashion

Architecture: In the simplest terms, the structure of an ANFIS consists of a first layer where the inputs are mapped to their respective and relevant input membership functions. These membership functions taken it on to the rules layer further and then onto the output membership functions and finally to the output characteristic function which computationally produce the final single[1] valued output.

The following synopsis about the ANFIS architecture has been adopted from [9].

Consider a set of standard Sugeno style if then rules i.e.
Rule1: if x is A and y is B, then F1 = px + qy + r and;
Rule2: if x is C and y is D, then F2 = sx + ty + u.

Modeling the above using an ANFIS would result in a 5 layer network (excluding the input and the output layers) with the following sequential operations being accomplished,

Layer 1: The objective function of the nodes (all of the nodes in the same layer are characterized by the same objective function) is basically to assign a relevant fuzzy membership function to the said inputs i.e.

$$\begin{aligned} O_{1,i(1-2)} &= \mu_{A,i}(x) \\ O_{1,i(3-4)} &= \mu_{B,i-2}(y) \end{aligned} \quad (1)$$

Moreover, initially a user chosen parameterized general membership function is also adopted and the parameters of μ_{Ai} , μ_{Bi} are assigned some random values to start off with.

Layer 2: The outputs of the individual preceding layers are multiplied to retrieve the first set of real parameters which are called the premise parameters. i.e.

$$O_{2,i} = w_i = \mu_{Ai}(x)\mu_{Bi}(y) \quad (2)$$

Layer 3: The outputs of this layer are normalized or weighted weights i.e.

$$O_{3,i} = \overline{w_i} = \frac{w_{i(1-2)}}{w_1 + w_2} \quad (3)$$

Layer 4: Unlike the nodes from layers 2 and 3, layer 4 nodes are adaptive with the following output, (the final output individual vector components are computed). The parameters of p, q, r, s, t, u, are all called Consequent parameters.

$$O_{4,i} = \overline{w_i} F_i \quad (4)$$

Layer 5: Usually composed of a single static node, the operation of summing up the incoming individual vector components of the final single valued output.

$$O_{5,i} = \sum_i \overline{w_i} F_i \quad (5)$$

As described earlier, the ANFIS network shall comprise of a minimum of 5 layers with at least two of them being adaptive in nature. This structure generates two sets of varied parameters to solve for, namely the premise and the consequent parameters. These parameters and their values are generated appropriately during the learning process (as described in the following section).

Hybrid Learning: Conventionally the Gradient optimization methods or back propagation techniques have been used to identify and optimize the various nodal parameters associated with adaptive networks. But Jang [8] proposed a hybrid learning rule approach to identify the parameters. This method involves combining the back propagation steepest descent and least squares method for a faster identification. The process of continuous parameter update can be achieved by two methods i.e. off – line or batch learning which involves updating the parameters iteratively at the end of all training data pair runs and on- line learning wherein the parameters are updated posthumously after every layered data run.

Batch Learning: Consider the following output function for an adaptive network

$$O = f(i, S) \quad (6)$$

where '*O*' is the output function, 'i' inputs set and '*S*' is the parameter set.

Now if there exists a function *'h'* such that applying it to Eq (1) would generate a new function linear in certain parameters of the set '*S*' i.e.

$$S = S_1 \oplus S_2 \quad (7)$$

therefore applying *'h'* to the objective function,

$$h(O) = h \circ f(i, S) \quad (8)$$

where *'h'* is linear in S_2 parameters.

Assuming a set of values for the parameter set S_2.,we can generate P number of linear equations in S_2 parameters. Representing the same in a matrix we get,

$$A\theta = y \tag{9}$$

and by the Least Squares Estimator the solution to the above equation can be given as,

$$\theta = \left(A^T A\right)^{-1} A^T y \tag{10}$$

The above described procedure is implemented within forward pass of the network. Once all of the S_2 parameters are identified, then the backward pass is initiated. The errors are calculated and the gradient vector is determined. At the end of all training pairs, in the backward pass, the parameters in S_1 are updated by steepest descent method.

Applying a similar procedure to the Rules (1 & 2) described earlier, we would get,

$$\begin{aligned} f = (\overline{w_1}x)p \; + (\overline{w_1}y)q \; + (\overline{w_1})r \; + \\ (\overline{w_2}x)s \; + (\overline{w_2}y)t \; + (\overline{w_2})u \end{aligned} \tag{11}$$

$\overline{w_1}$ and $\overline{w_2}$ are the premise parameters and p, q, r, s, t, and u are the consequents.

Data sets used: Following is a brief outline of the data sets which have been used, and a little bit more about them individually.

a) Breast Cancer:

Source: Orange Data Mining [10]

Attributes (9 in all): recurrence, age, menopause, Tumor Size, inv nodes, node caps, Deg Malig, breast and breast quad.

Missing Values: No

Data Objects: 190 (one per patient)

Decision Classes: 2

b) Lung Cancer:

Source: Orange Data Mining [10]

Attributes (56 in all): Attributes 1 to 56.

Missing Values: Yes

Data Objects: 32 (one per patient)

Decision Classes: 2

Data Reliability: Though emphasis was laid on the conceptual implementation in this work, an effort has been made to use reliable data. The data is a common testing data available for cost free downloads from various severs. The UCI data repository and the Orange Data Mining Repository are just a couple examples. This data is widely used by AI researchers in order to establish a performance scale for their respective techniques.

IV. Design scheme

Structure: Figure-1 is the depiction of the connections between the various components of the system that has been developed.

Knowledge Representation: The data sets have been represented in the form of a table, with each of the row representing an event, object, case or a patient (applicable). Every column represents an attribute, property of the object or a characteristic for every object. This table we call the '*information system'* [6]. This form of knowledge representation that has been employed for the entire work.

The same can be represented as, A = (U, A), where U is the non – empty finite set of objects called the Universe and A is a non – empty finite set of attributes [6].

Once processed to discover relationships and dependencies among the involved parameters, the same knowledge is stored in a set of *"if... then"* rules. This applies to both of the Rough models (breast cancer as well as Lung Cancer). And in the case of the neural network (developed from ANFIS) stores knowledge in a *black box* sort of a structure, where in the knowledge is distributed among the various components of the network, however, unlike in other techniques, this knowledge may not be accessed easily. As far as the computation part of the work goes, Matlab is a scientific tool and is based on matrix manipulations.

Techniques and Algorithms: Following is the list of various techniques and algorithms used along with a brief description of the rationale behind using them in particular.

(a) Neuro Fuzzy systems: Neural networks exhibit lack of interpretability and Fuzzy systems lack the capability of effective learning. Thus Neuro Fuzzy systems present the learning capability of the neural networks and the fuzzy interpretation skills both in one tool. Moreover with respect to dynamic systems, neural networks provide the requisite skills for knowledge acquisition through learning and the fuzzy systems top it up with their ability to automatically approximate the knowledge bases for non – deterministic events. Due to the presence of these characteristics belonging to both the techniques, Neuro Fuzzy systems are widely used in machine learning applications. ANFIS is one of the kinds.

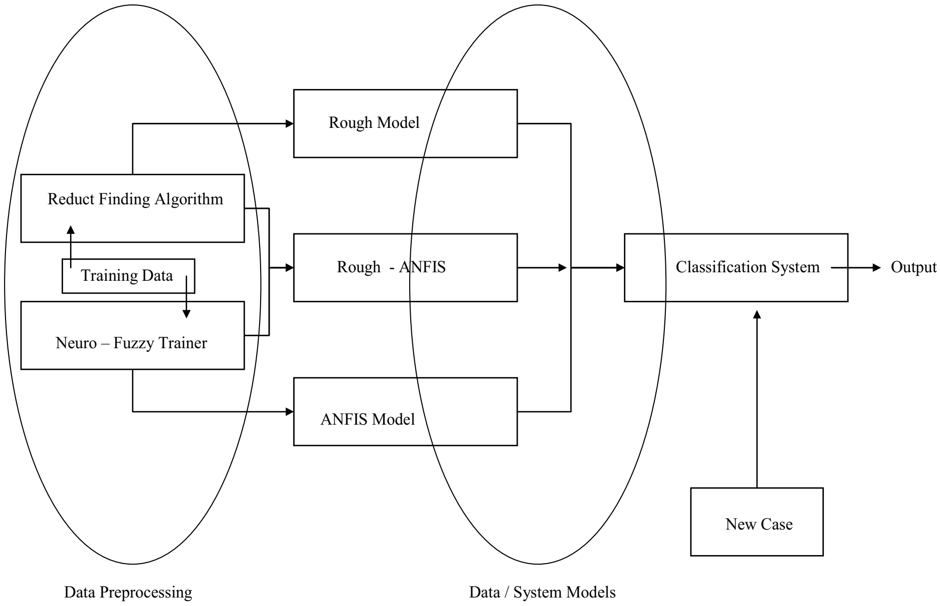

Fig. 1: Design Structure for the Classification system

(b) Rough Sets framework was chosen predominantly for its complimentary behavior when amalgamated with other approaches. Rough Sets Theory (RST) is well equipped to deal effectively with *imprecise, noisy and missing information* [11]. Unlike in other approaches, RST effectively sets the accuracy and precision value as per the requirement of the user for various classificatory processes. Further the concept of indiscernibility relations coupled with the concept of Reducts provides *discernibility – preserving elimination of irrelevant information* [11]. Also the issues arising due to multi and partial memberships of the objects in various sets have been reasonably addressed using the RST. And similar to the ES answering module, Rough Set models can be made capable of providing a description of analysis which led to the final decision. Fuzzy set theory and Rough Set theory are complimentary and not competitive. Amalgamation of the said techniques or theories would result in constructive determination since each of them refers to different aspects of imprecision i.e. Fuzzy set theory represents imprecision in the form of a partial membership whereas Rough sets accommodate the imprecision in the form of indiscernibility relations and the set upper and lower approximations.

(c) Algorithms: the quickReduct and Greedy Algorithms have been implemented to determine the optimal reducts for the classification system. The choice was made based on the transparency of the said algorithms and their efficiency.

Programming Style: Matlab has been used for this implementation. The only drawback with Matlab (with reference to this work) is that it does not support Dynamic matrices. This has made the programming aspect a bit more laborious and logically redundant.

V. System Structure Details

Rough Sets Based Model Modus Operandi:

i. Discretization of any continuous attributes while preserving quality of data,

ii. Consistency check to determine the quality of original data,

iii. Evaluation of attributes as per their inter dependencies,

iv. Determining optimal attribute reducts while preserving the quality of data or keeping in accordance with the user requirements,

v. Elimination of the redundant and less important attributes to prune the data set,

vi. And finally Rule Induction.

Discretization: The process of Discretization or quantization involves transforming the given set of continuous attribute values into discrete values. This usually involves breaking the continuous numeric scale down into various intervals or going further and allotting a numeric discrete value to each of the intervals. This process is better performed by consulting with the domain expert or on the contrary numerous Discretization techniques exist which implement global Discretization and the performance is optimized by ensuring that the quality of the original data is not lost.

A naïve global Discretization algorithm has been implemented but has not been integrated into the main system as it was not required. The algorithm optimizes the process of Discretization by initially evaluating the total number of discernible pairs in the data (based on the decision classes) and then gradually the search path yielding maximization of the discernible pairs is adopted. Once after establishing a cut for an attribute, the same procedure is repeated for all other attributes while taking into account all previous incisions on the value scale.

Consistency and Dependency of Attributes: Consistency in a data set implies nothing but a situation where the set (or subset) of conditional attributes of two cases match but they belong to different decision classes. Consistency in the data reflects directly upon the quality of the data and in case of the reduct, unaltered consistency measure is ideal but nevertheless any reduct within the permitted consistency threshold is acceptable.

Dependency of Attributes: Determining the dependencies between the attributes is an important aspect of KDD. We are particularly interested in determining if any subsets of attribute(s) exist, which are functionally and completely dependant on another subset of attribute(s). Given such a situation where C depends on D, the degree of dependency is defined as a ratio of the cardinality of the positive region defined by the indiscernibility relation of D as per the attribute set C and the cardinality of the universe / complete dataset. In the same lines, the degree of dependency of an attribute subset can be determined by initially determining the consistency level of the data as per the attribute subset in question and then determining the ratio of the said attribute set consistency and the overall data consistency. Thus in simpler words, the positive region defined by an attribute (sub)set is directly proportional to the degree of dependency of that (sub)set.

Data Reduction / Reducts: Data is plagued with redundancy i.e. often some or the other attributes are superfluous. Reducts help discard functionally redundant information. Hence, a reduct can be defined as a minimal set of attributes that preserves the indiscernibility relation and the quality of the approximations as that in the original set computed by taking in consideration the complete set of attributes. In layperson's words, reducts help in modular construction of relatively smaller and simpler models. This helps improve the attention given to the decision-making process.

A reduct is a subset of the overall attribute set, in a way that the quality of information is unaltered or is within the threshold value set by the user requirements. And an optimal reduct is primarily in lines of the above definition but with an additional criterion that the number of attributes in the reduct shall be minimal. Two reduct finding algorithms have been used, viz,

quickReduct Algorithm; Since determining and evaluating all possible reducts of a given set of attributes is humongous and unnecessary, the quickReduct works on the principles of the Best First Heuristic search, i.e. rather than traversing the entire search tree, it employs certain heuristic evaluation criterion. Each node of the tree (search space) represents the addition of one conditional attribute to the reduct from the previous node and the root obviously is an empty reduct set. The solution path (reduct configuration) is determined incrementally using the criterion that the *next attribute chosen to be added to the reduct is the attribute that adds the most to the reduct's dependency,* [12] i.e. the most significant attribute should be chosen. The termination criterion is a simple threshold value for the degree of dependency, meaning that the algorithm would stop iterating the first time it satisfies the termination criterion.

Algorithm:

Input: all attributes; and corresponding decision.

Output: attribute reduct.

(1) R = {empty}

(2) For every element of the attribute set determine the degree of dependency

(3) Add the attribute with the maximum degree of dependency to the reduct.

(4) Terminate loop when degree of dependency > threshold value

Discernibility Matrix: A data set of 'n' cases generates a discernibility matrix of size 'n x n'. An element d_m(i,j) of the discernibility matrix is the set of attributes which discern the objects i and j (if at all they are discernible).

Greedy Algorithm; Once after calculating the discernibility matrix, the attribute which appears in most number of the discern sets is chosen since the corresponding attribute can be used to discern between the maximum number of objects in the data set. This attributes' frequency is now used as heuristic evaluation for all future searches. Now all the elements where the chosen attribute occurs are removed since we don't need any further attributes to discern among them apart from the one that has been chosen. The same procedure is applied to the remaining discern sets and recursively the best attribute is added at every step. This process is repeated until a termination criterion is satisfied or until the threshold can be satisfied.

Algorithm:

(1) Generate discernibility matrix M, calculate the frequencies of the attributes,

(2) Chose the best attribute,

(3) Remove all discern sets where the chosen attribute occurs,

(4) Determine the positive region (P) of the reduct thus far,

(5) if P >= Threshold, terminate the loop,

System Evaluation: The rough model is evaluated by ensuring that all the cases in the given data set are classified correctly. This was performed and it was observed that all of the cases were identified correctly. New Cases: Rough model refers the user to the Rough – ANFIS model when it encounters a new case.

Adaptive Neuro Fuzzy Inference System based Model: Matlab Fuzzy logic toolbox has been used to generate 3 ANFIS structures (2 Rough – ANFIS and 1 standalone ANFIS). The ANFIS functions provided within the toolbox have been extensively used for this part of the work. Though the work has been devised to highlight the advantages of Rough - ANFIS architecture and the Hybrid Learning scheme, in this part of the work, the standalone ANFIS model has been developed in order to provide the reader with convincing information and comparison of the various methods applied to the same problem in a similar fashion in order to bring about a clear understanding of the virtues of the involved techniques. The comprehensive details of the models generated, trained, validated and implemented have been described in the following sections.

Modus Operandi: To implement an ANFIS architecture, the first thing that one needs is to have relevant data of inputs and outputs of the system (to be modeled), assuming that the identification of inputs whose influence on the output shall be modeled has already been done. This data is conveniently broken down into training data and a relatively smaller and distinct set of checking data.

Once reliable training data has been obtained, the next step is to choose appropriate number and kind of the membership functions that shall be used to model the system rules within the adaptive neuro – fuzzy architecture. The selection of the inputs to be modeled, the membership functions and the number of functions per input together constitute the creation of a fuzzy inference system generation.

What follows next is more the on hand job i.e. the training data along with the checking data is loaded and the FIS is generated. Now after choosing, one of the learning mechanisms (i.e. hybrid learning or the back propagation) and the number of training epochs, the FIS is allowed to train itself and meanwhile learn from the iterations. Once the training is complete and the error established, the checking error and training error are counterchecked with each other for harmony. In both cases, that the checking error and training error are not in a good understanding with each other or that the training error is in excess of the allowable error tolerance, the entire process should be repeated. Else, the ANFIS architecture is saved as *.fis file and further on, after it has been trained, tested and saved, it can be accessed more like a fis file rather than in an ANFIS format.

Membership Functions; The membership functions chosen initially have a role to play in the initial convergent behavior of the model. Though no hard and fast rules exist to help a user chose the membership functions, but the rule of thumb is that, an initial test run be made to identify the effectiveness of the various options. But in this work, due to the sensitive nature of the data and the involved parameters, I have adopted the statistical approach, wherein the input – output relationship is evaluated for trends in the behavior and based upon these findings the membership functions were chosen. Due to the discrete nature of the data under consideration, Trapezoidal membership functions have been chosen for all of the systems. i.e a graph was plotted for the data and accordingly Trapezoidal membership functions have been chosen.

The number of membership functions for each of the inputs again was based upon the input data profile. Depending upon how many discreet groups, the data could be categorized into; the number of membership functions was chosen. Such an observation was also supported with the preliminary knowledge about the system. But it should be noted that, despite of the initial selections for the number of membership functions and the type of the membership functions, the reported results are variant in nature due to the learning process and the adaptation of the initial parameters. Further the number of membership functions was altered after reviewing the initial test runs for the model.

Parameters: All of the parameters of the membership functions depicted in the earlier page, may be obtained from *.fis structure. This information has not been provided due to its uselessness in the current context of the work.

International Journal of Information, Control and Computer Sciences
ISSN: 2517-9942
Vol:2, No:3, 2008

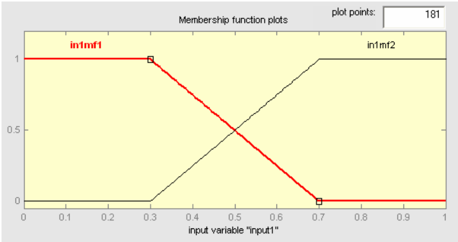

Fig. 2 Sample membership function profile for Breast Cancer data

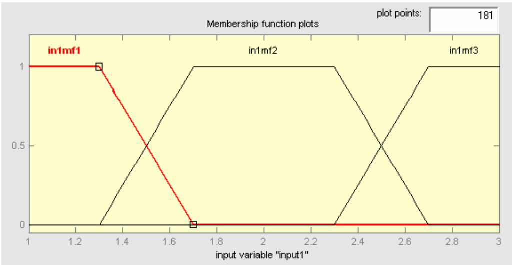

Fig. 3 Sample Membership Function Profile for Lung Cancer data

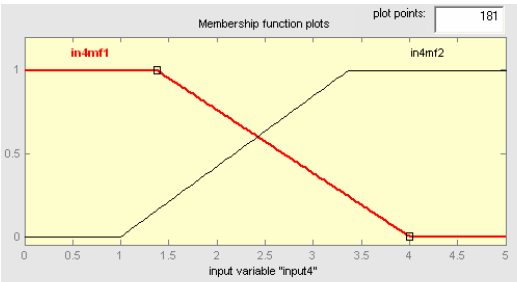

Fig. 4 Sample Membership Function Profile for Rule Map Reduced data

Training using Hybrid Learning: Most of the research and analysis is directed towards the effectiveness of a certain methodology, the accuracy, the internal architecture etc but rarely does one get to read about the ease with which a certain method can be constructed and tested as opposed to the more efficient but highly cumbersome to construct and test. In this section, we would like to bring the readers' attention towards the marvelously high speed with which the ANFIS based Hybrid Learning Algorithm can converge towards an optimal solution. It should be noted that not only is the Hybrid learning ANFIS more accurate than its counterparts but is also extremely fast and an easy to implement methodology.

VI. Results

For all of the models for both the classification systems, the data sets with their respective decisions have been verified. Further, following are some of the relevant figures. These can help the reader in reaching a conclusion about the involved techniques.

TABLE I BREAST CANCER

Attribute	Rough	ANFIS	Rough – ANFIS
Number of Rules	152	512	64
Error Rate	0	0.175	0.08
Attributes in use	6 out of 9	9 out of 9	6 out of 9
Reduct Cardinality	1 (100%)	1(100)%	- NA -

TABLE II LUNG CANCER

Attribute	Rough	Rough – ANFIS
Number of Rules	32	27
Error Rate	0	0.1 (per unity)
Attributes in use	3 out of 56	3 out of 56
Reduct Cardinality	1 (100%)	- NA -

VII. Conclusions

From the above classification results, it is evident that that proposed system effectively extracts knowledge from a given database with negligible or no human intervention. The modular structure not only performs the inclusive operations in a systematic fashion but also process the information contained within the database to give shape of appropriate domain knowledge. Considering the complicated aspect of knowledge modeling (from an expert and by a Knowledge Engineer), the presented model can to a certain extent completely eliminate the need for both knowledge extraction and articulation. The proposed structure is applicable to all cases where data collection is feasible. Pending extensive testing, it can be concluded that the proposed system is capable of proactive knowledge engineering. Further, with the reference to the hybridization of methodologies presented here, the proposed system brings about an understanding that the brittle boundaries encountered in Rough Sets can be tackled with the help of fuzzy smoothening. Same is the case with the Rough set constraint of working only with discreet value sets; Fuzzification of the parameters greatly reduces this inhibition. And as expected (from the earlier theory), Rough Sets prove to perform in a complimentary fashion to the Neuro – Fuzzy Systems. The Rough Sets Framework has a superior property that it comes with a strong mathematical and formal model of the knowledge and this coupled with its capacity to effectively deal with imprecise information makes it a forerunner to major classification techniques that are available today and among the ones that have been tested here. It has been observed from the experiments that the

Rough Set approach helps in bringing forward the hidden knowledge from intricate databases or knowledge systems, which otherwise would go unnoticed. It can also be stated that this is the first time when a hybrid model imbibing concepts of Rough Sets, Fuzzy Sets and Neural Networks has been developed for the purpose of extracting knowledge.

REFERENCES

[1] Abidi. S. S. R, Cheah. Y-N, Curran J.; "A Knowledge Creation Info-Structure to Acquire and Crystallize the Tacit Knowledge of Health-Care Experts"; IEEE Trans. on Information Technology in Biomedicine, v.9(2), pp. 193-204; 2005.
[2] Ohsuga. S; "Knowledge Discovery as Translation"; In Lin. T. Y et. al. [3]; 2005.
[3] Lin T. Y, Ohsuga. S, Liau. C-J, Hu. X, Tsumoto. S; Foundations of Data Mining and Knowledge Discovery; Springer-Verlag; Berlin; 2005.
[4] Hamilton-Wright. A, Stashuk. D. W; "Transparent Decision Support Using Statistical Reasoning and Fuzzy Inference"; IEEE Trans. on Knowledge and Data Engineering, v.18(8) , pp.1125-1137; 2006.
[5] Cao. L, "Domain-Driven, Actionable Knowledge Discovery"; IEEE Intelligent Systems, v.22(4), pp.78-88; 2007.
[6] Pawlak. Z; "Some Issues on Rough Sets"; In Peters-Skowron [7]; 2004.
[7] Peters. J. F, Skowron. A; Transaction on Rough Sets-I; Springer; Berlin; 2004.
[8] Jang. J. S. R; "ANFIS: Adaptive Network Based Fuzzy Inference Systems"; IEEE Transactions on Systems, Man and Cybernetics; May 1993.
[9] Jang. J. S. R, Sun. C. T, Mizutani. E; Neuro – Fuzzy and Soft Computing: A Computational Approach to Learning and Machine Intelligence; Prentice Hall; 1997.
[10] Demsar J, Zupan B; "Orange: From Experimental Machine Learning to Interactive Data Mining"; White Paper (www.ailab.si/orange), Faculty of Computer and Information Science, University of Ljubljana; 2004.
[11] Kudo. Y, Murai. T; "Missing Value Semantics and Absent Value Semantics for Incomplete Information in Object-Oriented Rough Set Models"; In Bello et. al. [13]; 2008.
[12] Jaganathan. P, Thangavel. K, Pethalakshmi. A, Karnan. M; "Classification Rule Discovery with Ant Colony Optimization and Improved Quick Reduct Algorithm"; Intl. J. of Computer Science, v.33(1), pp.1-6; 2007.
[13] Bello. R, Falcon. R, Pedrycz. W, Kacprzyk. J; Granular Computing: At the Junction of Rough Sets and Fuzzy Sets; Springer; London; 2008