Improved Ant Colony Optimization Algorithm based Expert System on Nephrology

1. Mrs.J.Anitha 2.Prof.P.V.G.D. Prasad Reddy 3. Prof.M.S.Prasad Babu 4. Mr.N.V.Ramana Murty

1.Asst.Professor, Dept.of IT, GITAM University, Visakhapatnam ,
Andhra Pradesh, India
anithanv28@gmail.com
2. Professor, Department of CS & SE, Andhra University, Visakhapatnam,
Andhra Pradesh, India.
prof.prasadreddy@gmail.com
3. Professor, Department of CS & SE, Andhra University, Visakhapatnam,
Andhra Pradesh, India.
msprasadbabu@yahoo.co.in
4. Associate Professor, Department of MCA, G.V.P.College for Degree &
P.G.Courses (A), Visakhapatnam, Andhra Pradesh, India.
murty_nadiminty@yahoo.com

Abstract:

Expert system Nephrology is a computer program that exhibits, within a specific domain, a degree of expertise in problem solving that is comparable to that of a human expert. The knowledge base consists of information about a particular problem area. This information is collected from domain experts (doctors). This system mainly contains two modules one is Information System and the other is Expert Advisory system. The Information System contains the static information about different diseases and drugs in the field of Nephrology. This information system helps the patients /users to know about the problems related to kidneys. The Nephrology Advisory system helps the Patients /users to get the required and suitable advice depending on their queries. This medical expert system is developed using Java Server Pages (JSP) as front-end and MYSQL database as Backend in such a way that all the activities are carried out in a user-friendly manner. Improved Ant Colony Optimization Algorithm (ACO) along with RETE algorithm is also used for better results.

Keywords: Expert System – JSP – ACO – RETE-MYSQL

1. Introduction:

1.1 About Aritificial Intelligence:

Artificial intelligence is a branch of computer science capable of analyzing complex medical data. Their potential to exploit meaningful relationship with in a data set can be used in the diagnosis, treatment and predicting outcome in many clinical scenarios. Research in Artificial Intelligence and Expert Systems is simply the transfer of intelligence to a machine. This paper introduces a novel integrated approach taking the advantages of current technological trends in World Wide Web, rule based expert systems, JSP, MYSQL, Object Oriented Paradigm and Medical Nephrology. The approach adopted here is a client server architecture where the client is a simple PC with graphical user interface and server is JSP & MYSQL data base application server.[1][2]

1.2 About Expert System:

An expert system is a computer program conceived to simulate some forms of human reasoning (by the intermediary of an inference engine) and capable to manage an important quantity of specialized knowledge

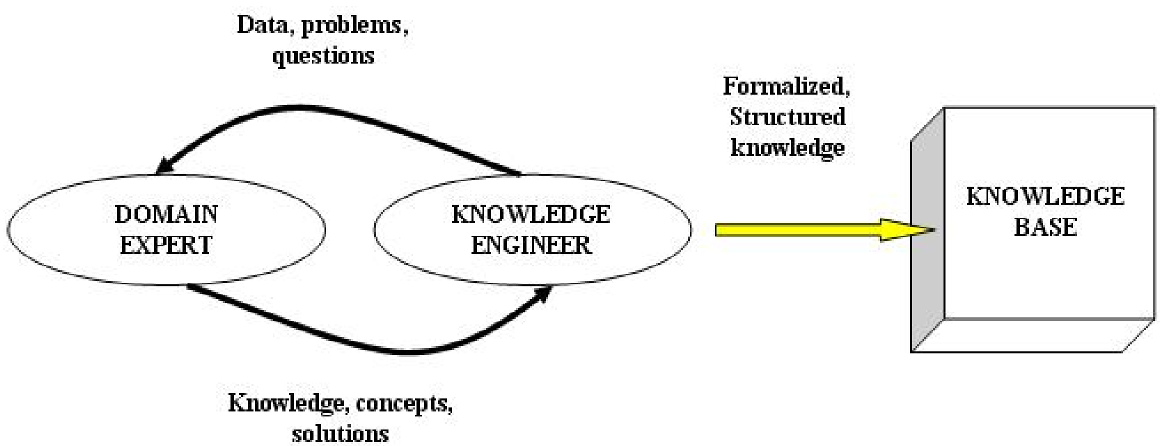

Figure : Typical knowledge acquisition processes for building an expert system

The basic components of the Expert System are

Expert:

Experts are Nephrologists and researchers in the field of Nephrology .Expert supply knowledge, both in the form of factual information and in relation to analytical methods they use to solve problems .

Knowledge Engineer:

The role of the Knowledge engineer is to collect expert advice from several experts and to convert them into rules and facts. The Knowledge Engineer is responsible for updating the changes made by the experts according to their research findings.

The Knowledge Engineer has to perform the following tasks:

1) Select Expert
2) Extract Knowledge Base
3) Develop the Knowledge Base
4) Plan the Expert System

Knowledge Base:

Knowledge Base consists of facts and rules as text files. User may be Nephrologists, patients and rural people.

User Interface:

The user interface is exchanging information between the user and the inference engine. It involves graphical user controls such as Buttons, Labels, List Boxes etc. This makes the system user friendly.

Inference Engine:

The inference engine design for the expert advisory system uses backward chaining strategy. The main objective of the inference engine is inferring the data from the knowledge base by using different rules of symbolic logic like modus ponens, resolution methods and interacts with the problem-solving unit

1.3. About Nephrology Expert Advisory System:

This Nephrology Expert System mainly deals with common disease, common symptoms and commonly used drugs in the field of Nephrology. Nephrology is a branch of internal medicine and pediatrics dealing with the study of the function and diseases of the kidney. It contains two modules namely Nephrology Information System and Nephrology Expert Advisory System provide access to the rural people at their doorsteps. Nephrology information system contains static pages in html providing information about different Common Diseases, Common Symptoms, Investigations, Drugs, Services and Preventive Measures collected from the domain experts in the field of Nephrology. This Expert Advisory System contains Nephrology Knowledge Base and provides JSP based dynamic pages. These web pages contain simulated expert advice on the subject, to the end users when they interact with the expert system online and submit answers to queries asked by the system.

2. Problem Analysis:

2.1 Domain Analysis for Nephrology Expert System

The domain is 'Nephrology Expert System' is the motivation for this system is to develop a new system which can give the entire information about the diseases, symptoms, chemical controls, preventions, self help and definitions etc… in the field of Nephrology.

2.2 Proposed System

Nephrology Expert System is the problem domain. The functionality behind the system is answering to the questions asked by the user and Experts using a data base connectivity between user and the expert system.

The proposed system is divided into two aspects

1) Static Part
2) Dynamic Part

In static part, the user can get all the static information about different Common Diseases, Common Symptoms, Preventions to be taken, and some Frequently Asked Questions (FAQ's) about different diseases in Nephrology field.

In Dynamic Part, the user is having an interaction with the expert system online, the user has to answer the questions asked by the Expert System in Self –Help option in the menu. Depends on the response by the user the expert system decides the disease and displays the disease and control of disease.

2.3 Functional Requirements

Inputs: The system needs the information about the symptoms from the user to produce the output.

Outputs: The outputs of the system will be:

1) Information Diseases
2) Small Description about the disease
3) Preventions

Store: The information collected through experts is stored as a database (Knowledge Base) that serves as a repository for quick processing and future retrieval. The system stores the information in html files.

- About Nephrology
- Structure and Function
- Common Symptoms
- Common Diseases
- Investigations
- Preventions

The System Stores the information related to expert design in knowledge base in the following ways.

Rules: A set of rules, which constitute the program, stored in a rule memory of production memory and on an inference engine required to execute the rules.

Dataset: The monitoring data is in the MySQL database. It can be used as any other data stored in a database. This greatly increases the opportunity with which you can conduct post-analysis of the monitoring data**.**

3. System Design

3.1 Nephrology System Architecture

In this model, the patient will provide the information about symptoms through Questionnaire, then the expert system process these symptoms with the help Knowledge based expert model using Machine learning algorithms and finds the probable disease, then the system will present the precautions for concerning to the diagnosed disease. Ant Colony Optimization algorithm and RETE algorithm is used for classification and finding the disease.

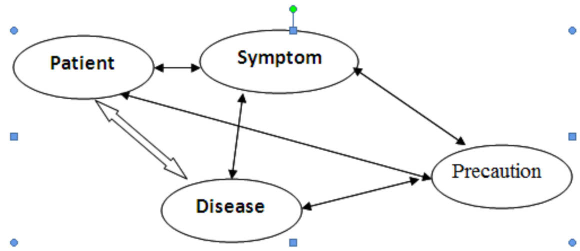

Fig: Nephrology System Architecture

3.2 Improved ACO Algorithm

Ant Colony Optimization algorithm used for the discovery of classification rules. In ACO algorithm each ant incrementally constructs/modifies a solution for the target problem. Target problem is the discovery of classification rules. Each classification rule has the form:

IF < *term1* AND *term2* AND ...> *THEN* <*class*> .

Each iteration of the REPEAT-UNTIL loop of Algorithm consists of three steps as follows comprising rule construction,6 rule pruning, and pheromone updating.

First Ant starts with an empty rule, that is, a rule with no term in its antecedent, and adds one term at a time to its current partial rule. The current partial rule constructed by an ant corresponds to the current partial path followed by that ant. Similarly, the choice of a term to be added to the current partial rule corresponds to the choice of the direction in which the current path will be extended. The choice of the term to be added to the current partial rule depends on both a problem-dependent heuristic function (h) and on the amount of pheromone (t) associated with each term, as will be discussed in detail in the next subsections. Ant keeps adding one-term-at-atime to its current partial rule until one of the following two stopping criteria is met:

• Any term to be added to the rule would make the rule cover a number of cases smaller than a user-specified

threshold, called Min_cases_per_rule (minimum number of cases covered per rule).

• All attributes have already been used by the ant, so that there is no more attributes to be added to the rule antecedent. Note that each attribute can occur only once in each rule, to avoid invalid rules such as "IF (Sex = male) AND (Sex = female) ...".

Second, rule *Rt* constructed by Ant is pruned in order to remove irrelevant terms may have been included in the rule due to stochastic variations in the term selection procedure and/or due to the use of a short-sighted, local heuristic function – which considers only one-attribute-at-a-time, ignoring attribute interactions.

Third, the amount of pheromone in each trail is updated, increasing the pheromone in the trail followed by Ant (according to the quality of rule *Rt*) and decreasing the pheromone in the other trails (simulating the pheromone evaporation). Then another ant starts to construct its rule, using the new amounts of pheromone to guide its search

This process is repeated until one of the following two conditions is met:

• The number of constructed rules is equal to or greater than the user-specified threshold *No_of_ants*.

• The current Ant has constructed a rule that is exactly the same as the rule constructed by the previous No_rules_converg – 1 ants, where No_rules_converg stands for the number of rules used to test convergence of the ants.

Once the REPEAT-UNTIL loop is completed, the best rule among the rules constructed by all ants is added to the list of discovered rules and the system starts a new iteration of the WHILE loop, by reinitializing all trails with the same amount of pheromone. From a data mining viewpoint the core operation of Ant-Miner is the first step of the REPEAT-UNTIL loop of Algorithm , in which the current ant iteratively adds one term at a time to its current partial rule. Let $term_{ij}$ be a rule condition of the form $A_i = V_{ij}$, where Ai is the i-th attribute and V_{ij} is the j-th value of the domain of A_i. The probability that $term_{ij}$ is chosen to be added to the current partial rule is given by Equation(1):

$$P_{ij} = \frac{\eta_{ij} \cdot \tau_{ij}(t)}{\sum_{i=1}^{a} x_i \sum_{j=1}^{b_i} \eta_{ij} \cdot \tau_{ij}(t)} \quad ....(1)$$

where: η_{ij} is the value of a problem-dependent heuristic function for $term_{ij}$. The higher the value of η_{ij}, the more relevant for classification the $term_{ij}$ is, and so the higher its probability of being chosen. The function that defines the problem-dependent heuristic value is based on information theory, and it will be discussed in the next section. $\tau_{ij}(t)$ is the amount of pheromone associated with $term_{ij}$ at iteration *t*, corresponding to the amount of 9 pheromone currently available in the position i,j of the path being followed by the current ant. The better the quality of the rule constructed by an ant, the higher the amount of pheromone added to the trail segments visited by the ant. Therefore, as time goes by, the best trail segments to be followed – that is, the best terms (attribute-value pairs) to be added to a rule – will have greater and greater amounts of pheromone, increasing their probability of being chosen.

• a is the total number of attributes.

• x_i is set to 1 if the attribute A_i was not yet used by the current ant, or to 0 otherwise.

• b_i is the number of values in the domain of the i-th attribute.

A $term_{ij}$ is chosen to be added to the current partial rule with probability proportional to the value of Equation (1), subject to two restrictions, namely:

• The attribute A_i cannot be already contained in the current partial rule. In order to satisfy this restriction the ants must "remember" which terms (attribute-value pairs) are contained in the current partial rule.

• A $term_{ij}$ cannot be added to the current partial rule if this makes it cover less than a predefined minimum number of cases, called the Min_cases_per_rule threshold. Once the rule antecedent is completed, the system chooses the rule consequent (i.e., the predicted class) that maximizes the quality of the rule. This is done by assigning to the rule consequent the majority class among the cases covered by the rule.

Heuristic Function

For each $term_{ij}$ that can be added to the current rule, Ant-Miner computes the value η_{ij} of a heuristic function that is an estimate of the quality of this term, with respect to its ability to improve the predictive accuracy of the rule. This heuristic function is based on Information Theory [7]. More precisely, the value of η_{ij} for $term_{ij}$ involves a measure of the entropy (or amount of information) associated with that term. For each $term_{ij}$ of the form $Ai=V_{ij}$ –where Ai is the i-th attribute and V_{ij} is the j-th value belonging to the domain of Ai –its entropy is

$$H(W \mid A_i = V_{ij}) = -\sum_{w=1}^{k} [P(w \mid A_i = V_{ij}) \cdot \log_2 P(w \mid A_i = V_{ij})] \quad \text{..........(2)}$$

• W is the class attribute (i.e., the attribute whose domain consists of the classes to be predicted).

• k is the number of classes.

• $P(w|Ai=Vij)$ is the empirical probability of observing class w conditional on having observed $Ai=Vij$.

The higher the value of $H(W|Ai=Vij)$, the more uniformly distributed the classes are and so, the smaller the probability that the current ant chooses to add $term_{ij}$ to its partial rule. It is desirable to normalize the value of the heuristic function to facilitate its use in Equation (1). In order to implement this normalization, it is used the fact that the value of $H(W|Ai=V_{ij})$ varies in the range 0 £ $H(W|A_i=V_{ij})$ £ log2 k, where k is the number of classes.

Therefore, the proposed normalized, information-theoretic heuristic function is:

$$\eta_{ij} = \frac{\log_2 k - H(W \mid A_i = V_{ij})}{\sum_{i=1}^{a} x_i \sum_{j=1}^{b_i} (\log_2 k - H(W \mid A_i = V_{ij}))} \quad \text{........................(3)}$$

where a, xi, and bi have the same meaning as in Equation (1).

Note that the $H(W|A_i=V_{ij})$ of $term_{ij}$ is always the same, regardless of the contents of the rule in which the term occurs. Therefore, in order to save computational time, the $H(W|A_i=V_{ij})$ of all $term_{ij}$ is computed as a pre-processing step.

Rule Pruning

Rule pruning is a common place technique in data mining [3], the main goal of rule pruning is to remove irrelevant terms that might have been unduly included in the rule. Rule pruning potentially increases the predictive power of the rule, helping to avoid its over fitting to the training data.

Pheromone Updating

Recall that each $term_{ij}$ corresponds to a segment in some path that can be followed by an ant. At each iteration of the WHILE loop of Algorithm I all $term_{ij}$ are initialized with the same amount of pheromone, so that when the first ant starts its search, all paths have the same amount of pheromone. The initial amount of pheromone deposited at each path position is inversely proportional to the number of values of all attributes, and is defined by Equation(4):

$$\tau_{ij}(t=0) = \frac{1}{\sum_{i=1}^{a} b_i} \quad \text{.............................(4)}$$

where a is the total number of attributes, and bi is the number of possible values that can be taken on by attribute Ai.

3.3 RETE Algorithm

RETE algorithm is an efficient and fast pattern matching algorithm that provides a generalized logical description of an implementation of functionality responsible for matching facts against the production rules in a pattern-matching production system. In expert system where each rule is checked against the known facts in the Knowledge base before executing (firing) that rule and resetting to the top of the rule stack after execution, the Rete-based expert system builds a network of nodes, where each node (except the root) corresponds to a pattern occurring in the left-hand-side of a rule. The left-hand-side of a rule is a path from the root node to a leaf node. Each node has a memory of facts against the pattern. Traversing through the root and leaf node, as the combination of facts relates exactly to the pattern the corresponding rule is triggered. [5]

4.Results & Discussions

Fig.Nephrology Expert System

Fig: Nephrology Expert System

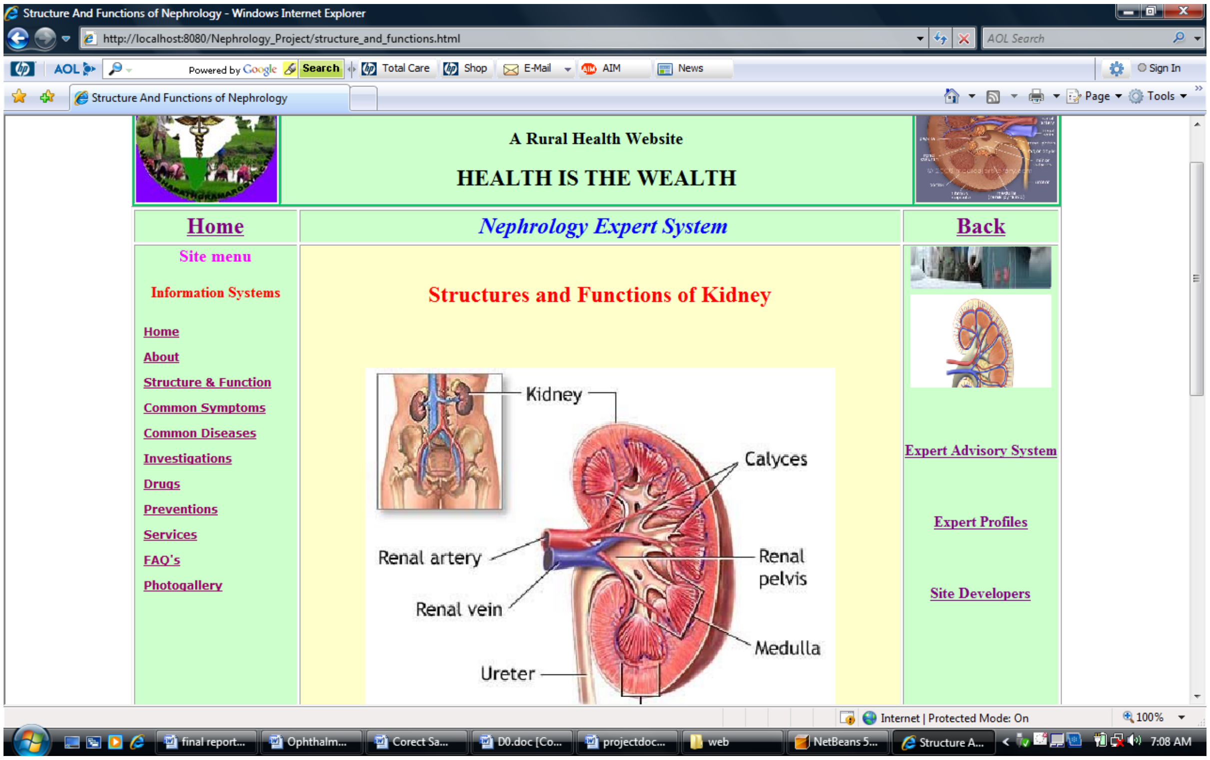

Fig:Structures and Functions of Kidney

Fig:Common Symptoms

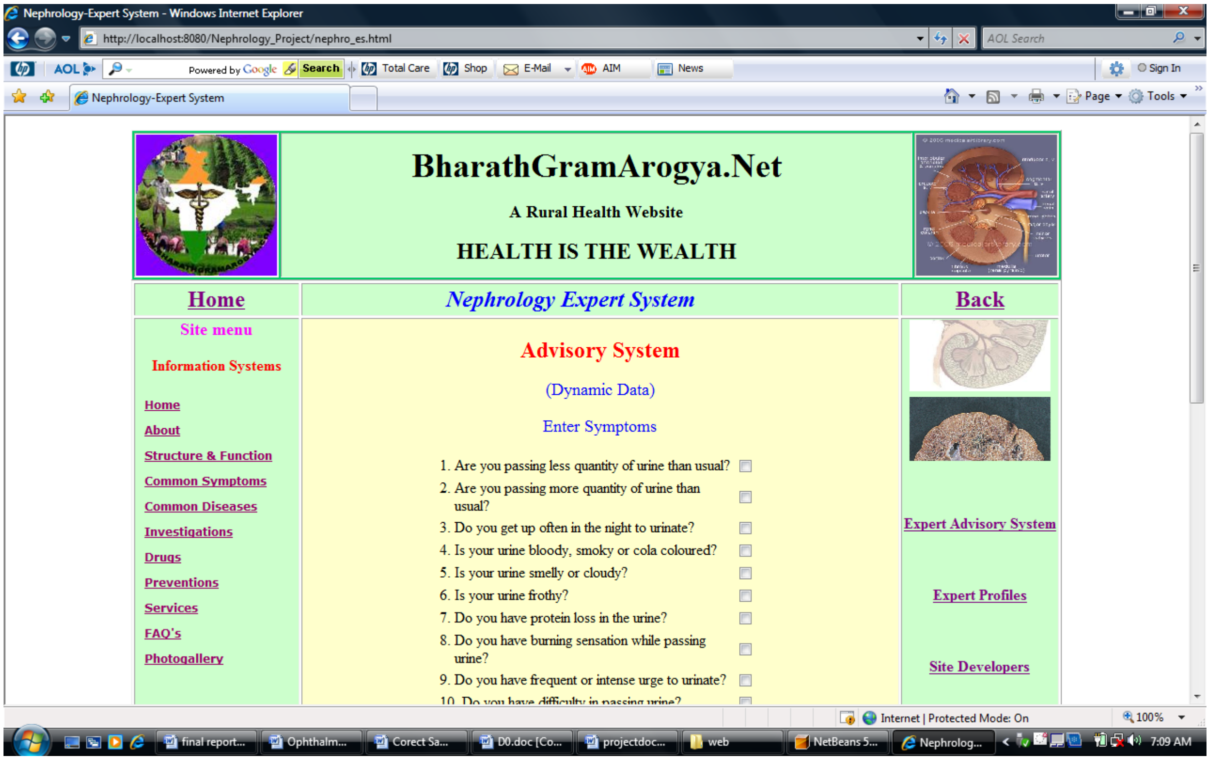

Fig: Expert System of Nephrology

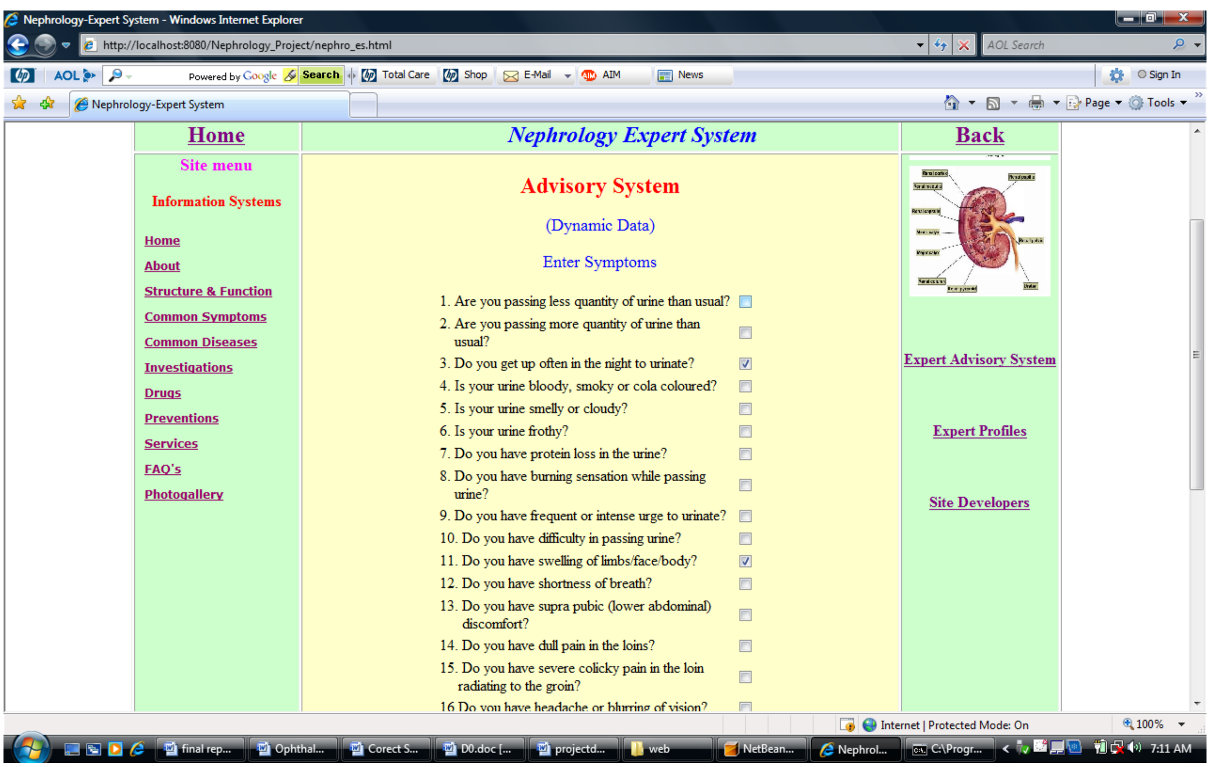

Fig: Expert System of Nephrology

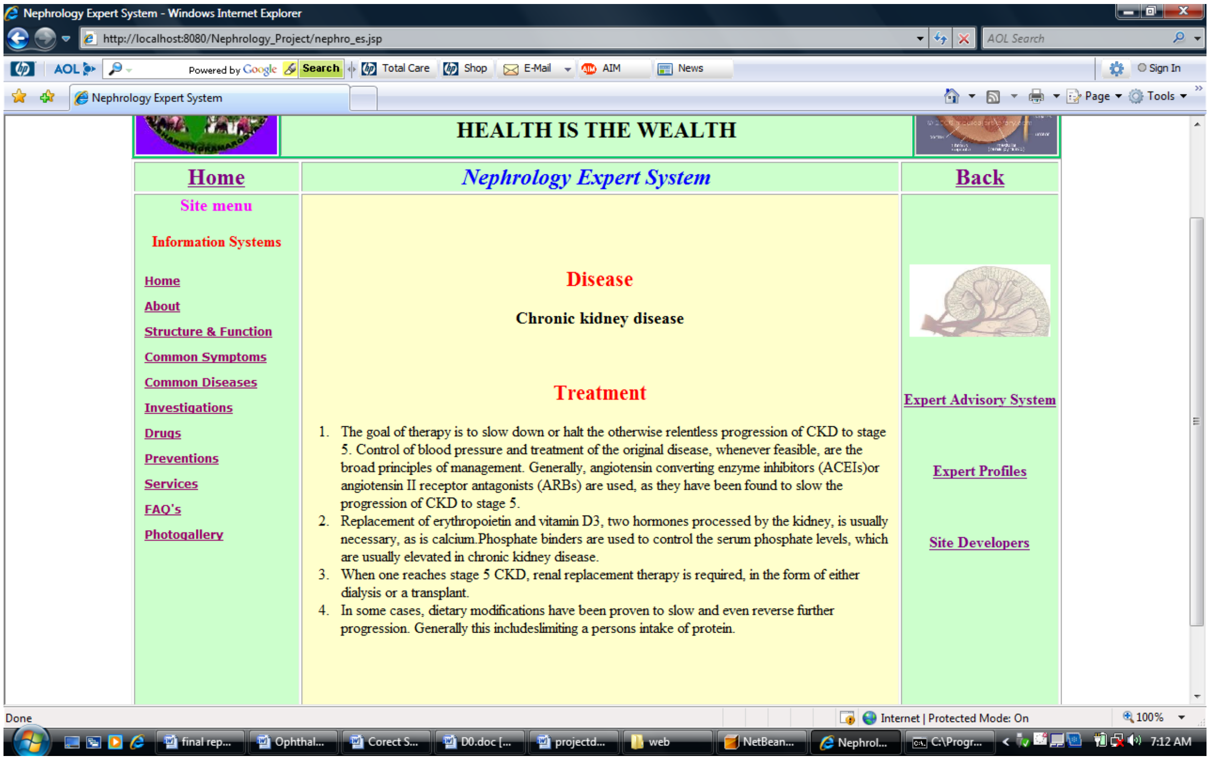

Fig: Expert System of Nephrology

5. Conclusion & Future Work

The work in this paper has revealed and emphasized the effectiveness and importance of expert system as a decision support tool for the patients and diagnosis services. The data analysis reveals that some disorders, which have not been identified and treated by doctors, can also be managed by expert systems. It is also very clear that there is difference in the advice quality and consistency given by the expert system almost near to the human experts. In Future there is a proposal that implementing this expert system in all rural & regional languages for better understanding of the users.

6. Bibliography & References

1. M. Bohanec and I. Bratko, "Trading accuracy for simplicity in decision trees," *Machine Learning*, vol. 15, pp. 223-250, 1994.

2. E. Bonabeau, M. Dorigo and G. Theraulaz, *Swarm Intelligence: From Natural to Artificial Systems*. New York, NY: Oxford University Press, 1999.

3. L. A. Brewlow and D. W. Aha, "Simplifying decision trees: a survey," *The Knowledge Engineering Review*, vol. 12, no. 1, pp. 1-40,1997.

4. "Rete: A Fast Algorithm for the Many Pattern/Many Object Pattern Match Problem", *Artificial Intelligence*, 19, pp 17-37, 1982

5.Ref: http://it.toolbox.com/wiki/index.php/Rete_Algorithm