A Multi-Agent based Medical System with Several Learning and Reasoning Capabilities

Fernando Manzanares

Instituto Tecnologico de Cd. Madero-Universidad
Central de las Villas
fer_manzanares@yahoo.com

Abstract. I present a Multi-agent System that represents a Medical community where each Doctor may have a different specialty and may do his/her work with specific techniques. In this sense, I propose the use of several Machine Learning techniques as a form to represent the use of different techniques to learn to diagnose diseases and the use of several data sources to produce knowledge, permitting that each data source can be related with some specific medical specialty. Also I propose the separation of the processes of learning and reasoning to increase the use of Machine Learning techniques whose implementations requires great amount of computer resources; for this, the knowledge extraction from data is done by software agents that are executed in high performance computers while the reasoning processes that exploit the knowledge discovered are carried out by software agents that are executed in average scale computers.

1 Introduction

In each community we can find doctors with perhaps different specialties: ophthalmology, bacteriology, cardiology, among many others. Also, each doctor could have attended his studies in a different University and have made his specialization in a different Hospital than the others. As well, each one can set in an independent way the amount of its fees. Nevertheless, the great majority of works related to processes of knowledge extraction from data do not follow this behavior since they are focused in the implementation of a single technique and/or in the creation of a model associated with a particular problem domain. In this work an analogy to the situations that occur in the real life is created through the construction of a multi-agent based medical system, where the agents have multiple capabilities of learning and reasoning. In their construction several machine learning techniques are used such as Artificial Neural Networks, Rough Set Theory, Bayesian Learning and ID3 algorithm, as well as reasoning processes according to the models that the previous techniques build.

The purpose is simulating the different approaches used by each doctor to carry out its work. Also several decision systems related to different medical areas are used in

© A. Gelbukh, S. Torres, I. López (Eds.)
Advances in Computer Science and Engineering
Research in Computing Science 19, 2006, pp. 211-224

order to create different artificial specialists. As development platform was utilized Java programming language and JADE framework.

2 Machine Learning

2.1 Problems Related to Machine Learning

During the last decades an enormous increment in the use of sensors as well as in the development of data storage devices has been presented. This has produced an enormous amount of data, available to be analyzed by specialists to be able to understand and to control in a better way the underlying problem domains. On the other hand the increment in the power of available computation for these specialists has done possible the use of more complex models, causing among others things an increase in the investigation and development in diverse fields of the Artificial Intelligence [1].

Within the techniques used in Artificial Intelligence for problem domains understanding the machine learning is found, with an extensive use of supervised inductive learning, which is used to acquire knowledge from examples previously classified.

Many techniques of supervised inductive learning have been used, within that we can mention: Artificial Neural Networks, Rough Set Theory, Bayesian Learning, Genetic Algorithms, the family of algorithms derived from ID3, among others. Each one of them has its strengths and its weaknesses. For example, it is well known that decision trees produced by ID3 are highly understandable and expressive, nevertheless is also known the incapacity of ID3 to deal with unstable, uncertain or incomplete data. On the other hand the Artificial Neural Networks are excellent universal approximators with capacity to process incomplete or noise data, but the models generated by them as a form of knowledge representation in numerical matrices form makes no sense for a human being.

Another situation that we must consider as advantages and disadvantages of these techniques consists of the type of inputs that are able to process. For example, techniques such as ID3 or Bayesian Learning are not adequate to process images unlike the Artificial Neural Networks.

Another disadvantage of the majority of the Machine Learning techniques is the great amount of computational resources that they require for his execution. This diminishes the number of potential users that can be benefited from their use.

It is by that a multi-agent system in which some agents takes charge of the process of supervised inductive learning and other agents to use the models created by the first in order of putting in use the acquired knowledge. Creating with this an artificial medical community.

2.2 Supervised Inductive Learning

Many induction problems can be described as follows [2]. One begins with a training set of preclassified examples, where each example (also called observation or case) is described by a vector of values of characteristics or attributes, and the objective is to form a description that can be used to classify with high precision examples non previously seen. In a formal way we say that an example is a pair *(x, f(x))*, where *x* is the input and *f(x)* is the output of the function applied to *x*. The objective of the inductive inference is, given a set of examples of *f*, to produce a function *h* that approximates *f*. Normally *x* is a vector of attributes each one with a particular domain and function *f* is the valuation done by a human expert of the values of *x*. For example *x* can be a set of symptoms, vital signs and results of biochemical analysis of human patients and the output *f(x)* can be the diagnosis done by a doctor.

According to [9] the construction of a procedure of classification from a data set for which the classes are known has also been called in an indistinct way as pattern recognition, discrimination, discovery of knowledge or supervised learning. Being distinguished of the non supervised learning or grouping in which the classes are inferred from the data.

2.2.1 Decision Systems

Independently of the used technique to carry out the induction, the sets of examples that are used can be presented in a standard form of a Decision Table, which is an implementation of a Decision System.

Given an Information System, defined like a pair A = (U, A), where U is a non empty finite set of objects called universe of examples (objects, entities, situations or states, etc.) and A ={A1, A2,...,An} is a non empty finite set of attributes, such that the elements of U are described using the attributes Ai. If to each element of U a new attribute d called decision is added, indicating the decision taken in that state or situation, then a Decision System (U, A∪{d}, where d∉A) is obtained [2]. The values of the decision attribute d are, as already was mentioned, the outcome done by a human expert. For example, the Ai□A attributes can be the symptoms or characteristics of the patients whom go to medical consultation in a particular clinic and the values of d can be the diagnosis done by the doctors of the clinic whom attended each patient.

3 Artificial Neural Networks

Artificial Neural Networks (ANNs) are known by diverse names, among them: connectionist models or parallel distributed processing models. Instead of executing a program sequentially as in a Von Neumann architecture, the ANN explores many hypothesis simultaneously using massively parallel networks of many elements of processing connected by weighted connections. The ANNs are inspired in the biological model of the human brain, without reaching to duplicate it. The main purpose of all the Biological Neural Systems is the centralized control of several biological functions, some of them responsible for supplying of energy, therefore the neuronal system is connected with the metabolism, the cardiovascular control and the breathing. In the human beings, as well as in the majority of the superior animals, the greater capacity of the neurological system is related to the behavior, this is, the control of the state of the organism with respect to its environment [4, 5].

In the cerebral nervous system the central element is a cell called neuron. An important difference of these cells with respect to the rest of the alive cells is its capacity to communicate. In general terms a neuron receives input signals, combines and integrates them and emits a output signal. In fact a single neuron generally does not do anything, performance and results are determined by the cooperative work of several of these neurons.

The ANNs normally consist of Artificial Neurons as the one that is shown in figure 1. The Artificial Neuron is seen like a node connected with other by means of links that correspond to connections axon-synapse-dendrite, which are present in the biological neuron.

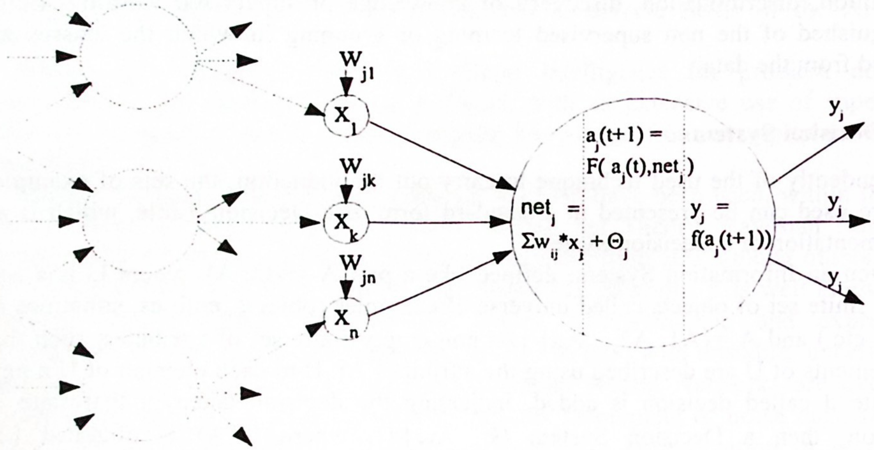

Fig. 1. Artificial Neuron

There is associated a weight with each connection. As in the case of the synapse in the biological neurons that weight determines the nature and intensity of the influence of one node on another one. In more specific way, the influence of one node on another one is the product of the signal of the input neurons by the weight of the connection that connects them with the node in which influences. For example, a large positive weight corresponds to a strong excitation, and a small negative weight corresponds to a weak inhibition. This interaction causes that in every single moment of time t all the neurons that compose the ANN be found in a certain state. In a simplified way, we can say that there are two possible states: rest and excitation, that we will call activation states. These activation values can be continuous or discreet. In addition, they can be limited or unlimited. If, for example, they are discreet binary, an active state would be indicated with a 1, and is characterized by the emission of an impulse by the neuron (action potential), while a passive state would be indicated for a 0 and would means that the neuron is in rest.

To generate the state of activation aj(t) of each node j, these combine the individual influences that receive in their input connections in a single global influence, by means of an activation function. A single activation function passes the weighted sum of the input values through a transfer function to determine the output of the node. In case of production of binary outputs, this can be 0 or 1, depending on if the weighted sum of inputs is down or above the threshold value utilized by the node's activation function.

The connections that link the neurons that form the ANN have an associated weight, which is where the knowledge acquired by the network is placed. We consider the case where a neuron j is connected in its inputs with N units. Let us denominate wji the weight on the connection between the neuron i and the neuron j. Also we denominate xi the output value that neuron i transmits to neuron j. As simplification we consider that the effect of each input signal is additive, in such way that the net input netj that receives a neuron is the sum of the product of each individual signal by the value of the synapse that connects both neurons:

$$\mathrm{net}_j = \sum_{i=1}^{N} \mathrm{w}_{ji} \cdot x_i - \Theta_i$$

This rule determines the general procedure to combine the input values of a unit with the weight of the connections that arrive at the same one and is known as propagation rule, where □j represents the threshold value of the neuron or a bias term on it. Also a called activation rule exists [7], the one that determines how combines the value of the weighted inputs netj and the present state of the neuron a(t – 1) to produce a new state of activation:

$$\mathrm{a}_j(\mathrm{t}) = \mathrm{F}(\mathrm{a}_j(\mathrm{t-1}), \mathrm{net}_j)$$

This function F produces a new state of activation in the neuron j from the state in the previous instant t-1 and the combination of the weighted inputs in the present instant t.

In most cases F is the identity function, reason by which the state of activation of a neuron will be the value netj of the same one. In this case, the parameter that is passed to the output function will be netj, directly, without being taken into account the previous activation value. In agreement with this the output yj of a neuron j will be according to the expression:

$$\mathrm{y}_j(\mathrm{t}) = \mathrm{f}(\mathrm{net}_j - \Theta_j) = \mathrm{f}(\sum_{i=1}^{N} \mathrm{w}_{ji} y_i(t-1))$$

The modification of the weights of a network can be carried out in diverse ways, grouped in two great classes: supervised learning and unsupervised learning. Within the main types of supervised learning we have:

a) Learning by correction of errors. It consists of adjusting the weights in function of the difference among the output values from the network and the expected values. One of the error correction rules more extensively used is the called Generalized Delta Rule or Gradient Descent Rule, base of Backpropagation algorithm:

$$\Delta w_{ij}(t+1) = \alpha(d_{pj} - y_{pj}) f'(net_j - \Theta_j) y_{pi}$$

where

Δw_{ij}: is the amount by witch to change the weight of the connection between neurons i and j

α: is the learning rate parameter
p: is the pth training example
d_j: is the desired output of neuron j
y_j: is the output of neuron j
Θ_j: is the threshold value of neuron j

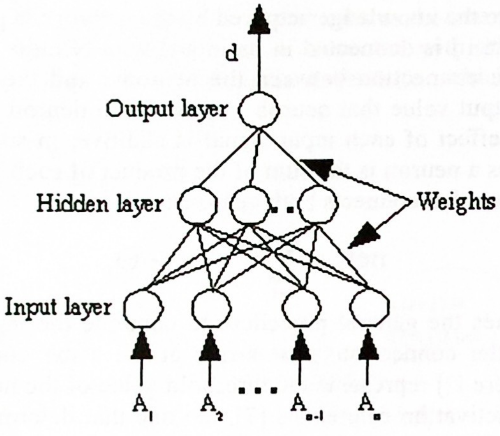

Fig. 2. A Multi-Layer Perceptron

b) Learning by reinforcement. It is a subtype of supervised learning, characterized by there are not complete examples of the desired behavior, but a supervisor exists that simply informs by means of a reinforcement if the network output is adjusted or not o the desired output and, in function of it, the weights used are adjusted using a probabilistic base.

c) Stochastic learning. This type of learning consists of making random changes to the weights values, evaluating its effect from the desired objective and based on probability distributions [8].

Attending its relation with the environment, there are four types of artificial neurons: input, output, hidden and composite. Input neurons are which receive from the external environment the information that the ANN will learn or will process. Output neurons are the units that are responsible to carry out the results of the processing done by the ANN in the input data. Hidden neurons do not have any contact with the external environment of the ANN. Composite neurons are input/output neurons.

We say that the neurons of the same type are grouped in a layer. Therefore, the ANNs could have one or more layers forming the network topology. Besides, related with the form in which the signals are transmitted over the network, there are feed-forward, feedback and recurrent connections.

The kind of ANN used in this work is the Multi-Layer Perceptron (MLP) using the backpropagation algorithm. This is a multi-layer network, with feed-forward connections, with continues inputs and outputs lessen in the interval [0, 1], and which uses the Generalized Delta Rule as learning rule. Here the MLPs used are restricted to only one hidden layer and one output. This is illustrated in figure 2.

3.1 Rough Set Theory

Rough Set Theory was introduced by Zdzislaw Pawlak in 1982 [10] as a formal mathematical theory for the modeling of the knowledge about a domain of interest in terms of a collection of equivalence relations. Its main area of application is in the acquisition, analysis and optimization of computer processable models from data. The models can represent functional, partial functional ant probabilistic relations existing in data through the extended approaches of the Rough Sets.

This theory often has been proved to be an excellent mathematical tool for the analysis of vague descriptions of objects, particularly when the vagueness refers to inconsistencies or ambiguities due to the level of information granularity [12]. Its importance in the Artificial Intelligence and the Cognitive Sciences is related to the areas of machine learning, knowledge acquisition, decision analysis, knowledge discovery from databases, expert systems, decision support systems, inductive reasoning and pattern recognition [5].

The starting point of the philosophy of the rough sets is the assumption that with each object of interest, in a problem domain, exists some associated information. For example, if the objects are patients that suffer a particular disease, the symptoms of the patients form the information about the patients [10].

In [11] it is mentioned that the problems that can be resolved with the rough set theory are:

1. characterization of a set of objects in terms of attributes values
2. total or partial dependencies, among attributes
3. reduction of attributes
4. significance of attributes
5. decision rules generation

One of the main advantages of this theory is that it does not require to having any preliminary or additional information about the data, just as probability distributions in statistics, assignment of basic probabilities in the Dempster-Shafer theory or membership functions in the fuzzy set theory.

From the contained information in the decision system, is desired to discover rules of the form X □ Y (C), where X is the condition or antecedent of the rule built from the attributes of the data set A, Y is one of the values that can take the decision attribute d, and C is the certainty of the rule.

From an equivalence relation B and the use of upper and lower approximations (rough sets) such rules can be constructed. Any subset of A (B □ A), can be used as equivalence relation, although the use of the attributes of A with greater relevance or importance in the application domain are recommended. The idea is to construct the Yi sets, in which are all the elements of U which have same value yi of the decision attribute. To these sets Yi their upper and lower approximations are determined and from them the rules are generated. The algorithm of construction of the rules is the following one (LRRS, Learning Rules using Rough Set) :

1. Build the decision system $(U, A \cup \{d\})$.
2. Define the subset $B \subseteq A$ of attributes that are considered relevant.
3. Build the sets $Y_j \subseteq U$, such that in Y_j are all the elements of U that have y_i as value in the decision attribute.
4. Build the equivalence classes X_i from the relation B.
5. Build the lower and upper approximations for each subset Y_j:

$$B_\bullet(Y_j)= \{x \in U / B(x) \subseteq Y_j\}$$
$$B^*(Y_j)= \{x \in U / B(x) \cap Y_j \neq \}\phi$$

6. Build the limit region of each subset Y_j for the equivalence relation of B:

$$BNB\ (Yj) = B^*\ (Yj) - B^*\ (Yj)$$

7. Build rules of certainty 1.

```
For all $X_i$ do
    For all $Y_i$ do
            If $X_i \subseteq B_\bullet(Y_j)$ then generate the rule: $X_i \Rightarrow Y_j$ $(1)$.
```

8. Build rules of certainty smaller than 1.

```
For all $X_i$ do
    For all $Y_i$ do
    If $X_i \subseteq BN_B(Y_j)$ then generate the rule: $X_i \Rightarrow Y_j$ $(C)$,
    where $C=/X_i \cap Y_j//X_i$
```

The previous algorithm presents a simple form to build rules. Nevertheless, from the use of rough sets more complex procedures to knowledge discovery have been developed.

3.2 Bayesian Learning

A Bayesian Classifier is trained by the estimation of the conditional probability distribution of each attribute, producing the label of the class, from the database. A case is classified from its set of attributes values, using the Bayes' rule. The case then is placed in the class with the greater probability. The underlying assumption that simplifies the Bayesian classifiers is that the classes are exhaustive and mutually exclusive and that the attributes are conditionally independent once the class is known. A Bayesian classifier is defined by a set C of clàsses and by a set A of attributes. We denote a generic class as c_j, and a generic attribute as A_i. The set C of classes can be tried as a stochastic variable taking one of the values c_i with a probability distribution that represents a unknown state of the world. The decision system is used to determine the probabilities $P\ (c_j)$ and $P\ (A_i|c_j)$ for each attribute A_i. These probabilities are determined counting the number of instances. All the attributes values depend on their class only, and connections between attributes are not permitted[3]. Figure 3 shows a Bayesian Network that is used like a Bayesian classifier.

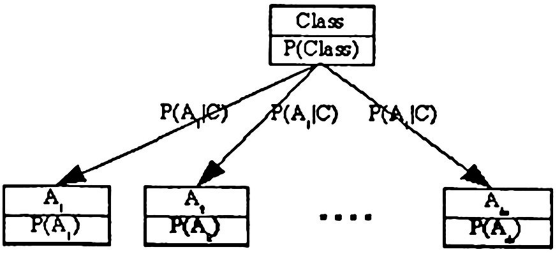

Fig. 3. Bayesian network.

Related with each link in the Network exists a Conditional Probability Table such that is shown in table 1. Suppose that you observe a new case with A1 = v2, A2 = v2,..., Ak = vk. We use the Bayes' rule to determine the posterior probability of the class cj of the new case, conditioned in the attributes values as follows:

$$P(c_j \mid A_1 = v_1, A_2 = v_2, \ldots, A_k = v_k) = P(A_1 = v_1, A_2 = v_2, \ldots, A_k = v_k \mid c_j) P(c_j) / P(A_1 = v_1, A_2 = v_2, \ldots, A_k = v_k)$$

Using the independence assumption this is simplified to:

$$P(c_j \mid A_1 = v_1, A_2 = v_2, \ldots, A_k = v_k) = P(A_1 = v_1 \mid c_j) * \ldots * P(A_k = v_k \mid c_j) / P(A_1 = v_1, A_2 = v_2, \ldots, A_k = v_k)$$

The values P(A1 = v1 | cj) are obtained from the conditional probability tables. The denominator P(A1 = v2, A2 = v2, ..., Ak = vk)
is a normalization factor to force the addition of probabilities is one.

Table 1. Conditional Probability Table

$P(A_I \vert C)$	c_1	...	c_n
$a_{1,1}$	$P(A = a_{1,1} \vert c_1)$	...	$P(A = a_{1,1} \vert c_n)$
$a_{1,m}$	$P(A = a_{1,m} \vert c_1)$		$P(A = a_{1,m} \vert c_1)$

3.3 ID3 Algorithm

This algorithm produces knowledge in a decision tree representation. Learning of trees is a method to approximate functions of discrete values. A decision tree classifies the instances ordering them top-down form root to leaves. Each internal node of the tree specifies a test from an attribute and leaves are the classes in which instances are classified. Each link of an internal node corresponds to some possible value of the attribute tested in that node. A decision tree represents a disjunction of conjunctions on the attributes values. Each branch from the root to a leaf node corresponds to an attribute conjunction, and the tree is itself a disjunction of that conjunctions. This is illustrated in figure 4.

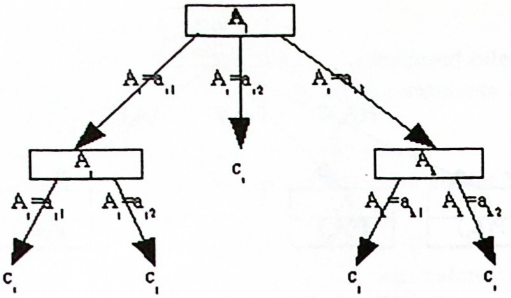

Fig. 4. A decision tree

```
The ID3 is a recursive algorithm:
ID3(DS:Decision System) : Decision tree
        1. Build a root node.
        2. If all examples have the same decision $d = d_i$ then
        - Label root node with $d_i$
        - Return root node
        3. If attribute list A is empty then
        - Label root node with most common value of d
        - Return root node
        4. bestNode ← attribute with minimum entropy
        5. Label root node with bestNode
        6. For each value $v_i$ of bestNode do
        6.1 add new descendant branch to bestNode related with test bestNode = $v_i$
        6.2 Let $\text{examples}_{vi} = \{X \mid X \in U \wedge \text{bestNode}(X) = v_i\}$
        If $\text{examples}_{vi}$ is empty then
        Build a leaf node
        Label the leaf node with the most common value of d in U
        Link the descendant branch with the leaf node
        else Build a new node = ID3({$\text{examples}_{vi}$,
        A-bestNode, {d}})
        Link the descendant branch with new node
        7. Return root node
```

4 Architecture of the multi-agent based medical system

4.1 System Architecture

The Multi-agent System proposed consists of the development, in first place, of at least four agents with inductive machine learning capacity, based on each one of the techniques mentioned in the section 2 of this document, with the additional capacity to deliver the models created to other agents who therefore requested it. These agents should be executed in a high scale computer that includes a Java virtual machine based on the J2SE and the JADE framework.

In second place the development of at least other four agents with capacity of reasoning is proposed. This type of agents should interact with the learning agents to request the models that can be employed in the solution of specific problems, for which these must know what models are available for be able to reason with each one of them. Besides these agents can be consulted for some another agent that have not capacity of reasoning and whose has the roll of avatar of a human being. Reasoning agents also can be avatars, and must have graphical user interfaces. Reasoning agents can be executed in average scale computers that includes the J2SE Java Virtual Machine and the JADE framework.

The third type of agent that is proposed only has the capacity to consult models available, verify if those are adequate to particular problem, and then acquire the data related with the problem instance to be sent to a reasoning agent for its solution. It is for this that them receive the User Interface Agent or Avatar Agent name. These should be capable of interacting as with learning agents to know the available models, as with reasoning agents to request them reasoning services. For their execution low scale computers can be used, such as PDAs or Smart Phones, which include the J2SE or J2ME Java Virtual Machine and the JADE framework. Figure 5 shows this plan.

4.2 System Operation

The process initiates when learning agents carry out the process of induction on a subset of learning of a decision system related to cases of some medical specialty. Once it extracted the knowledge this is validated on the base of a subset of test of the same decision system. A human expert must provides to each learning agent the goal to obtain a minimum percentage of effectiveness. The minimum effectiveness can be established as a set:

effectiveness = (minimum effectiveness rate, minimum false positives rate, minimum false negatives rate)

Once the agent has reached the goal, should publish in the yellow pages service the technique used, the price of its services and the name of the specialty related to the decision system from the knowledge was extracted. The price is used as comparison base for selection in the event that two or more agents have produced models related to the same specialty. In this work, the price is calculated in terms of the effectiveness, for it the same expert that established the minimum effectiveness should establish how affect false negatives and false positives in the associated specialty, therefore the price published is: Price = AccRate - w1 * PosRate - w2 * NegRate

where

AccRate : accuracy rate,
PosRate : false positives rate,
NegRate : false negatives rate,
w_1 *: weight associated with PosRate,*
w_2 *: weight associated with NegRate*

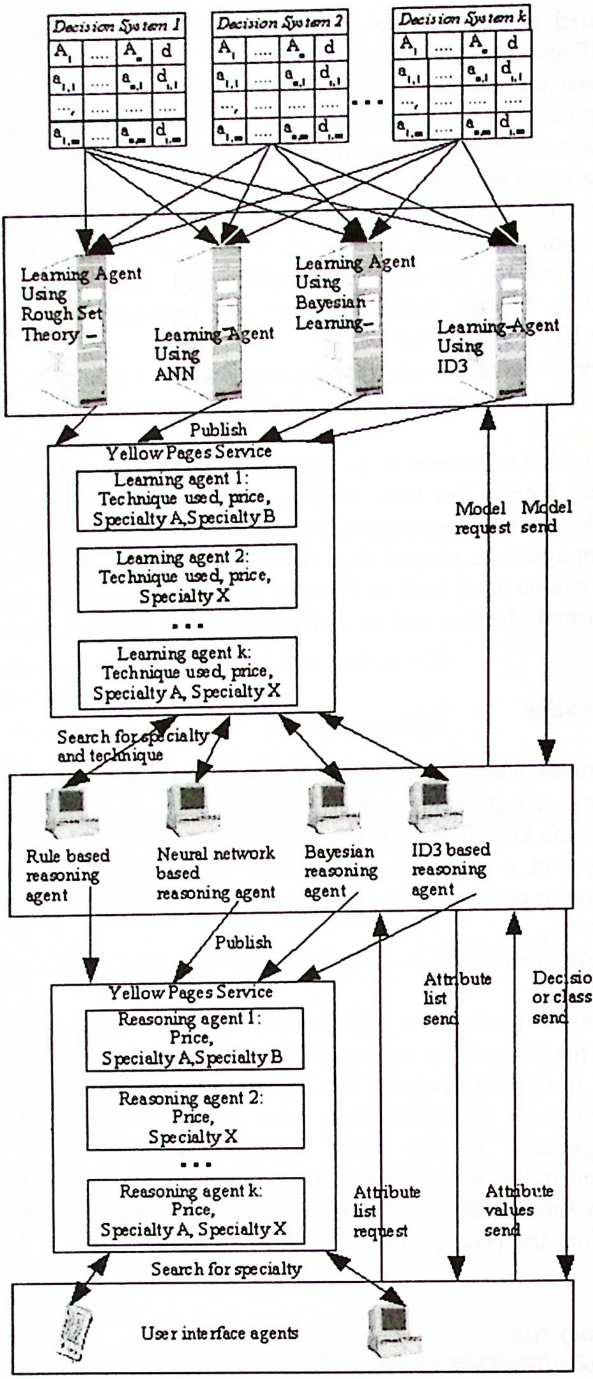
Decision System 1
Decision System 2
Decision System k
A1 .... An d
a1,1 .... an,1 d1,1
...,
a1,m .... an,m d1,m
...
Learning Agent Using Rough Set Theory
Learning Agent Using ANN
Learning Agent Using Bayesian Learning
Learning Agent Using ID3
Publish
Yellow Pages Service
Learning agent 1: Technique used, price, Specialty A, Specialty B
Learning agent 2: Technique used, price, Specialty X
...
Learning agent k: Technique used, price, Specialty A, Specialty X
Model request
Model send
Search for specialty and technique
Rule based reasoning agent
Neural network based reasoning agent
Bayesian reasoning agent
ID3 based reasoning agent
Publish
Yellow Pages Service
Reasoning agent 1: Price, Specialty A, Specialty B
Reasoning agent 2: Price, Specialty X
...
Reasoning agent k: Price, Specialty A, Specialty X
Attribute list send
Decision or class send
Attribute list request
Attribute values send
Search for specialty
User interface agents

Fig. 5. System architecture

As each technique produces a particular knowledge representation and since the operation of knowledge extraction will do it another agent, then the learning agent must publish the used technique so that only the agents who use the knowledge representation associated with the corresponding technique can offer the medical consult service related to the underlying specialty.

Therefore, each reasoning agent is capable of using a reasoning method related with a knowledge representation. Each reasoning agent that wants to classify a new case will request the corresponding model created by the learning agent. Each model will be sent codified according to the type of representation created by the learning agent. In this case four types exist, as already we know:

1. Network weights. In this case the learning agent delivers to the reasoning agent the network weights in form of two numerical matrices. Preceding the matrices, the data of the network structure is sended: number of inputs and number of hidden units. The reasoning process consists of using the weights matrices to calculate the network output. To be able to do this, is also necessary that the learning agent send the attributes domain both inputs and decision, as well as the form to codify the input attributes and to decode the network output.
2. Rules with certainty factors obtained with the algorithm LRRS. In this case the set of rules is sent to the reasoning agent that serves as knowledge base. Each rule is of type:
   $IF\ A_1 = a_1\ AND \ldots AND\ A_n = a_n\ THEN\ d = d_i\ (C)$
3. Bayesian network. In this case the learning agent sends to the reasoning agent the set of attributes A, the domain of each attribute A_i, the domain of the decision attribute d and the tables of conditional probability for each combination attribute-class.
4. Decision tree. In the case of the models produced by ID3 the decision tree generated is sent to the corresponding reasoning agents.

In cases 2 and 4, the set A of attributes as their domains also must be sended to the reasoning agents.

Finally, reasoning agents can play the roll of avatar of a human being, but another agent with this particular roll must be developed, specially if we want use a low scale computer, such as a PDA or a cell phone.

JADE framework provides all the functionality needed for agents creation, communication between agents and yellow pages service; among other functionalities related with agent oriented develop.

4 Conclusions

This approach was developed mainly with the purpose of creation of an artificial medical community in which may be possible the implementation of multiple machine learning approaches. Tests done have demonstrated that the effectiveness rate of the four techniques used are very similar in the training phase, except the ANNs which performance has depended of correct selection of its characteristics parameters.

However, in the operation phase the performance vary significantly among them. The best situated have been the Bayesian Classifier, mainly when not all the attributes values are available. The other three models have had similar performances.

The distributed approach has permitted the use of machine learning models in middle and low scale computers, increasing their use.

We will have to do more work in the future, in order to incorporating more machine learning techniques, as well as another approaches of creating artificial communities.

References

1. Baldi, P. y Brunak, S. 2001. *Bioinformatics: the Machine Learning Approach.* The MIT Press. Cambridge, Massachusetts, 2001.
2. Bello, R. y Gaitán, J.J. *Tomando Decisiones Basadas en el Conocimiento.* Universidad Cooperativa de Colombia, 2003.
3. Cantú, F. J. *Learning and Using Bayesian Networks for Diagnosis and User Profiling.* CIA-ITESM, 2000.
4. Freeman, J. A. y Skapura, D. M. *Neural Networks. Algorithms, Applications and Programming Techniques.* Addison-Wesley Publishing Company, Inc., 1991.
5. Grzymala-Busse, J. W. And Ziarko, W. Data Mining Based on Rough Sets. In *Data Mining: Opportunities and Challenges.* Idea Group Publishing, 1994.
6. Jordan, M. I. and Bishop C. M. *Neural Networks.* CRC Handbook of Computer Science, CRC Press. Boca Raton, FL., 1996.
7. Kröse, B. and Van Der Smagt, P. *An Introduction to Neural Networks.*The University of Amsterdam., 1996.
8. Masters, T. *Advanced Algorithms for Neural Networks. A C++ Sourcebook.* John Wiley & Sons, Inc. Canada., 1995.
9. Michie, Spiegelhalter, D.D.J. and Taylor C.C. *Machine Learning, Neural and Statistical Classification.* MRC Biostatistics Unit, Institute of Public Health, University Forvie Site, Robinson Way, Cambridge CB2 2SR, U.K., 1994.
10. Pawlak, Z., Grzymala-Busse, J. Slowinski, R. and Ziarko, W. *Rough Sets.* Communications of the ACM. November 1995. Vol. 38, N° 11., 1995.
11. Pawlak, Z. *Rough Sets and Data Analisys.* Proceedings of the 1996 IEEE Fuzzy Systems Simposium, 1996.
12. Voges, K. E., Pope, N. K. Ll. and Brown, M. R. Cluster Analysis of Marketing Examining On-line Shopping Orientation: A Comparision of k-means and Rough Clustering Approaches. In *Heuristic and Optimization for Knowledge Discovery.* Idea Group Publishing, 2002.