Designing Self Learning Expert Troubleshooter for PC to Enhance User Satisfaction

Thesis submitted in partial fulfillment of the requirements for the award of degree of

Master of Engineering
in
Computer Science and Engineering

Submitted By
Mahak Agarwal
(Roll No. 801232011)

Under the supervision of:
Dr. Shivani Goel
Assistant Professor

COMPUTER SCIENCE AND ENGINEERING DEPARTMENT
THAPAR UNIVERSITY
PATIALA – 147004

June 2014

CERTIFICATE

I hereby certify that the work which is being presented in the thesis entitled, **" Designing and Implementing A Self learning Expert Troubleshooter for PC for Increasing User Satisfaction"**, in partial fulfillment of the requirements for the award of degree of Master of Engineering in Computer Science and Engineering submitted in Computer Science and Engineering Department of Thapar University, Patiala, is an authentic record of my own work carried out under the supervision of **Dr. Shivani Goel** and refers other researcher's work which are duly listed in the reference section.

The matter presented in the thesis has not been submitted for award of any other degree of this or any other University.

(Mahak Agarwal)
Roll No. 801232011

This is to certify that the above statement made by the candidate is correct and true to the best of my knowledge.

(Dr. Shivani Goel)
Assistant Professor
Computer Science and Engineering Department
Thapar University
Patiala

Countersigned by

(Dr. Deepak Garg)
Head
Computer Science and Engineering Department
Thapar University
Patiala

(Dr. S. K. Mohapatra)
Dean (Academic Affairs)
Thapar University
Patiala

Acknowledgement

The successful completion of any task would be incomplete without acknowledging the people who made it possible and whose constant guidance and encouragement secured the success.

First of all I wish to acknowledge the benevolence of omnipotent God who gave me strength and courage to overcome all obstacles and showed me the silver lining in the dark clouds.

With the profound sense of gratitude and heartiest regard, I express my sincere feelings of indebtedness to my guide **Dr. Shivani Goel**, Assistant Professor, Computer Science and Engineering Department, Thapar University for her positive and excellent guidance, constant encouragement, keen interest, invaluable co-operation, generous attitude and above all her blessings. She has been a persistent source of inspiration for me.

I am thankful to **Mr. V.P. Singh**, System Analyst, for guiding me in collecting the data for designing the knowledge base of the expert PC troubleshooter.

I am grateful to **Dr. Deepak Garg**, Head of Department, and **Dr. Ashutosh Mishra**, P.G. Coordinator, Computer Science and Engineering Department, Thapar University for providing the motivation and inspiration for doing this thesis.

Last but not the least I would like to express my heartfelt thanks to my parents and my friends who with their thought provoking views, veracity and whole hearted co-operation helped me in doing this thesis.

Mahak Agarwal
Roll No. 801232011

Abstract

In today's world there is tremendous growth in the IT sector and with it's this huge growth people's dependence on computers is increasing day by day. As presence of computers is the basic requirement to settle any IT sector then it is always necessary that they always work smoothly without any trouble to the user. So, to help the smooth working of computers i.e. to troubleshoot the PC faults and to diagnose the faults this Self learning Expert PC Troubleshooter (SLEPCT) is developed which is discussed in this thesis work. This will help users to solve their PC related problems by themselves on their desk itself without contacting any technician. And directly it will reduce the work load on the technicians as computer fault diagnosis is automated here. The system is composed of a user interface, a knowledge-base, an inference engine, and an expert interface and a hybrid learning algorithm. Additionally, the system features a mail server that helps admin to get notifications about new queries asked from different users and then helps in responding back to the new queries and sending replies through mails to the users by mails. The learning algorithm embedded in the system enhances the performance of the system as it increases the user satisfaction level by providing the solutions to the faults that are widely accepted by the people. The working of the system is based on the feedback taken from the user itself. So, in all the nominated system frees-up human technicians from manually performing laborious and time consuming maintenance tasks.

Table of Contents

List of Figures

List of Tables

List of Abbreviations

AI	Artificial Intelligence
BP	Bayesian Path
CAD	Computer Aided Design
CB	Case Based
CF	Certainty Factor
DFD	Data flow Diagram
ES	Expert System
ER	Entity Relationship
IASRI	Indian Agricultural Statistics Research Institute
IFDCP	Intelligent Fault Diagnoses and Control Paradigm
IT	Information Technology
JAD	Joint Application Design
MRETS	Methodology for Requirement Engineering Techniques Selection
PC	Personal Computer
SAINT	Symbolic Automatic Integrator
SOA	Service Oriented Architecture
SQL	Structured Query Language
UML	Unified Modeling Language

Chapter 1

Introduction

1.1 Background

There are number of fields where artificial intelligence has been used like stock trading, medicine, agriculture, robotics, scientific discoveries and many more. Not only it has been used, it has also provided tremendously good results in each and every field wherever it is used. It has now become an essential field of study where new researches can be done to aid people from different backgrounds ranging from medicine to agriculture and many more. There are number of definitions that are already given for artificial intelligence. Its earliest definition is given something like this: it is the act of making computers or some other machines to think like humans and behave like humans in all respects [1]. Whenever there is need to make some complex decisions or do some judgments computers are not able to take place of people but artificial intelligence helps in training computers to do so. AI (Artificial intelligence) helps computers in learning from experience i.e. learning from a sequence of happenings, helps in pattern recognition from large set of complex data, and to make decisions based on the human reasoning skills and knowledge. It has its application ranging from military for target recognition, to the entertainment world where it has its existence in computer games and many more. It is also been used in banks, insurance companies and hospitals where it is used to detect trends and predict customer behavior [3].

Artificial Intelligence is the stream of computer science that works with making decisions intelligently, learning, and adaptation in machines. Research in Artificial Intelligence is mostly concerned with producing some sort of machines that can help in automating the tasks which may require intelligent behavior. Artificial Intelligence is important because it enables to create a never-ending thought process and can help in solving our problems. With artificial intelligence, we can build computers, in collaboration of thousands of computers, that could all work in parallel to solve our complex and most dire problems.

According to Miller A. AI research can be subdivided into several major topic areas. These subdivisions according to Miller A. are [2]:

1. Robotics: This division deals with the working production of mechanical robots. It attempts to make the working of a mechanical robot more intelligent by some addition of vision or other sensing programs.

2. NLP: It covers logic and natural language programming. Here, user can communicate with a computer program through commands in English or another "human language."

3. Automatic Programming: A third area of AI research is automatic programming. Here researchers are trying to construct some systems or some other machines which can help in generating programs automatically from a set of rigorous specifications, or which can help software developers by providing other "intelligent aids ".

4. Expert System: The fourth area concerns expert systems. Expert system aids in solving difficult problems from a particular domain using human expert's knowledge and decision making ability.

1.2 Expert System

An expert system (ES) is a computer system that has the decision making capability of a human expert and provides its judgments in different scenarios. It behaves like human expert in all respects. Expert systems are made to solve rigid problems by reasoning about knowledge like an expert and in contrast to conventional programming, it does not follow the steps of a normal developer. In ES the knowledge of human experts are stored in a knowledge base and using some inference techniques decisions are made regarding a situation with considerable degree of accuracy [4]. The main part of an ES is the knowledge base. It is needed that the knowledge base remains updated and correct

always. But the help required from knowledge engineers should be kept minimum, so that it could be automated. This is done by making a system learn through machine learning. Different learning approaches can be used to train an expert system. Basically ES works with its 2 major constituents i.e. knowledge base and the inference engine which are shown in the rudimentary diagram of the ES architecture in figure 1.1:

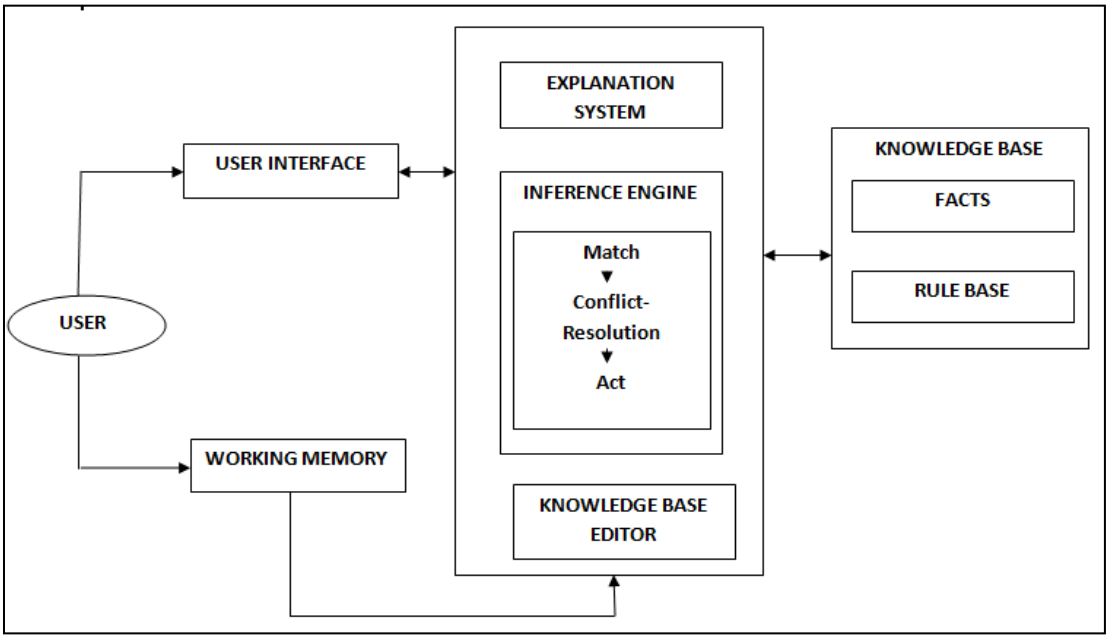

Fig1.1 Block Diagram of Expert System

First expert system was given by researchers in the Stanford Heuristic Programming Project, including the "father of expert systems" with the Dendral and MYCIN systems. Major contributors to the technique were by Bruce Buchanan, Edward Shortliffe, Randall Davis, William Van Melle, Carli Scott and others at Stanford. Expert systems were among the first truly successful forms of AI software.

1.3 Evolution of Expert System

One pioneering effort in AI concerned is information processing and to learn more about human cognition by constructing computational models. The earliest approach in this field was the General Problem Solver [2]. It was basically produced to solve the problems which are solved by an individual on a daily basis. There were number of

problem solving programs developed in 1960. Like James Slagle of MIT wrote SAINT which stands for Symbolic Automatic Integrator which solved integral calculus problems and when it was tested it was able to solve 84 out of 86 problems correct[2]. After few years at MIT, Daniel Bobrow wrote a program entitled STUDENT which solved high school algebra word problems. But a general problem solving program was not sufficient for what was actually needed at that time. There was no program that could simulate intelligent human behaviour without sufficient knowledge. So, researchers developed programs that could behave like an expert. As a human expert has significant knowledge in some particular field and not in all fields and also a human expert can solve easy to difficult all type of problems in that field of specialization and is able to solve very difficult problem in that field only. In those cases where a solution is not clear-cut, human experts can provide some solution's reliability. And also he can explain the derived solution. Hence, ES basically reasons about the knowledge in the knowledgebase and provides decisions like humans in some specific field to solve a problem in that particular field only. In addition to this expert system is also able to provide the reasoning about the solution and its reliability too [2].

1.4 Composition

There is difference between an expert system and a standard conventional program. In conventional program intelligence is there within the code only and it will solve a problem by following a sequence of instructions called algorithm which is basically designed only for that problem. Here code execution takes place in sequence until and unless a branch is explicitly introduced. But unlike this, in expert systems intelligence is separated from the reasoning portion. Hence any new knowledge can be added or some modification can be done to the existing knowledge without affecting the reasoning portion.

The general architecture of an expert system is shown as:

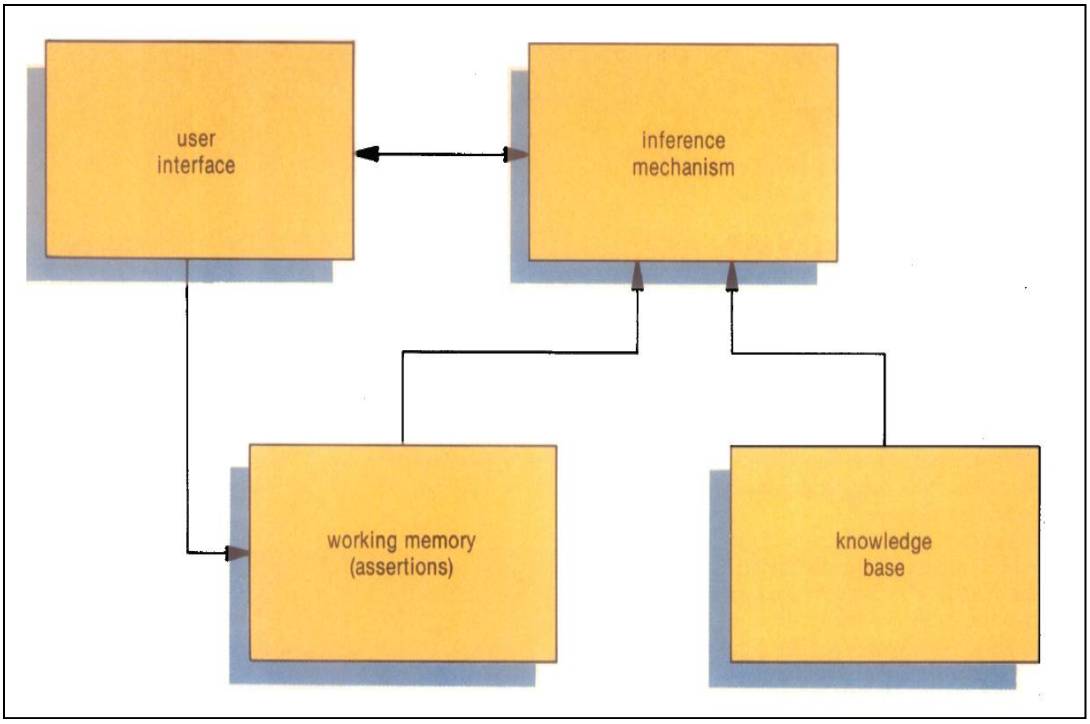

Fig1.2: Architecture of Expert System[2]

Knowledge Base: It consists of the data and information gathered from the human experts in the form of facts and rules. To attain the expert like performance this knowledge gathering is important. Knowledge here is collected from the experts and stored in the knowledge base in an encoded form. Generally it is stored in natural language. Knowledge base is restricted to a specific area only i.e. it contains information about a certain topic only, it is domain specific. The knowledge base is expressed with natural language rules like: "IF.. THEN..".

For example :

"IF figure D is a rectangle and IF all four sides of D are equal THEN the figure is a square or a rhombus."[2].

This format of representing rules is advantageous because it is used in everyday language and is very rare in computer science. Rules provides the knowledge that is used by the expert system.

Inference Mechanism: This module of an expert system is called as the brain of the system as it helps in deriving the new knowledge from the current knowledge that is stored in the knowledgebase. Algorithms are contained in this which help in searching the required information from the knowledgebase and other data that is required to choose appropriate rules and facts that match to the given condition. It also provides the features that provide the explanation of how a particular result has been obtained and which particular rules have been used to attain that result. In case where there is no clear cut decision for a problem, it allows to attach a probability factor there that results into certainty of that solution. An expert system then applies reasoning and problem solving techniques to knowledge in some specific area. The inference mechanism is independent of the topic area of the knowledge base.

Working Memory: The information concerning the specific problem to be solved is stored in the working memory section, sometimes referred to as the assertions portion.

User Interface: A user communicates with the expert system through a user interface.

1.5 Pioneers of Expert System

Many successful expert systems have been developed since 1960's. A brief introduction of the main expert systems is discussed below:

DENDRAL Project: This project began is considered as the first expert system which began in 1965 at Stanford University. This project mainly aims at inferring molecular structure of an organic compound from mass spectrographic data [2]. Knowledge base here is developed by encoding the experience and heuristics which were used by chemists. Prior to DENDRAL an algorithm existed that could provide all possible molecular structures but the exhaustive search was not possible because possibilities were very large in number. A significant feature of DENDRAL is, it not only consists of rules and facts in its knowledge base, it also consists of the heuristics. That is, knowledge base

consists of that rules of thumb and processes that chemists would use to infer constraints on the mass spectrographic information.

MYCIN Project: This system is designed to diagnose infectious blood diseases and provide appropriate antibiotics to cure that disease[2]. This system provides the explanation of how it reached to a conclusion. Knowledge base consists of production rules with their certainty factors. There are two more systems that are evolved from MYCIN which have helped in development of number of other expert systems.

As in expert systems knowledge base is not attached with the inference engine, so it may be theoretically possible to detach knowledge base from the expert system and substitute the knowledge base with some other one from some other domain. That is what actually was done with MYCIN, researchers took the expert system as it is without the knowledge base and called the system as EMYCIN. This system provided base for several other systems that helped in diagnosing respiratory diseases.

The second important system which evolved from MYCIN is TEIRESIAS. It automates the acquisition of new rules in the knowledge base. This acquisition program is independent from the MYCIN domain and became an expert system of its own. The knowledge in TEIRESIAS consists of facts and rules about the structure of the facts and rules in MYCIN. This knowledge is called as meta knowledge because here the level of abstraction is higher than the knowledge about a specific domain. This system can be used to develop knowledge base for any other domain as long as the structure of rules is same as that of MYCIN.

1.6 Types of Expert System

ES can be classified into following categories on the basis of how the knowledge is stored in its knowledge base and how it acquires the knowledge:

1. Rule Based Systems: This is the most popular paradigm for the knowledge representation in the knowledge base. Here, the domain specific knowledge is stored in the form of rules in the knowledge base. Knowledge is captured in a qualitative and propositional form. Knowledge is separated from the reasoning part.

2. Case Based Systems: Instead of rules, case based systems uses cases and their entire solutions. Here, similarity and analogy play a great role as similar kind of cases searched and solutions are found based on analogy. An operational cycle is followed here: find the most similar case, case is reused to find the solution of the problem, the given solution is analyzed and then that solution is provided as the final solution for the new case.

3. Hybrid Systems: In these types of systems both kind of approaches i.e. rule based and case based are followed.

4.Shell based Systems: It allows the quick development of the system as it has everything except the actual knowledge in the knowledge base. Different domain specific systems can be developed using the same shell. Hence, development efforts are reduced. It may include developer tools, debugging utilities, APIs, knowledge acquisition interface and other tools.

1.7 Learning Systems

A given expert system can fall in any of the above categories mentioned but the systems developed so has foremost shortcoming i.e. lack of dynamism. This lack of dynamism is highly paid as it cannot cope with the fast changing environment. And because of this expert system cannot perform at par in industries. To tackle this problem, learning should be embedded in expert system so that it can stay in tune with the changing environment, keep itself updated and keep on improving so that it can perform in a better way as compared to its previous performance.

"Learning" can be defined as the capability of a system to adapt to a phenomenon such that whenever that phenomenon takes place it performs in a better way as compared to the previous attempts. Different learning strategies are used for making a system to be a learner. Learning strategies can fall into any of the following five categories:

1) Supervised Concept Learning,

2) Conceptual Clustering,

3) Analytical learning,

4) Genetic Algorithms and

5) Connectionist learning.

These may use many learning technologies classified as: 1) Fuzzy Logic, 2) Neural Networks, 3) Genetic Learning and 4) Self-Learning Systems [5]. All these technologies have their own benefits which are currently used in wide spectra.

- Fuzzy Logic: Fuzzy logic was given by Zadeh in 1965. It's a multivalued logic where intermediate values are allowed between conventional values like low-high, true-false, yes-no, small-large etc. It's a very beneficial technique for controlling very complex systems and expert systems are also handled using fuzzy logic. It employs mathematical relationships among the variables to find the solution of the given problem. It does not provide an exact solution rather provides a solution between possible solutions. Normally fuzzy logic is used with rule based expert systems.

- Neural Networks: It depicts the process of human brain. A large number of data sets are needed to train the system to get good results. These are used in classification, pattern recognition, and to generate some predictions on problems which it has never encountered before. Even if a node fails it will work and provide good results having the fault tolerance ability. Neural network is being used in a number of applications like stock market forecasting, pattern recognition, and other different application areas [5].

- Genetic Learning: It works on the principle of "survival of the fittest". A model of the problem is developed where possible solutions of the target problem are proposed using different genetic operators like selection, crossover, mutation. Then the solutions which are generated if do not fit to the problem are killed while others survive in a pool where these are mixed together to generate some new good solutions.

- Self-Learning Systems: Mainly self-learning ES are supervised learning systems. Self-learning systems get their rule base from some a priori information. The system keeps on improving itself based on the past experience. There exist a number of methods to help ES in adaption. These systems may be recursive if the goal is not reached and its success/failure ratio keeps on altering depending on its performance. To enhance its success rate it should be coupled with different learning technologies like fuzzy logic, neural networks, genetic algorithms etc.

By embedding several learning technologies learning potential of the system gets raised and the profitability of the system is enhanced. Example is a system called NET-Link+ which was developed with neural net, fuzzy logic and genetic learning coupled together in order to recognize optimal telecommunication routes in real time [5]. There are a large number of expert systems that are developed in diverse areas using different learning strategies and different learning technologies. Existing expert system can be broadly categorized in following categories depending on the learning criteria followed: Supervised Learning ES, Unsupervised Learning ES, Self Learning ES and e-Learning/Web based ES.

1.8 Characteristics of an Expert System

Characteristics of expert systems that are normally observed are [1]:

- Increased Performance: System should be able to respond either equally or somewhat better than the human experts.

- Reliability: System should be reliable enough so that they are not prone to any crash, otherwise the system will not be in any use.

- Flexibility: Because an expert system may have large amount of knowledge therefore system should be flexible so that some new information can be added to that or some modifications can be done to that.

- Reasonable: System should be capable of providing reasons of how it attained the result. It should not just behave like a black box that provides miraculous answer, instead it should behave like human expert and provide the explanation for its results.

- Appropriate Response Time: Expert system should be able to provide answers in a proper time so that the time taken is either same or somewhat better than the time taken by human experts.

1.9 Advantages of Expert systems

Major advantages of expert systems are [1]:

- Decreased Cost: Cost of granting solutions to people is decreased to a greater extent.

- Permanence: The knowledge present in the system is permanent as it will last indefinitely unlike human experts who may quit or die.

- Collection of Expertise: As expert system consists of knowledge from varying number of experts, therefore, using an expert system for solving a problem helps in getting knowledge from different experts simultaneously at any time of day or night.

- Rapid Response: Expert system may be able to provide response faster than a human being depending on the hardware or software that are used. So, in some

emergency situations where fast responses may be needed, real time expert systems may play a great role.

- Steady, practical, and complete response at all times: Expert systems are capable of performing steadily in all conditions. This is important in emergency conditions where efficiency of human expert may decrease due to fatigue or stress and may not provide their peak efficiency.

- Increased Reliability: Expert systems always provide reliable solutions as they always agree with the expert, until and unless expert does not make any mistake.

- Intelligent Database: Here databases are accessed by an expert system in an intelligent manner. An example is data mining.

- Explanation: The expert system also has the capability of explaining the reasoning that can lead to a conclusion in detail.

- Low degree of Danger: Even in hazardous environments expert systems can be used where a human expert may fail to provide its services.

1.10 Summary

In this chapter brief introduction about artificial intelligence is given. Expert system which is one of the classes that comes under artificial intelligence is discussed. Basic constituents of expert system are: knowledge base, inference engine, working memory and its user interface. Alone expert systems can't perform at a par to meet industry requirements. So learning technologies are discussed which can be embedded in expert systems to increase their performance. There are number of advantages of expert systems that make them so useful are given.

1.11 Thesis Outline

The first chapter briefly describes the background of expert systems which comes under the classification of artificial intelligence. Its evolution, basic architecture, pioneers in expert system development, types of expert systems, learning systems, characteristics and advantages are covered here.

The second chapter covers the literature survey in which requirement engineering process for expert system are covered. Along with this various expert systems that are developed in varied domains are discussed.

The third chapter covers the problem definition and the scope of the thesis work. It describes why this work has been done and how it reduces the load on technicians and normal users.

The fourth chapter covers the work done for design and implementation of the learning system that has been developed and describes how it works.

The fifth chapter includes the results obtained from the experimental procedure.

The sixth chapter includes conclusion and future scope.

Chapter 2

Literature Review

2.1 Requirement Engineering Techniques for Developing Expert System

Expert systems are basically developed to help in solving complex problems by reasoning about knowledge already known like a human expert does. It does not follow the procedure as followed in the conventional programming by a developer. It is not as easy to develop successful expert systems as it seems. There are certain factors which can lead to failure of expert systems and among them requirement engineering for expert systems is the one. While developing expert systems, developers pay least attention to the requirement engineering that are needed for an expert system. Expert system is considered failed if does not meet the user's needs even if it works fine. Therefore, for successful development of expert systems it is necessary that emphasize should be on requirement engineering process of expert systems. Here, analysis of expert system attributes, requirement engineering procedure in expert system development and the proper techniques that can be applied to expert system development are given. After that, the most suitable techniques for the expert system development based on the analysis are given.

So, similar to other software projects, expert systems also need requirement engineering to be done. Here, requirement engineering techniques help in collecting the requirements from the users to design a system that can do tasks that are expected to be done by human experts. Requirement engineering is the phenomenon of determining, analyzing, pruning, documenting, and validating the wishes of stakeholders for a system [6]. Jiang et al. [7] has stated that bad requirement engineering techniques or procedure will lead to project's failure. Therefore, suitable techniques should be used which prevent time and cost wastage and prevent redundant work during the expert system development.

2.1.1 Need of Requirement Engineering Techniques

While the development of expert systems suitable requirement engineering techniques should be selected for each system to be developed as every system has different requirements from the other one. Therefore, care should be taken while selecting the combinations of different techniques because a mistake made at this step may lead to failure of the expert system. One of the remarkable work is done by the Jiang et al. called *Methodology for Requirement Engineering Techniques Selection (MRETS)* which helped requirement specialists to select proper techniques on hand [8]. This was basically applied to industrial software projects which presented the combination requirement engineering techniques because Jiang et al. has given the statement that there is no one technique which has the ability of providing solutions to all requirement engineering problems [7]. On the other hand, it is stated in a study on requirement syntax that stakeholders provide their requirements in natural language that is unstructured and creates complexity, ambiguity, duplication, omission, inappropriate understandability and implementation during the system development[9]. During the early stages of system development normally requirements are not well defined which may lead to excess or deficiency of information and then knowledge engineers may waste their time to search the proper problem domain so that stakeholder's requirements are fulfilled. To develop a good quality ES, there should be active involvement from all the stakeholders like users, customers etc. so that it is easy for knowledge engineers to understand their requirements. So, it is in the hands of stakeholders to show all the concerned issues and have their participation during the progress of project and during its implementation. So, there active participation is needed [8]. And for this, interviews, workshops and evolutionary prototyping for requirement elicitation can be done.

2.1.2 Expert System Requirements

First the analysis of expert system development is done and after that analysis on the requirement engineering process in expert system development is done to find all the

requirements and the goals of the system and then specific techniques that can be applied to develop that system are analyzed by providing justification for the selected techniques.

1. Analysis of Expert System Development

It is very important to know that what is the procedure or points that are required to be taken care for the development of a successful system. A successful system is one that is:

- Intelligent: Means which can perform the task as intelligent as human thinking.
- Economic: Errors should be minimized to lowest so that it incurs low cost.
- User Friendly: Whenever a system is used for the first time by any user, user friendly system will help in saving his time and effort.

Tyran and George [10] have given some factors which are stated below which affect the success or failure of any expert system:

- Organizational Factors: It includes some external factors such as participation from the users, customers i.e. all stakeholders for their commitment towards the project and managerial support.
- Implementations Activities: It includes identification of user's needs, demonstration of the ES, training and its maintenance.
- Technical Factors: It includes all the system development tools and software needed and all the technical aspects needed to develop the system.
- Ease of use: System should be easy to use.

In the table given below there are some characteristics and attribute that may affect an ES:

Table 2.1 Factors Effecting expert system

Factors	**Description**
Project Size	Different sized projects need different requirement techniques
Requirement Volatility	High requirement volatility projects need more flexible techniques in requirement engineering
Project Category	Different techniques needed for different categories of projects
Safety Criticality	More caution and serious techniques needed for more safety critical systems
Project Complexity	Complex projects need different requirement engineering techniques
Project Cost Constraint	Knowledge engineers will have difficulties in completing projects if budget is not sufficient
Project Time Constraint	Expert system might not meet its requirement if time is limited.

2. Analysis of Requirement Engineering Procedure in ES

There are basically five important tasks in the requirement engineering process according to Cheng and Atlee [11] that are as follows:

Table 2.2 Task in Requirement Engineering Process

Tasks	Description
Requirement Elicitation	• Here the requirements are identified to attain the objective of ES • All the stakeholders need to be identified here to collect all the requirements during this early stage of elicitation as it will affect the success rate of the system • Focus group, brainstorming and interviews are conducted here
Modelling	• Abstract description is created that is amenable to interpretation • Data is organised so that it can be understood by all the stakeholders • ER diagram, UML and data flow diagrams are used to create models here
Requirement Analysis	• It is a task of understanding data, functions and interface for the system • Here, ambiguity, inconsistency and incompleteness can be identified • Non-functional requirements of expert system are also analysed • Card sorting, conflict management and viewpoint based analysis is done
Validation and Verification	• It is ensured that the requirements of expert system are valid so that they are suitable for the further development cycle that need to be covered • Verification makes clear of ambiguous requirements • Prototype, inspection, requirement testing and requirement checklist are conducted
Requirement Management	• It has the responsibility of managing the requirements for all time to come • Requirements are documented

The set of techniques used are shown in figure 2.1:

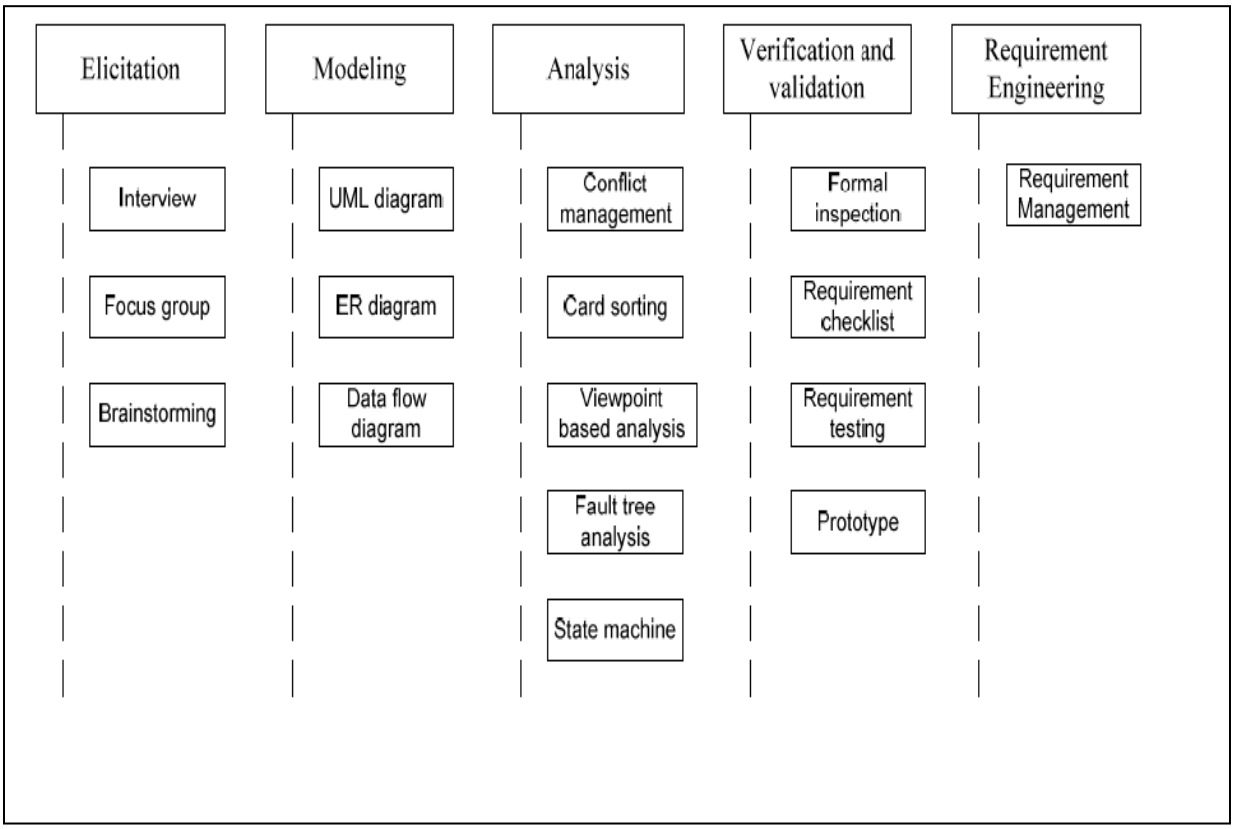

Fig 2.1: Series of Requirement Engineering Techniques in Expert System Development[8]

2.1.3 Specific Requirement Engineering Techniques for Expert System

There exist a number of requirement engineering techniques which can be used for expert system development in different tasks like Requirement Elicitation, Modeling, Analysis, Verification and Validation and Requirement Management in requirement engineering process. The factors used for selection will be based on the factors given by Kheirkhah and Dareman [6]. The different techniques use different methods to help out the requirement engineering process. Each task deploys number of techniques to completely and successfully accomplish the requirement engineering process. The detailed techniques used by different tasks are given in the following table. The selected techniques are given below in the table 2.3:

Table 2.3 Specific Requirement Engineering Techniques

Tasks	**Techniques**	**Conditions to apply**
Requirement Elicitation	JAD	When there are different stakeholders from different domains in the team
	Ethnography	Where defining implicit requirements is important and where usability and functionality plays a great
	Focus Group	Where communication is possible
	Interview	When some stakeholder's opinions are very important
	Brainstorming	Where ideas are to be generated
Modelling	UML Diagram	When project is very complex
	DFD	When project is very complex
	ER Diagram	When project is very complex
	Structured Natural Language Specification	Where unambiguous requirements can be written using notations
Analysis	State Machine	When project demands consistent and unambiguous states
	Fault Tree Analysis	When project is very complex
	Goal Oriented Analysis	When there are different objectives for different people
	Card Sorting	Where requirements can be prioritized
	Conflict Management	Where conflict of requirements can be solved
	Viewpoint based Analysis	Where different views of stakeholders can be identified
Verification and Validation	Formal Inspection	Where precise and consistent requirements are needed
	Requirement	When incomplete requirements can be identified and the

	Checklist	project is very complex
	Prototype	When project is very complex
	Requirement Testing	When the stakeholders can directly review requirement artifacts
Requirement Management	Requirement Management	When the requirement specification quality is the main issue and project is very complex

Based on the expert system characteristics suggested techniques are selected. In each process of requirement engineering these techniques are applied and they should be done in sequences.

2.2 Expert Systems and Optimization

This can be considered as a problem solving strategy where learning technique of expert systems is integrated with the optimization techniques. Optimization techniques have been successfully used since last three decades to solve problems in manufacturing and service organizations. Here first a problem is modeled and then proper algorithm is applied to solve that problem. Its advantage lies in the fact that once problem modeling takes place, an optimal solution can be found for that. But modeling a problem is bit difficult such that required data is easily available and it can be easily solved[12].

2.2.1 Operations Research as an Application Area of Optimization

Operation research is one of the application areas of scientific methods that aid in decision making. The operations research approach involves providing a model for the given problem and then solving that model using some heuristic or optimal algorithm[13].

Queuing, network models and mathematical programming are some of the most commonly used operations research models. Queuing is applicable to dynamic problems.

It can be used to find waiting time etc. Network model formulate a problem using graphs consisting of nodes and arcs. Mathematical model consists of an objective function and constraints which helps in modelling the problem.

Optimal algorithms are required to solve the formulated problem models but they may need high computation time. This can be correct to certain extent if [12]:

1. Problem is very large , or

2. The objective function or constraints are non linear, or

3. The variables are integers.

Operation research models have some limitations. In order to formulate a problem it may consider some assumptions and these assumptions may not hold in practice. One another issue is, it only considers the quantitative aspects without paying any heed to qualitative aspects.

2.2.2 Classes of Expert Systems

There are two classes of expert systems based on the operational modes [14]:

1. Stand-Alone Expert Systems: In this mode expert system uses data and constraints that are related to the problem and uses simple procedures to solve the problem. Operations research approach is not used.

2. Tandem Expert Systems: The tandem expert system, on the other hand combines the operations research approach with the expert system approach in order to solve a problem. This can be considered as expert system is attached to database of models and algorithms. The basic approach utilized in a tandem expert system is as follows: For the given problem a suitable model is selected, if not present then a proper model is constructed. After that for solving that model, a heuristic or an optimal algorithm is selected. Algorithm generates a solution which may be further modified to incorporate quality measure and to get an implementable solution.

2.3 Survey of Expert Systems developed in various domains

An enormous number of expert systems have been developed in past in diverse application areas like medicine, agriculture, machinery, labor production estimation etc. They all use different learning technologies while some of them are not based on any of the learning technique. This section gives an insight into different systems developed in different domains.

2.3.1 Agriculture

Expert systems in agriculture are basically developed to help farmers to solve problems related to pests, fertilizers, irrigation etc. A number of expert systems are developed like Islam et al. have developed an expert system for wheat crop that provided solutions regarding problems with the wheat crop and it also provided the reasoning facility [15]. Islam et al. have developed an expert system for Seed Spices that has data on 10 seed spice in its knowledge base [16]. It was a multicrop system that had information regarding the varieties, disease, insects and weeds. Devraj and Jain have developed an expert system called PulsExpert for the diagnosis of diseases in pulse crops [17]. The knowledge base of the system consisted information about 19 major diseases of pulse crops. The system provided a friendly interface to the farmers and could ask the questions in textual as well as pictorial form.

Hence, developing an expert system from the scratch is a tedious work. So, concerning this issue this expert shell was developed which allowed a single inference engine to be attached to number of knowledge bases thus virtually getting different expert systems for different crops.

An expert system shell ShellAG was developed which helped in developing a multicrop expert system that used the same inference engine used by “Expert System on Wheat Crop Management” which was developed by Division of Computer Applications, Indian Agricultural Statistics Research Institute (IASRI), New Delhi [18]. The main benefit of

this expert system shell was that it could deal with multiple crops. It had completely new interface as compared to the wheat expert system which allowed experts to add new information regarding specific crops using the same interface without actually accessing the database for each crop that is maintained under this interface. This means no special efforts were needed to develop whole new expert system for different crops. Hence, it was an efficient approach in the field of agriculture as the field is very vast and can't be dealt easily.

The changes that were made using this shell were also visible in the expert system. This is a case based system as it used the information from the previous system to develop a new system which worked for multiple crops.

Basic architecture of the system is shown in figure 2:

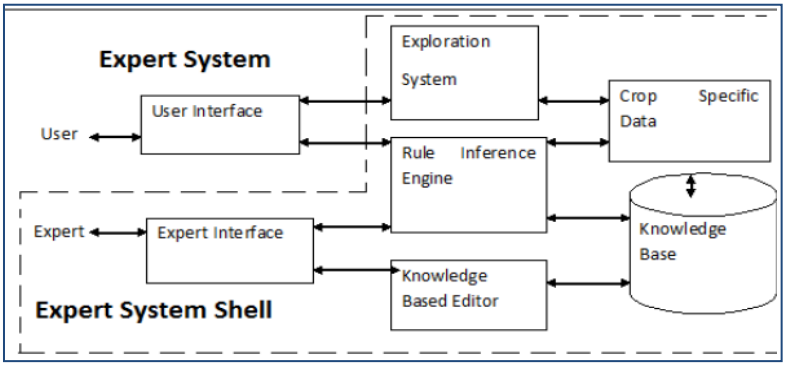

Fig 2.2Block diagram of shell based system used for agriculture [18]

There are a large number of rules in the expert system's rule table. A sample of rule table that the system used for variety selection in shown in table 1:

Table 2.4. Rule table of the shell based system used for agriculture [18]

State	Sowing Condition	Sowing Time	Variety
Haryana	Irrigated	Timely	PBW502

It indicated the rule that if state is ‘Haryana’ and sowing condition is ‘Irrigated’ and sowing time is ‘Timely’ then variety is ‘PBW502’.

There are some other systems that use this shell development like Bohanec has developed an expert system shell called DEX which provided decision support for the interactive construction of the knowledge base [19]. It aimed at developing qualitative multiattribute decision models. It helped decision makers to solve very complex problems by selecting the options that fulfills all the goals of the decision makers. Here the problem is divided into small sub problems where decomposition is presented in a hierarchical manner where options are achieved after applying aggregation method from leaves upwards towards the root. The system is rule based where fuzzy logic was applied to get the incomplete options. For providing explanation selective explanation method was used.

Another shell based system was developed by James et al. called AGNESS where rules were represented using the computational networks [20]. It dealt with both numeric and non-numeric type of data. It used computation networks for representing the rules defined by the experts. It used menu driven interface where eight explanation queries were given to enhance the amount of information that is provided to the user. This system has been used in varied field in real life like medicine etc.

2.3.2 Religious scripture

In 2012, Bilal and Mohsin have proposed a distributed expert system for the classification of Ahadith which used cloud as its backbone for its proper working [21]. The proposed approach was a very novel approach which helped in the classification of the Ahadith. Ahadith is the plural of Hadith i.e. the sayings of Prophet Muhammad. This expert system was very important because it helped in the classification of Hadith which is one of the essential task of a muslim. This system classified Ahadith among 24 types from seven broad categories and made computer to act like a Hadith expert to differentiate between authentic Ahadith and unauthentic one along with the reasoning details that helped the user to know why a certain type of classification has been done and to which

extent it is correct. This expert system also provided solutions to the communicational problems which the legacy web based expert systems were facing. This system also provided multilingual architecture for the localization of the application.

As this system was distributed web based expert system, communication played a very important role there and this worked by dividing a complex problem into number of small sub problems which could be solved in parallel to increase the system efficiency. This system was designed using Service Oriented Architecture (SOA). This basically enabled the system for cloud computing and software as a service as SOA provided the required infrastructure for cloud implementation [22].

A Hadith basically consisted of two things: text of the Hadith and chain of the reporters. Different rules on text of the Hadith and chain of reporters were applied to check their authenticity. Seven major classes of Ahadith can be categorized based on: Reference to a particular Authority, Links in the chain, Number of Reporters, Nature of text, Manner in which Hadith is reported, to Hidden defect found in Isnad or text of Hadith, Reliability and memory of reporters.

Major components in design of Muhadith were: 1) Web Interface , 2) Inference Engine , 3) Knowledgebase , 4) Parser and Fact Extractor , 5) Distributor and Aggregator , 6) Sub Inference Modules , 7) Explanation facility , 8) Database.

A web interface was provided to the users where users could query the expert system. The query was then passed to the web server where the data was passed to the parser module for parsing. User provided the text of Hadith and it might include chain of reporters. Names of the reporters were extracted from the chain by parser and parsed information was then passed to the fact extraction module. Before extracting the facts, fact provider module provided all the information about the reporter's name collected from the database. Then the fact extractor module after receiving the information from the fact provider module extracted the required facts. Facts were passed to the distributor module. It behaved like a scheduler and it distributed the facts to different sub inference

modules which independently handled each class of Hadith. Each sub inference module had its own knowledge base which consisted of different rules and information. Each sub inference module could also interact with the explanation module to provide reasoning details. Once the rules were applied on the provided facts results were returned to the combiner module along with the explanation details. Combiner module arranged all the information in a proper format and provided it back to the user. Knowledgebase was in the form of binary decision tables and rules were defined in IF THEN ELSE statements. Hence, it was a rule based expert system.

This system was implemented as cloud based expert system. Microsoft.NET platform was used and programming languages used were C#.Net and ASP.Net. SQL server was used for database management.

2.3.3 Power System

Another web based Expert System Shell for Fault Diagnosis and Control of Power System Equipment has been designed by Jain et al. [23]. This system was based on the pilot implementation of an internet based system in one of the electrical factory in Vishakhapatnam. This system was used to develop an Intelligent Fault Diagnosis and Control Paradigm (IFDCP) which was capable of handling problems with power system equipments generally Transformers, DC motors, AC motors and street lamps and it also helped operators for fault diagnosis. The system was very easily extensible as new equipments can always be easily added. The system was developed using ASP.NET platform and C#.NET programming language and MS Access DB tools. The knowledge base consisted of rules in the form of IF THEN ELSE statements and data were gathered from the experts of different areas from different electrical factories. It also consisted of historical data of different fault occurrences and their solutions; how those were recovered. This system was able to work as stand-alone and web based applications.

The system was developed in two phases. Initially a web based expert system shell was developed with a friendly graphical user interface and connectivity to database and after that Intelligent Fault Diagnoses and Control Paradigm (IFDCP) was developed for

electrical equipments using the web based expert system shell. This system provided access to multiple users in the world as it is a web based system. The basic design components of the expert system shell were:

1.Knowledge Base: It stored the knowledge gathered from experts, the historical data and data from books too. Rules were in the form of IF THEN ELSE statements. The knowledge base consisted of 3 sections:

a) Variable section: for variables in the knowledge base.

b) Rules section: for rules used in the knowledgebase.

c) Ask section: It consisted of all the questions from the user and all the options that are provided to the user to continue using the system.

2. Database: The data which can be stored as questions and answers were stored in the database. The table consisted of three fields called id, question and its answer.

3. Inference engine: It helped in determining the new knowledge from the existing knowledge in the knowledgebase. This process leads to the solutions to the faults that arose.

4. Graphical user interface: It consisted of an input interface which was the interface for the admin to access the system for some addition, modification, deletion or updating of data in the knowledgebase. A user interface which was for users to have an access to the system. the third was the working memory which consisted of information about a problem that the system has received and the temporarily solutions were stored here.
After the development of expert system shell, it is used to develop IFDCP. As fault diagnoses need extensive knowledge base, therefore, knowledgebase deals with the symptoms, causes and remedies of various faults that occur in a power system equipment and it also included the specifications and standard ratings of the equipment. All this data

was first stored in tabular form. After that, decision trees were developed to form the final rules of the knowledge base.

2.3.4 Animal Husbandry

There are some disadvantages of traditional rule representation. Therefore, an approach to correct the certainty factor(CF) of a rule was proposed using BP network by Liu et al. in 2005 which helped the system to gain the ability of self learning [24]. The method was applied afterwards to the cow disease diagnosis expert system which improved its performance.

BP algorithm provides a training algorithm for acyclic multistage networks. But it provides slow training speed and also has convergence problem. Therefore, to reduce this effect, it was integrated with the production system in the expert system so that the training speed got improved and the system could get the self learning ability. BP network provided a supervised algorithm for network training in two phases i.e. forward-propagation phase and back-propagation phase.

Rule in the production system were basically written as:

IF e1 AND e2 AND … AND en THEN h CF[R] where e1, e2.....were the premises combined by logical operators and h was the conclusion and CF[R] was the certainty factor of the rule. If the knowledge was confirmed then CF[R] value was 1 otherwise it was a real number in the range [0,1]. The CF value of the rule can be given as:

CF[h] =f (CF[R], CF[e1], CF[e2], … , CF[en])

Here, the function f was used to calculate CF[h], and the parameters like CF[R], CF[e1]...etc were given by the domain experts. But somehow, some quantization errors might get creped into when the CF values of rules and proof were finalized. And finally these errors might cause errors in the result of reasoning. Therefore, in order to reduce the error in the reasoning result and the dependence on the domain experts a procedure called self-learning was introduced.

To correct the CF values, rules were expressed in the knowledge base using the BP network. Neurons expressed the rules in the knowledge base. A rule expressed by neuron was given below:

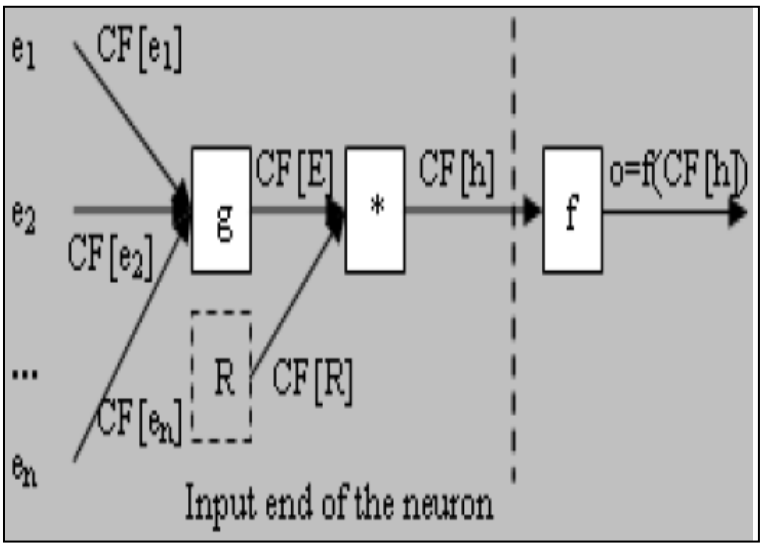

Fig 2.3.A neuron expressing a rule [24]

Here, CF[E] was the certainty factor of the preconditions that were computed according to the proofs and CF[h] was the output that is computed with the help of function f. Multiple rules can also be represented using BP network and hence reasoning task could be performed over that. Now the errors of the reasoning results were decreased as BP algorithm was used to train and correct the CF values. To speedup the process, the initial values could be given by the domain experts.

In cow disease diagnosis expert system this approach was followed. It had the ability of self learning and the slow convergence was not there and the performance was improved for the system after using BP network.

2.3.5 Database System

A frame based expert system is used for knowledge representation of expert systems having large knowledge base. Frames were first given by Minsky in 1975 [25]. Frames are basically used to represent concepts and classifications and taxonomy hierarchy too

[26][27]. A frame based object relational expert database system was developed which had a tight coupling with the expert system and the external knowledge base. Many systems have the capability to get connected to some other external databases and get knowledge from them using the inference engine of the system. So, such an expert system is developed where the system has its own inference engine and inference engine also exists on the external knowledge base side. So not only facts are sent back to the expert system for further inference rather inference results are sent back to the expert system.

A number of systems have been developed that had the capability to access some external database. Perk database connected to a database and loaded the frames from that database in the main memory for inferences [28]. It used OKBC operations for making connections. Two systems called EcoCyc and PARKA-DB loaded the frames into main memory only when they were required otherwise the frames were stored in the relational database [29]. These all were not having inference engine on the external database side. A system was developed where frames were kept in the object relational database. There was an inference engine on the external database side.

The developed system could be used in a medical diagnosis system in remote areas where experienced doctors may not be available. Therefore, to treat them medical staff there could take the help of the system to gather required information from the external knowledge bases and treat the patients. Hence, the system provided a good solution for acquiring more knowledge from number of external knowledge bases and improved the correctness of the solutions provided by the expert systems.

2.3.6 Traffic Management System

A self-learning-process based decision support system for traffic management in Beijing, China has been developed by Chen to provide solutions for different traffic circumstances for the traffic management in the city [30]. In this system a similar situation was

recognized in the past and suitable solutions to the problem were recommended. This was done by first matching the rules to provide a robust solution and then search was continued to get a similar case in past that had been solved successfully and the cases could be then stored in database after rigorously testing it. The more the number of similar cases found, the more the system performs efficiently. This system provided a suitable self learning approach based on knowledge stored in the knowledge base gathered from the experts and case based reasoning. Here in this proposed system a hybrid learning approach had been used which was based on rule based knowledge base, case based reasoning and simulation based development of new scenarios.

In rule based approach a manual was followed and corresponding procedures and measures were selected to implement to find the solution to any traffic situation. This was simply not the copy of the manual rather it was a robust study of the manual and then entry was done to the database. But what rule to use was effectively told by the case based studies. Rule based system actually provided the first level of the solution. Fuzzy logic was implemented to find out the similarities between the cases where pattern matching was done based on the likelihood probability. Dynamic Traffic Assignment (DTA) technology was used for the dynamic simulation to measure the evaluation based on the performance indicators calculated by DTA. Knowledge in the knowledge base is stored with the historical data and the data gathered from simulated cases. Hence, system is based on the self learning technique to find the solution to a given traffic situation for the sake of traffic management.

2.3.7 Fault Diagnosis

There are a large number of expert systems that are developed for fault diagnosis in diverse domains. One of the system was built called coal-cutter fault diagnosis expert system(CFDES) which was developed by Xu and Zhang in 1994 [31]. In this system the learning rules were adding some new rules and keeping the original rules to the knowledge base, adding some new rules and modifying some original rules. And to run these rules functions were designed to contribute to the OBSL technique. The functions

were designated as MR, MF, AD and CHECK to modify rules that contained the correct diagnostic, modify rules that lead to the incorrect diagnostic results, to add some new rules that can lead to correct conclusions and for checking the consistency of the added and changed rules respectively. For a fault, a shallow inference was chosen to make a diagnosis. If correct result can't be made by the system, deep inference was done to complete the diagnosis. Shallow inference resulted into diagnostic efficiency while the deep inference ensured the diagnostic effect. CFDES drew the correct conclusion using deep inference and using OBSL after that, which could maintain the consistency of the rules after the addition of some new rules and their modification which leads to the correct results. Similarly an expert system that diagnosed faults in an electronic hydraulic braking system was proposed [32]. This architecture combined the given knowledge with the fuzzy knowledge and used it for diagnostic reasoning.

2.3.8 Machinery

In diverse fields of machinery, expert systems exist which helped to improve the efficiency of the machines and to rectify the problems the machines faced. One of such system is used in pelletizing process. In pelletizing process, machine speed of chain grate depends on the material layer depth. As there is big fluctuation in the raw ball quantity there is instability in the return quantity and there is delay in material layer depth detection. So, an expert system was developed by Xu and Jing that helped in identifying the proper machine speed of chain grate and material layer depth was taken as the major control objective for this [33]. System had self-learning capability during the control process to minimize the effect on material layer depth control which was caused by the instability in the return quantity. Q_learning algorithm was used in the system to update the database quickly and efficiently.

In the pelletizing process three parts are there: Exhaust and preheating of raw ball, roasting of preheated pellets, roasted pellets cooling. Initially raw balls move on the machine tool of grate, air blast is exhausted and raw balls are preheated using residual heat and then these balls are roasted in the kiln. The machine speed of grate affects the

material layer depth. In manual control of the machine speed depending on the changes in solving the problem of material layer depth control. Control can be done in two ways like:

1. Feed forward control method of machine speed for grate

Here machine speed is calculated every two minutes according to the quantity of balls measured machine speed is given by S1 (m / min):

S1 =W / (KL0 W0)

Here, S1: machine speed (m / min),

W0: Width of grate in mm

L0: given value of material layer depth in mm

W : quantity of balls in kg / min

K is the ratio of balls ,t / m3 *106 . K is not a constant value.

Calculation by this method is simple but control effect is not so good.

2. Feedback control method of material depth for grate

As quantity of balls can't be determined but material layer depth can be detected, hence a feedback control method is used, where machine speed is given by

S1=VPLi/L0

where Li is the material layer depth at the ith period, Vp is the current machine speed before automatic control, Lmin and Lmax are the allowable deviation range of material layer depth. Q_Learning algorithm is implemented here to adjust the parameters of machine speed so that an acceptable convergence speed can be obtained. This automatic control system is implemented in AnGang pellet plant in China where material layer depth was controlled automatically (Xu and Jing, 2011).

2.3.9 Labor Productivity estimation

A fuzzy expert system has been given by Muqeem et al. which aimed at estimating the labor production rates by using the influence of qualitative and quantitative factor [34]. All the influential factors were gathered from the questionnaire survey. The system thus

developed gave higher linguistic and numerical accuracies with the Root Mean Square Error method as compared to the previous systems that were developed. As labor productivity is the main concern for performance of the construction industry, so, this system is very important for the good performance of the construction industry. In this fuzzy ES, input variables were related to the output variables using the linguistic values based on the if-then rules represented in the form of fuzzy logic. Fuzzy inference engine was used which consisted of rule base, database and reasoning mechanism. Similar systems for estimating construction labor product has been developed by Hongwei where fuzzy inference engine, fuzzification module and defuzzification module were used to estimate the labor productivity [35].

One system has been developed by Oduba where labor productivity for industrial rig pipe and weld pipe was estimated using fuzzy logic [36]. In the given system the factors were ranked according to a Likert scale between 1 to 5 where 1 means not important and 5 means very important. These factors were ranked according to the Relative Importance Index calculated using a formula. The scale was divided into three terms called low, average and high where membership value is assigned to every linguistic term between 0 and 1. Here, Pearson Correlation Coefficient method was used to carry out the analysis between input and output variables. Two variables were highly dependent on each other if the correlation coefficient was greater than 0.8 between the two variables. This system performed better than the systems developed by Hongwei and Oduba in terms of numerical and linguistic accuracy because Gaussian shape of membership function was used instead of triangular and linguistic terms were used five instead of three, weightage average method of defuzzification was used which resulted into better results.

2.3.10 Medicine

A self-learning expert system was developed by Wang et al. in Traditional Chinese medicine (TCM) for diagnosis using learning techniques like hybrid Bayesian network learning algorithm, Naı¨ve–Bayes classifiers with a novel scorebased strategy for feature selection and an approach for finding constrained association rules [37]. Causal diagram,

reasoning rules and association rules were used to present the learned knowledge. Here pathologic condition of patient was diagnosed with the help of the data that was gathered after having an observation with the naked eyes. After having such an observation and the patient's own report might help to draw conclusions which were saved in terms of syndrome. And accordingly treatment could be done for the patient. Key elements for syndrome differentiation were categorized and grouped into two categories: one indicated the place where the disease occurred like skin, heart etc and second revealed that what are the possible causes of that disease like cold, wind etc.

A standard syndrome-name database was created where the problem of giving different names to same syndrome by different people was solved. Diagnosis procedure can be divided into two phases: according to the patient's symptoms, diagnosis was performed. Then after that from the standard syndrome-name database, it was confirmed whether those key elements were included in that syndrome or not. The system was able to diagnose the diseases and was successful to a great extent.

2.3.11 Troubleshooter

An expert system for diagnosis of electronic devices was proposed by Ghaemi [38]. The knowledge was acquired during the development phase from computer aided design (CAD), in addition to this engineering data provided by the design engineer was also gathered. While troubleshooting when a new problem is fired the system reasons on the knowledge that is already being stored in the knowledge base so that the diagnose can be done efficiently. A knowledge base system based on case based reasoning used for troubleshooting complex equipment was proposed by Gupta [39]. This system helped in decision making by making use of stored cases that represents the previous experience. Another system is PCDIASHOOT which was made to help computer users and PC technicians in case when their computers fail to boot [40]. This system supports only IBM compatible PCs during the power-on self-test (POST). POST checks PC's hardware at boot time to check whether all hardware parts are present and are properly functioning. An expert system for troubleshooting car problems was proposed [41]. It helped a car's owner to diagnose their car problems and get solutions how to solve them. It consisted of set of rules and the starting problems and their sub problems.

2.3.12 Expert System for greenness identification in agriculture

An ES for processing the images captured from cameras for identifying the greenness in plants was developed by J. Romeo et al. in 2013 [42]. The images were collected from outdoor environments where the plants are affected by variable illumination conditions caused by either sunny or cloudy environment or it can be due to both. It was mainly experimented for maize and barley fields with weeds also, so that some actions can be taken for performing some treatments with chemical substances or some mechanical manipulations can be done. Here, the proposed system consisted of two modules: first one is for decision making that is based on the image histogram analysis and second one is the greenness identification. Here , two strategies are followed, one is based on the classical greenness identification methods and second is the Fuzzy Clustering approach. The system was able to apply treatments over site specific problems.

2.3.13 Traffic management in wireless local area networks

Frantti and Majanen [43] have done work to explore the congestion based on delay and flow control in real time wireless local area networks(WLANs) and mobile cellular networks(MCNs). It controls the traffic using the control system having hierarchy of expert systems embedded in it. In wireless networks, packet transmission interval and packet size affects the delay and throughput. Hence, the delay and the change in the one way delay is monitored by the controller on the destination node. Depending on this it can adjust the packet size on the source node for the incoming traffic by sending a control command to the source. If the traffic exceeds the capacity traffic is offloaded to some other access network. It was done by simulation of Voice Over Internet Protocol in OMNet and network simulator. This expert system is able to optimize the packet sizes depending on the traffic in the network. The work is proposed and simulated for IEEE 802.11b system, but the approach not only limited to this, this can be used for packet switched access networks.

A brief comparison of a lot of expert systems developed in diverse domains is done which is given in the Table 2:

Table 2.5 Comparison of various expert systems

Expert System	**Author**	**Domain**	**Type**	**LT**	**Cat.**	**Method/Algo.**
Diagnosis of electronic devices	Ghaemi, 1989	Trouble-shooter	RB	-	-	ID3-algorithm of Rulemaster
DEX	Bohenac M. and Rajkovic V., 1990	Decision Support	RB	Fuzzy Logic	-	Selective Explanation
Coal Cutter Fault Diagnosis ES	XU C.S. and Zhang Y.Q., 1994	Fault Diagnosis	RB	Self-Learning ES	Self Learning	OBSL Algorithm: MR, MF
PCDIASHOOT	Sidek, 2000	Trouble-shooter	RB	Rapid prototype	-	Forward chaining with depth first search
A self-learning expert system for diagnosis in traditional Chinese medicine	Wang X. et al., 2004	Medicine	RB	Neural Network	Self Learning	Hybrid Bayesian network learning algorithm, Naı¨ve–Bayes classifiers with a novel scorebased Strategy
Decision Support System for Beijing Traffic Managemnet	Chen Y.S. et al., 2005	Traffic Management	Hybrid	Fuzzy Logic	Self-Learning ES	Dynamic Traffic Assignment technology
Cow Disease Diagnoses Expert System	Liu L.J. et al., 2005	Medicine	RB	Neural Network	Self-Learning	BP Network
Frame Based Object Relational	Rattanaprateep C. and Chittayasoth	-	RB	-	Non-Learning	Frame Representation

Database System	orn S., 2007					
Self – Developing Fuzzy Expert System for Optimization of Machining Process	Iqbal A. et al., 2008	Machinery	RB	Fuzzy Logic	Self-Learning, Self-Correcting, Self-Expanding	Auto development of Fuzzy sets and automatic updation of interface
Expert System to Increase Agriculture Production and Water Conservation	Mahmoud M., 2008	Agriculture	-	-	-	Verification, Validation and Evaluation
Expert System Shell for Fault Diagnosis and Control of Power System Equipment	Jain B. et al., 2008	Power System	RB	-	Web Based	Shell Development
ES for troubleshooting car problems	Verma et. al, 2010	Trouble-shooter	RB	Syntactic Unification for pattern Matching	-	Backward Chaining
Self-Developing Fuzzy Expert System for Manufacturing Domain	Iqbal A. et al., 2010	Manufacturing Domain	RB	Fuzzy Logic	Self – Developing	-
Expert System for Material Layer Depth of Chain Grate	Xu S. and Jing Y., 2011	Manufacturing Domain	RB	Self-Learning	Self-Learning	Q-Learning Algorithm

Muhadith: A Cloud Based Distributed Expert System for Classification of Ahadith	Bilal K. and Mohsin S., 2012	Religious Scripture	RB	-	Cloud Based	Service Oriented architecture
Fuzzy Expert System for Construction Labor Productivity Estimation	Muqeem S. et al., 2012	Labor Productivity Estimation	RB	Fuzzy Logic	-	Relative Importance Index
ShellAg: Expert System Shell for Agricultural Crops	Islam S.N., 2013	Agriculture	CB	-	-	Shell Development

*RB: Rule Based, CB: Case Based, LT: Learning Technology, Algo: Algorithm

It can be observed that the number of expert systems are developed for helping people from varied domains like medical, agriculture, power supply, labor productivity estimation and in religious scriptures too. Most of the systems are rule based while some used the hybrid approach using both rule based and case based knowledge acquisition. A number of systems have the self-learning capability that makes them to perform in a better way as compared to their previous attempts. Different learning techniques are used like fuzzy logic, neural network etc. while some expert systems have used the hybrid approach. Few systems are shell based systems that helped in reducing the man effort in developing the expert system as only knowledge base needs to be changed.

In consideration with this literature survey it can be concluded that troubleshooter expert system developed so far are not self learning. Self-learning systems are much more efficient as compared to the non-learning systems in terms of accuracy, user satisfaction, etc. So, a self-learning expert system for troubleshooting hardware and network problems for a Personal Computer (PC) is developed which is elaborated in the next section.

Chapter 3

Problem Statement

Today as we all know there is huge advancement in the IT industry and with its tremendous growth people's dependence on computers is increasing day by day. As presence of computers is the basic requirement to settle any IT sector, it is always necessary that they always work smoothly without any trouble to the user.

But once computers are developed they also need their maintenance. It is not that they always run as if they were new. So, users face different kind of problems like hardware, software, network, hard drive and many more problems with their PC. And once they get problems usually technicians are contacted to repair them. This causes a bulk of pressure on the less number of technicians in the world.

So, in consideration with the number of problems faced by the users regarding their PC's and the work load on the technicians, this learning system called Self learning Expert PC Troubleshooter (SLEPCT) is made. This will help users to solve their PC related problems by themselves on their desk itself without contacting any technician. And directly it will reduce the work load on the technicians as computer fault diagnosis is automated here.

In the existing work that has been already done there exists no learning troubleshooter that can improve by itself. The given system has been embedded with a hybrid learning algorithm that helps it to enhance its performance by increasing the user satisfaction level. As the system is used it keeps on modifying its knowledgebase by removing the data that is least accepted by the users. Also, the system features a mailing facility to mail the replies to the user's queries to their mail ids. None of the existing systems has this facility that can provide help to users by providing them solutions on their mail ids which makes it easy for users to tackle their problem without any trouble.

Objectives that are covered for the work done are:

- System should be capable of troubleshooting all hardware and network problems that occur in a PC. Exhaustive search on all the hardware and network faults is done.
- System provides a sequence of solutions for the same query so that user is highly satisfied.
- A hybrid learning approach is embedded in the system which helps in calculating the acceptance factor of each solution and helps in providing the solutions in a sequence of their decreasing value of acceptance factor.
- System should always work to maximize the user satisfaction and keeps on learning to achieve this goal.
- Mailing module of the system should always support the system so that user can be given the asked information as soon as possible and helps in keeping the system updated.
- Information which is not acceptable by any user need to be eliminated from the system. System should be able to remove that information to reduce the memory in use.

Chapter 4

Design and Implementation of a Learning System that Diagnoses PC Problems

From the last few years fault diagnosis has achieved a lot of attention in research literature. Several expert systems have been developed which can help in automating the diagnostic process of electronic devices. So, in consideration with the number of problems faced by the users regarding their PC's and the work load on the technicians, a self learning expert system called Self learning Expert PC Troubleshooter (SLEPCT) is designed. This will help users to solve their PC related problems by themselves. And directly it will reduce the work load on the technicians as computer fault diagnosis is automated here.

Generally expert systems use human knowledge stored as rules in the PC to solve problems that generally require human intelligence. This expert system also stores number of generally occurring faults in a PC categorized under their category and providing number of solutions to repair that fault. A hybrid learning algorithm is embedded in the system which leads the system to improve itself and provide most accurate solutions for the problems occurring.

A user friendly user-interface, a vast knowledge-base, an inference engine with the learning algorithm, and an interface for expert has been included in the expert system designed here. Additionally, the system features a mailing server that can post solutions to the queries of the users to their e-mail identities. The system is basically meant to automate the maintenance, repair, and operation process of the expert handling queries related to PC.

In this chapter the design and implementation of the learning system that troubleshoots and diagnoses PC problems is covered which can be used by different users to tackle their problems which they are facing with their PCs. The system will diagnose only

computer hardware and network related faults. The system is a knowledge-based expert system where information in the form of rules is stored. The system keeps on improving itself by taking feedbacks from the user and hence, increases the user satisfaction by providing the most suitable and accepted solutions to the users. The system is implemented using ASP.NET 4.0 and C#.NET 4.0 under the MS.NET Framework 4.0 and MS Visual Studio 2010. The rule-base is implemented as a relational database using MYSQL. The chapter covers the architectural design of the system, working of the system along with the screenshots and the implementation details of the system is covered.

4.1 Architecture and Working of the Expert Troubleshooter for PC

The given learning system for troubleshooting PC faults comprises of four major modules: a graphical user interface which help the users to interact with the ES developed, a knowledgebase which stores all the troubleshooting information in encoded form, an inference engine that is the brain of the system which does all the matching to provide solutions to the faults generated and applies learning algorithm for better results so that users get highly satisfied an admin interface which helps in updating, modifying, adding and deleting of information from the knowledgebase and sending mails to the users regarding their queries. The nominated system helps in automatic fault detection and decision making that how that fault can be removed. From a business point of view, the system will help in reducing the maintenance costs of a PC, will provide faster problem resolution, reduction in employee training time, and will deliver higher job quality. Block diagram of the system is given as:

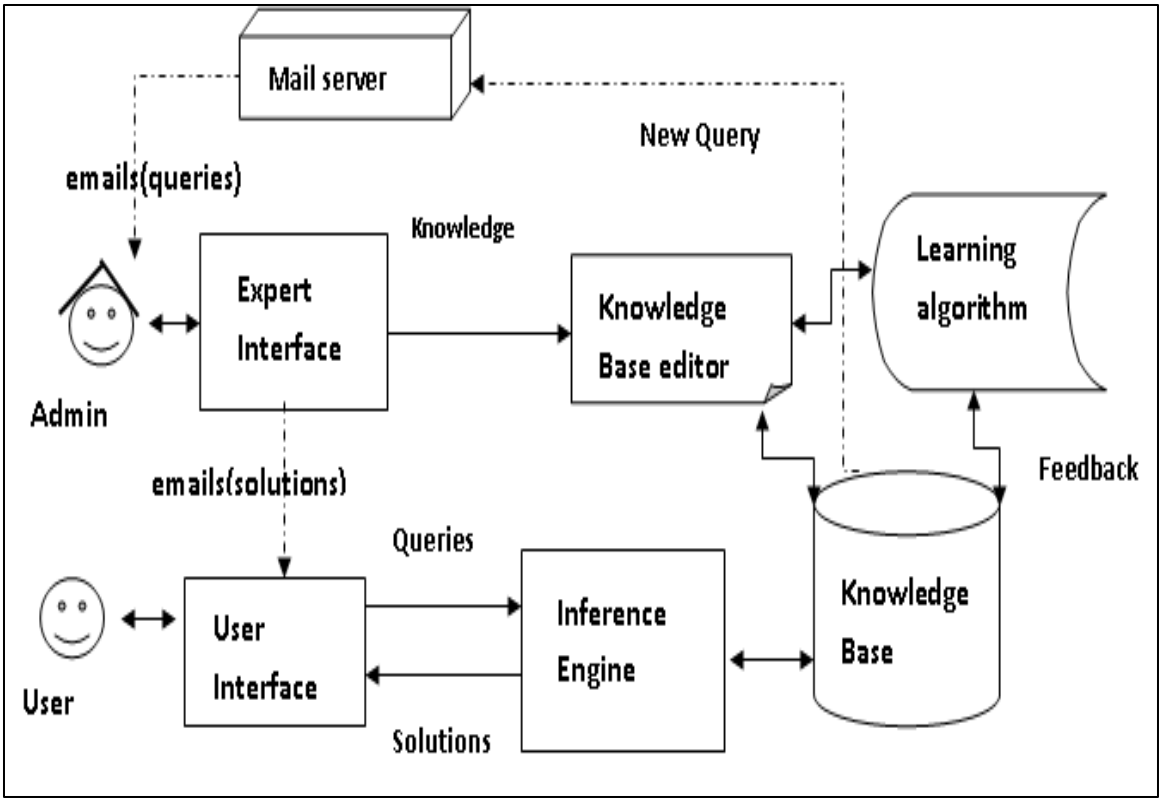

Fig 4.1 Block diagram of expert hardware and network troubleshooter for PC

4.1.1 The User Interface

The user interface of the given troubleshooter system provides bi-directional communication between the user and the system. It is akin to a step-by-step diagnostic questionnaire containing column asking for the category of the problem the user is facing like whether its mouse, keyboard, webcam, hard drive or some other problems. After such a questionnaire system will diagnose the problem and will provide number of solutions to the user. User need to give back its feedback whether the information provided by the system was helpful for him or not. If not, he will be provided with some other better solutions by the admin on his mail id using the system only.

Figure 4.2 provides a sample snapshot of the user interface where a solution to user's query regarding the repeated shutdown of the system is provided by the system:

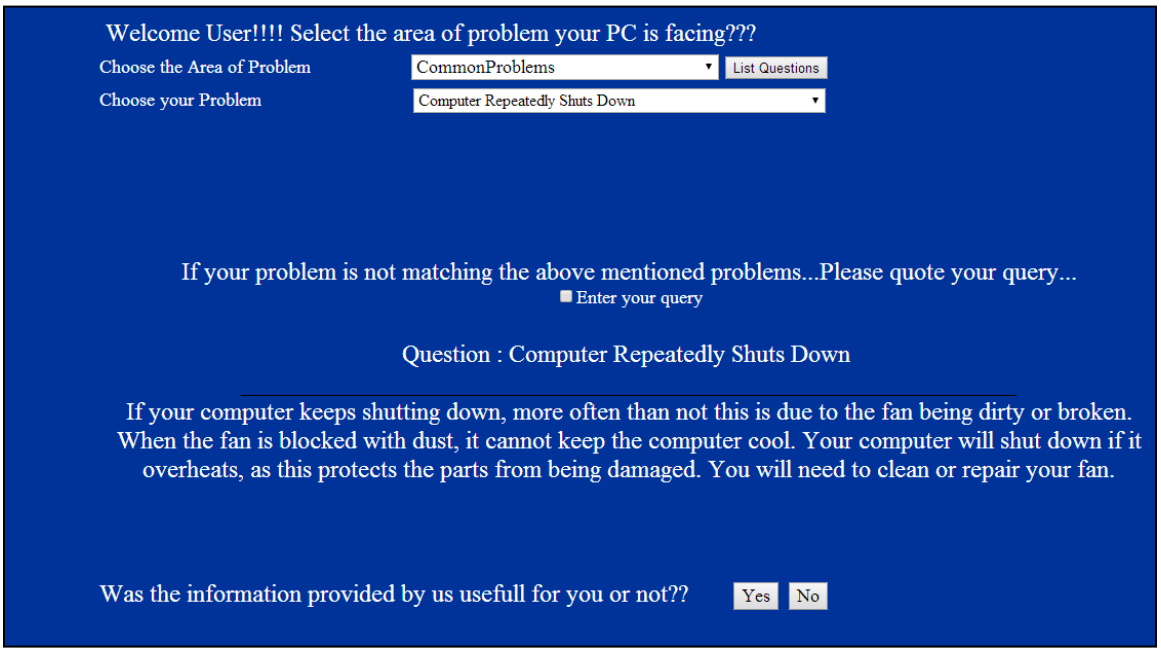

Fig 4.2 Sample snapshot for the user interface providing solution to user

Figure 4.3 shows the sample snapshot where user is asked for his mail id:

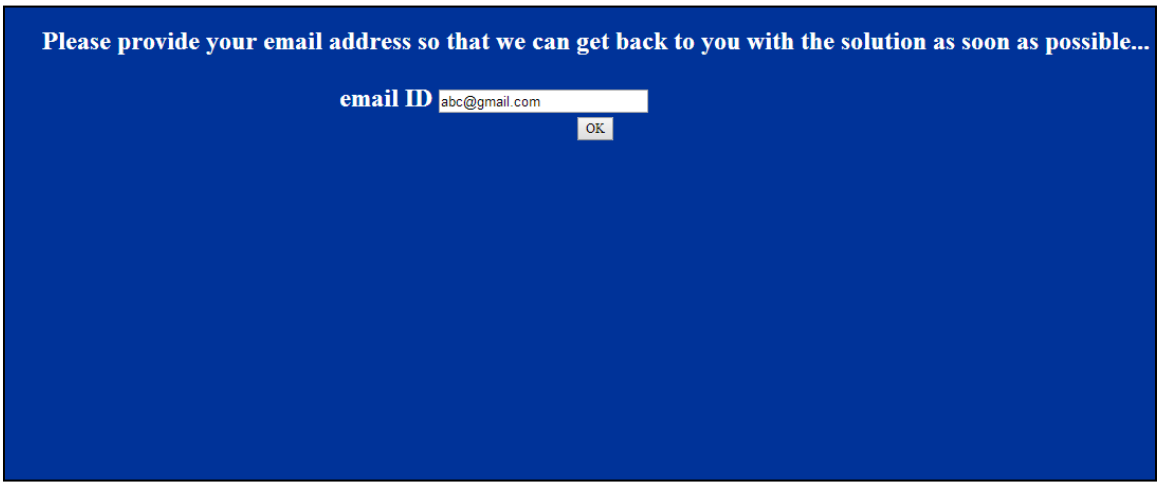

Fig 4.3 Sample snapshot for the user interface asking user for his mail id

4.1.2 The Knowledge-Base

The knowledge-base stores all the troubleshooting related information gathered from the human experts. It is basically a data storage where knowledge can be gathered, organized, saved for further reference and then can be accessed for use. While reasoning ES transfers facts from the knowledgebase to the working memory and does matching of the user given facts against the knowledge present in the knowledge base. Once matching is done, fault is identified and appropriate solution is given to fix that fault. Fig 4.4 and fig 4.5 shows a subset of the facts stored in the database:

id	question	solutions	category	countsol	prob
1	Computer is no...	Insert Operatin...	CommonProbl...	21	0.4883721
2	Computer is no...	Remove the Ra...	CommonProbl...	8	0.1951219
7	Windows Oper...	Start Menu -> ...	CommonProbl...	0	0
8	Windows Oper...	Check the CPU ...	CommonProbl...	0	0
5	Computer is no...	Try to start the ...	CommonProbl...	3	0.06818182
6	Computer is no...	Check the Basic...	CommonProbl...	6	0.1428571
9	g too long to st...	Remove unnee...	CommonProbl...	3	0.25
10	Windows Oper...	Try to uninstall ...	CommonProbl...	4	0.5714286
11	Windows Oper...	Ensure that you...	CommonProbl...	12	0.5714286
12	How do I locate...	Select "Find" fr...	CommonProbl...	0	0
13	How do I locate...	How do I locate...	CommonProbl...	0	0
14	How to clean u...	Use Scandisk. T...	HardDrive	0	0
15	How to clean u...	Uninstall unnec...	HardDrive	0	0
16	How to clean u...	Use Disk Defrag...	HardDrive	0	0
17	How to clean u...	Delete tempora...	HardDrive	0	0
18	How to clean u...	Disk Clean up. ...	HardDrive	0	0
19	Hard Drive is n...	If your hard dri...	HardDrive	0	0
20	How to clean u...	Use Scandisk. T...	HardDrive	0	0

Fig 4.4 Knowledgebase of the system showing queries and their solutions with calculated acceptance factors.

id	question	countques
1	Computer is no...	44
2	Windows Oper...	21
3	How do I locate...	0
4	How to clean u...	0
5	Hard Drive is n...	0
6	Username and ...	0

Fig 4.5 Table in database showing the count of each query

4.1.3 The Inference Engine

It is the brain or master mind of the ES which does all the matching of facts and rules and applies problem solving strategies to obtain answers and provide the conclusions. It produces the results of the faults that how they occur and provides the possible solutions for that and the reasoning details are not known to the user. The main job of the inference engine is the matching of facts provided by the user which are stored in the working memory with the information present in the knowledgebase to determine which rule is applicable in the given scenario and then to provide the final conclusion and result too. A hybrid learning algorithm is applied here to make the system more efficient. Learning in the system is based on the feedback collected from the users of the system. Users provide the information whether the provided information was helpful for the user or not. Based on that the inference engine calculates the accuracy of each and every solution in the knowledge base regarding any query. Acceptance of any solution is directly proportional to the accuracy of that solution which is calculated as described below. Accuracy level of a solution in the knowledge base is calculated by using the given formula:

$$Ac = Ns/N_q$$

where Ac is the accuracy of a given solution for a particular query, Ns is the number of user getting satisfied by that solution for that query and N_q is the number of times that particular query has been fired.

Based on that accuracy the solutions regarding to that query are sorted so that the most accurate and the most acceptable solution is given on the first attempt of using the system which leads to higher user satisfaction.
A threshold value is fixed for the accuracy Ac for each solution. If the Ac goes below that threshold value for a particular answer then that particular solution is dumped to maintain the performance of the system.

Below is the pseudo-code of the algorithm used by the inference engine of the proposed Expert PC Troubleshooter.

Step 1: Read initial category of the fault from the user and store it into working memory.
Step 2: Check the particular query of that particular category in the knowledge-base
Step 3: If all the conditions are matched, fire the rule.
Step 4: If the category or the particular query is not listed then email id of the user is stored in the database.
Step 5: Admin gets a notification of the query on its mail id.
Step 6: Count of the unanswered queries increases by 1 for each new query.
Step 7: Admin can answer those queries and new facts are stored in the knowledge base.
Step 6: Solutions to the user's queries are mailed to their respective mail ids using the mail server in the system.

Focusing on improvement in performance in terms of increased user satisfaction using learning, a supervised learning algorithm has been used for implementing this. Pseudo code for the learning algorithm that is shown in figure 4.4:

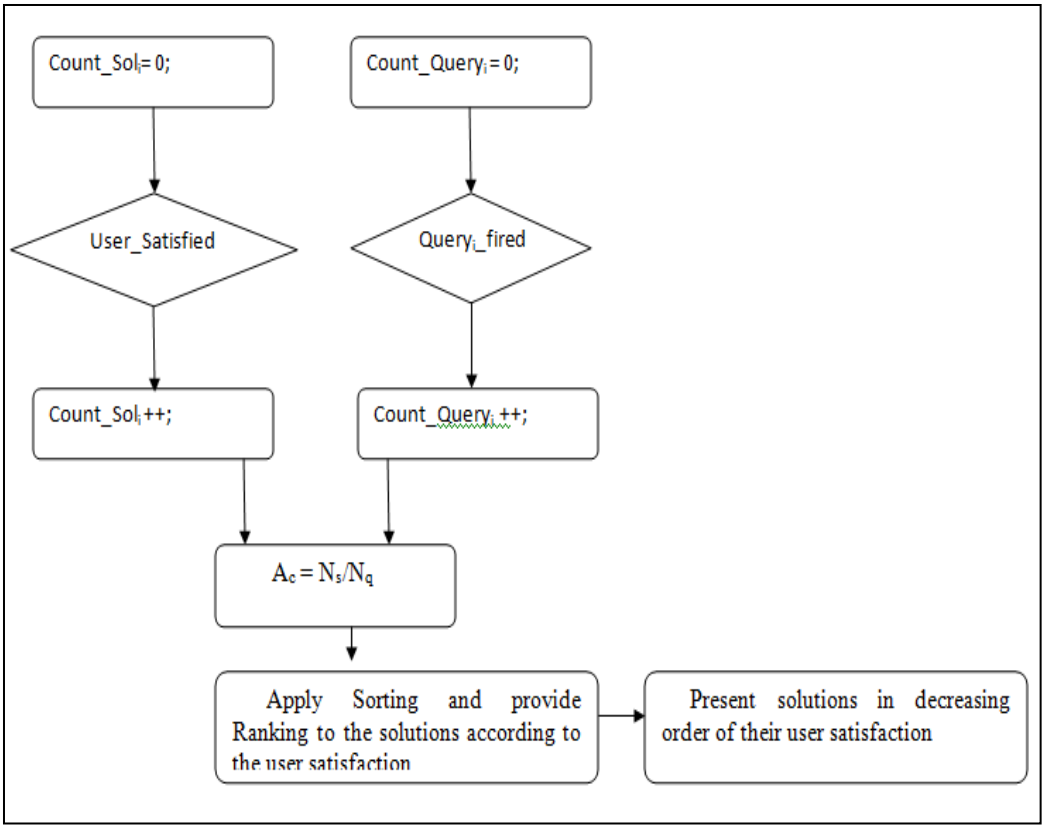

Fig 4.6 Flowchart showing the learning algorithm used by the system

Figure 4.7 shows the flowchart of the system showing the control flow in the system:

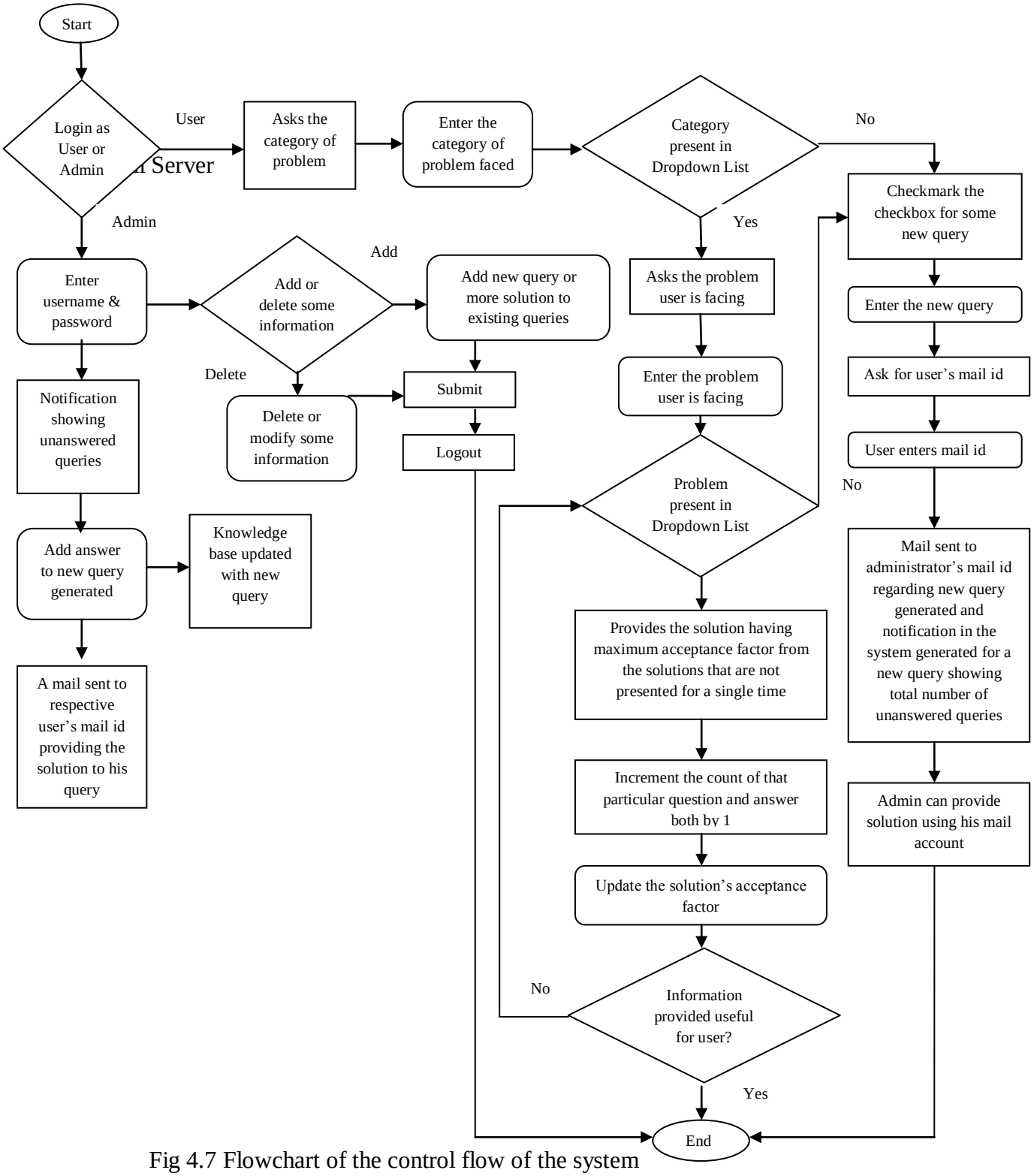

Fig 4.7 Flowchart of the control flow of the system

4.1.4 Mail Server

Mail server is basically used to send mails. Here, Simple Mail Transfer Protocol (SMTP) is used for providing the mailing facilities. SMTP provides Internet standard for e-mail transfer across the networks. The protocol is in widespread use today. Amongst the 7 layers of the OSI model, it is an Application Layer protocol in the OSI reference model. Here, mail server is basically used to send notifications about the new query to the admin and then after that it helps in sending the solutions of the user's queries to their respective mail ids. Whenever there comes a query for which the user is not satisfied by the solutions provided by the system, system asks the user for his mail identity. For each mail identity entered in the system, admin gets a notification about the new query generated on his mail identity. And simultaneously a notification is provided by the system to the admin whenever he accesses the system. So, after getting that notification admin can use the system to reply to the user with proper solutions to the query given by the user. Reply by the admin will be sent to the user to his mail identity and also the same reply will be stored in the database corresponding to the query generated. Hence, this section of the system also helps in updating the system with the new queries and their corresponding solutions. Hence, system keeps on upgrading itself with new knowledge.

4.1.5 The Expert Interface

The expert interface is the interface that is used by the admin of the system. It basically aims at managing the knowledgebase and helps the admin to review the existing knowledge and to add, delete, modify or update the present information. In addition to this admin answers the new queries asked by the users and can send mail to their mail ids.

Figure 4.6 shows the admin interface showing that there are 10 unanswered queries which the admin needs to reply for and simultaneously new data entered by the admin is shown:

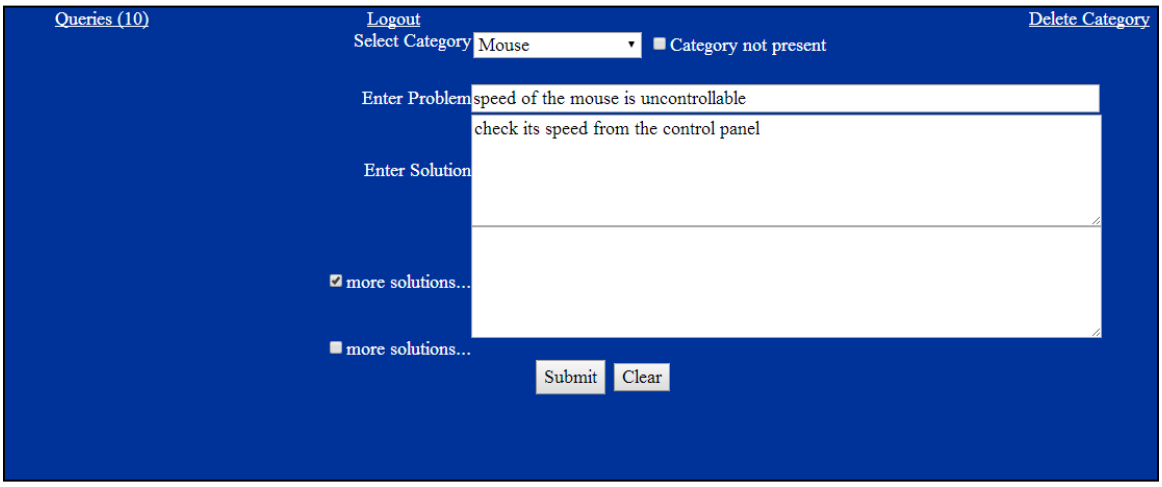

Fig 4.8 Admin interface of the system

4.2 Implementation

The learning system is implemented using ASP.NET 4.0 and C#.NET 4.0 under the Microsoft.NET Framework 4.0 and Visual Studio 2010. MYSQL is used as a relational database to generate the rule base for the system.

Chapter 5

Experimental Results

5.1 Performance of the Learning System

Performance of the system is measured by the user satisfaction level here which is calculated by the Acceptance factor given by A_c which is calculated by the ration of the number of times a solution is accepted and number of times that particular query is fired. As the system is used for fault diagnosing its performance goes on increasing as is shown in the following experimental data. Table 5.1 shows the first experimental data calculated for the category: "Common Problems" and query: "Windows operating system is taking too long to start" :

Table 5.1 Experimental Data 1

CATEGORY		Common Problems										
QUERY		Windows operating system is taking too long to start										
ITERATION	**SOL. SELECTED**	**N_S A1**	**N_S A2**	**N_S A3**	**N_S A4**	**N_S A5**	**N_Q**	**A_C A1**	**A_C A2**	**A_C A3**	**A_C A4**	**A_C A5**
1	A1	1	0	0	0	0	1	1	0	0	0	0
2	A2	1	1	0	0	0	2	0.5	0.5	0	0	0
3	A1	2	1	0	0	0	3	0.67	0.33	0	0	0
4	A3	2	1	1	0	0	4	0.5	0.25	0.25	0	0
5	A3	2	1	2	0	0	5	0.4	0.20	0.4	0	0
6	A1	3	1	2	0	0	6	0.5	0.16	0.33	0	0
7	A1	4	1	2	0	0	7	0.57	0.14	0.28	0	0
8	A1	5	1	2	0	0	8	0.625	0.125	0.25	0	0
9	A1	6	1	2	0	0	9	0.66	0.11	0.22	0	0
10	A4	6	1	2	1	0	10	0.6	0.1	0.2	0.1	0
11	A2	6	2	2	1	0	11	0.54	0.18	0.18	0.09	0
12	A1	7	2	2	1	0	12	0.58	0.16	0.16	0.08	0
13	A1	8	2	2	1	0	13	0.61	0.15	0.15	0.07	0
14	A1	9	2	2	1	0	14	0.64	0.14	0.14	0.07	0
15	A2	9	3	2	1	0	15	0.6	0.20	0.13	0.06	0
16	A1	10	3	2	1	0	16	0.625	0.18	0.125	0.06	0
17	A1	11	3	2	1	0	17	0.647	0.176	0.117	0.058	0
18	A1	12	3	2	1	0	18	0.67	0.16	0.11	0.055	0
19	A1	13	3	2	1	0	19	0.684	0.15	0.105	0.052	0
20	A1	14	3	2	1	0	20	0.7	0.15	0.1	0.05	0

*A1: Check the CPU usage by pressing "Ctrl + Alt + Del" -> Task Manger -> Performance Tab -> If the CPU percentage in use is more than 25%, there is a possibility of viruses presence on the PC. Run an anti-virus scan for scanning of malicious content.

A2: Try to uninstall larger software and servers (if any) that are not necessary from the Control Panel -> Add/ Remove programs.

A3: Remove unneeded icons on your desktop, install a firewall, install antivirus and Anti spyware tools, schedule regular registry scans.

A4: Insert Operating system and then choose to repair the operating system from the CD

A5: Ensure that you perform correct uninstall (or install) of all applications. In case you are prompted about a shared DLL being removed (or overwritten) ensure that this DLL is not used by any other program on the system. Also, if a DLL is being overwritten, ensure that you are not overwriting a more current version of DLL with an outdated version.

Here, it is observed that when the first time the query is fired solution by which user got satisfied is A1 and for the second user satisfactory answer was A2 and so on. At the first attempt the acceptance factor or accuracy factor of all answers is calculated which came out to be 1 for A1 and 0 for others. At each attempt acceptance factor is calculated for all solutions. The solutions are projected to the user in the decreasing order of their acceptance factor. It can be explained as in the 9^{th} iteration solution 1 i.e. A1 has the highest acceptance factor that is 0.66, this means that this will be projected to the user first. If he is not satisfied by this solution then the solution 3 i.e. A3 having second highest acceptance factor i.e. 0.22 is projected. If the user is still not satisfied by the answers given then the solution 2 i.e. A2 will be projected i.e. having acceptance factor 0.11 will be given. In the whole iterations A5 is not selected by any user and its acceptance factor remains 0 even after 20 iterations. At its 21^{st} iteration this solution will be deleted from the knowledgebase as this is the answer which is not accepted by any of the user. By the experimental data it is seen that A1 has got the maximum acceptance level, so this solution will be projected at the first attempt only and maximum number users are getting satisfied with this solution only. Hence, the satisfaction level of users increases as they get the required answers in their first attempt only.

Another experimental data for category: “Mouse” and query: “Moving the mouse does nothing” is given in table 5.2:

Table 5.2 Experimental Data 2

CATEGORY		Mouse										
QUERY		Moving the mouse does nothing										
ITERATION	SOL. SELECTED	N_S A1	N_S A2	N_S A3	N_S A4	N_S A5	N_Q	A_C A1	A_C A2	A_C A3	A_C A4	A_C A5
1	A2	0	1	0	0	0	1	0	1	0	0	0
2	A2	0	2	0	0	0	2	0	1	0	0	0
3	A2	0	3	0	0	0	3	0	1	0	0	0
4	A4	0	3	0	1	0	4	0	0.75	0	0.25	0
5	A1	1	3	0	1	0	5	0.20	0.6	0	0.20	0
6	A5	1	3	0	1	1	6	0.167	0.5	0	0.167	0.167
7	A2	1	4	0	1	1	7	0.142	0.57	0	0.142	0.142
8	A2	1	5	0	1	1	8	0.125	0.625	0	0.125	0.125
9	A2	1	6	0	1	1	9	0.11	0.66	0	0.11	0.11
10	A3	1	6	1	1	1	10	0.1	0.6	0.1	0.1	0.1
11	A2	1	7	1	1	1	11	0.09	0.63	0.09	0.09	0.09
12	A2	1	8	1	1	1	12	0.083	0.67	0.083	0.083	0.083
13	A2	1	9	1	1	1	13	0.076	0.69	0.076	0.076	0.076
14	A2	1	10	1	1	1	14	0.071	0.71	0.071	0.071	0.071
15	A2	1	11	1	1	1	15	0.066	0.73	0.066	0.066	0.066
16	A2	1	12	1	1	1	16	0.0625	0.75	0.0625	0.0625	0.0625
17	A2	1	13	1	1	1	17	0.058	0.76	0.058	0.058	0.058
18	A1	2	13	1	1	1	18	0.11	0.72	0.055	0.055	0.055
19	A2	2	14	1	1	1	19	0.105	0.73	0.052	0.052	0.052
20	A2	2	15	1	1	1	20	0.1	0.75	0.05	0.05	0.05

*A1: Try the mouse on another computer to verify it is not defective.

A2: The USB connection may be loose. Try unplugging the mouse and plugging it back in.

A3: If your mouse is plugged into a USB hub or USB switch box, try removing it and plugging it directly into the computer.

A4: Plug the mouse into a different USB port.

A5: Unplug other USB devices that are not in use.

Table 5.3 shows the first experimental data calculated for the category: “Keyboard” and query: “Keyboard does not seem to be responding” :

Table 5.3: Experimental Data 3

CATEGORY		Keyboard										
QUERY		Keyboard does not seem to be responding										
ITERATION	SOL. SELECTED	N_S A1	N_S A2	N_S A3	N_S A4	N_S A5	N_Q	A_C A1	A_C A2	A_C A3	A_C A4	A_C A5
1	A1	1	0	0	0	0	1	1	0	0	0	0
2	A2	1	1	0	0	0	2	0.5	0.5	0	0	0
3	A2	1	2	0	0	0	3	0.33	0.67	0	0	0
4	A1	2	2	0	0	0	4	0.5	0.5	0	0	0
5	A3	2	2	1	0	0	5	0.4	0.4	0.2	0	0
6	A3	2	2	2	0	0	6	0.33	0.33	0.33	0	0
7	A4	2	2	2	1	0	7	0.285	0.285	0.285	0.142	0
8	A3	2	2	3	1	0	8	0.25	0.25	0.375	0.125	0
9	A3	2	2	4	1	0	9	0.22	0.22	0.44	0.11	0
10	A3	2	2	5	1	0	10	0.2	0.2	0.5	0.1	0
11	A5	2	2	5	1	1	11	0.18	0.18	0.45	0.09	0.09
12	A3	2	2	6	1	1	12	0.16	0.16	0.5	0.083	0.083
13	A3	2	2	7	1	1	13	0.153	0.153	0.53	0.076	0.076
14	A3	2	2	8	1	1	14	0.142	0.142	0.571	0.071	0.071
15	A3	2	2	9	1	1	15	0.133	0.133	0.6	0.067	0.067
16	A3	2	2	10	1	1	16	0.125	0.125	0.625	0.0625	0.0625
17	A3	2	2	11	1	1	17	0.11	0.11	0.647	0.058	0.058
18	A3	2	2	12	1	1	18	0.11	0.11	0.67	0.055	0.055
19	A3	2	2	13	1	1	19	0.105	0.105	0.684	0.052	0.052
20	A3	2	2	14	1	1	20	0.1	0.1	0.7	0.05	0.05

* A1: If you are using a wireless keyboard, troubleshoot the wireless connection.

A2: Try the keyboard on another computer to verify it is working correctly.

A3: Unplug other USB devices that are not in use.

A4: Plug the keyboard into a different USB port.

A5: If your keyboard is plugged into a USB hub or USB switch box, try removing it and plugging the keyboard into the computer.

5.2 Comparison of the learning system with the system not having learning algorithm

It's observed that the system with learning ability has an edge above the system without the learning ability. After number of runs of the system by different users for the same query the solution with which they get satisfied is shown in the graphs given below:

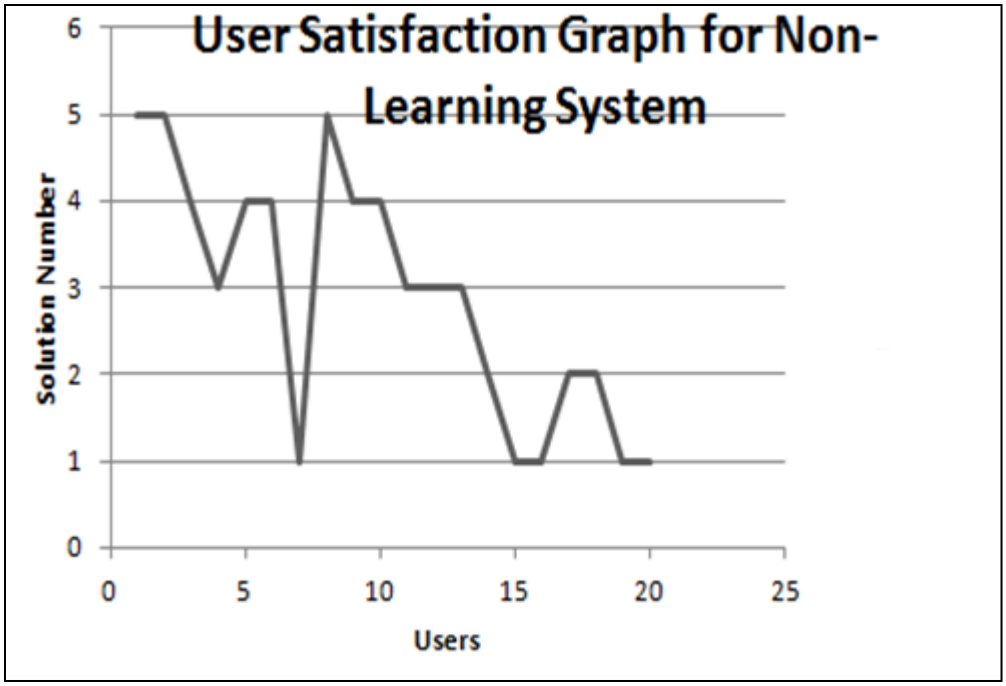

Fig 5.1 User Satisfaction Graph for Non-Learning System

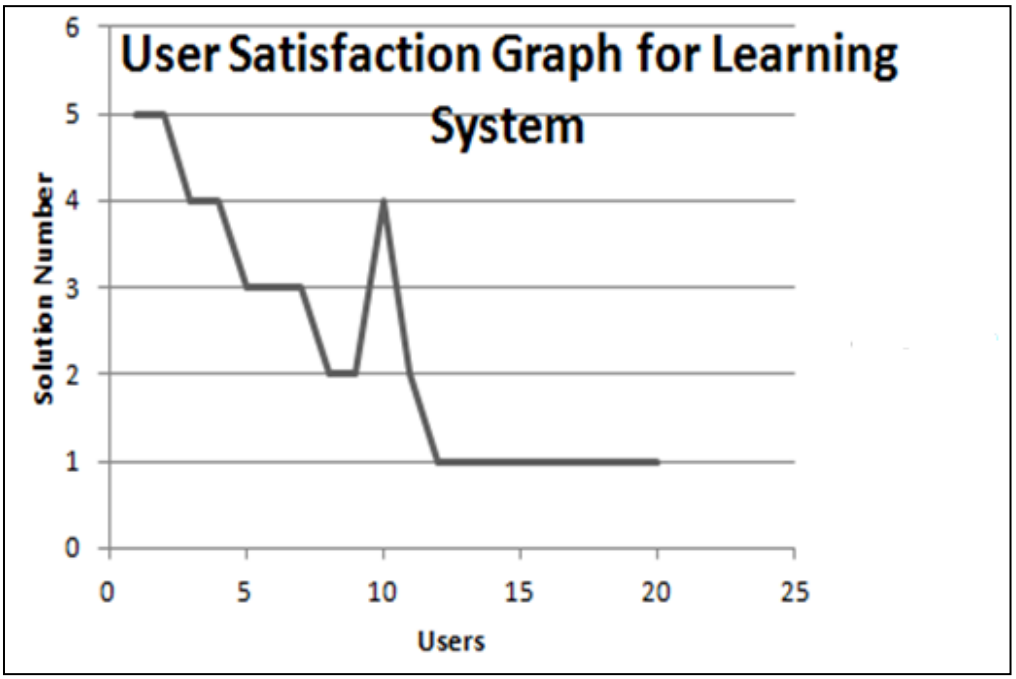

Fig 5.2 User Satisfaction Graph for Learning-System

It's very much clear that the system with the learning algorithm embedded is more efficient as it provides the most accepted answer at its first attempt which leads to higher user satisfaction.

In the system with the learning algorithm embedded user satisfaction is higher as there are more number of users that get satisfied with the first answer that is provided by the system as compared to the system without having learning algorithm embedded in it. Also in the learning system as the number of times the same query is fired, satisfaction level of the users tends to increase as they get satisfied with the first answer only as system improves itself on its each attempt.
Table 5.4 given below shows the experimental data showing the number of users out of 20 getting satisfied with different solutions that are provided with the system having learning algorithm and without the learning algorithm:

Table 5.4 Comparison of number of user satisfied with the system with and without learning algorithm

Solution Number	**Number of users without Learning Algorithm**	**Number of users with Learning Algorithm**
Solution 1	4	9
Solution 2	3	3
Solution 3	5	3
Solution 4	5	3
Solution 5	3	2

Figure 5.3 given below shows that the number of users satisfied by the first prompted answer is more when the learning algorithm is embedded in the system. It shows the graphical representation of the data collected in above given table. Hence, user satisfaction level is enhanced using the learning technique in the system.

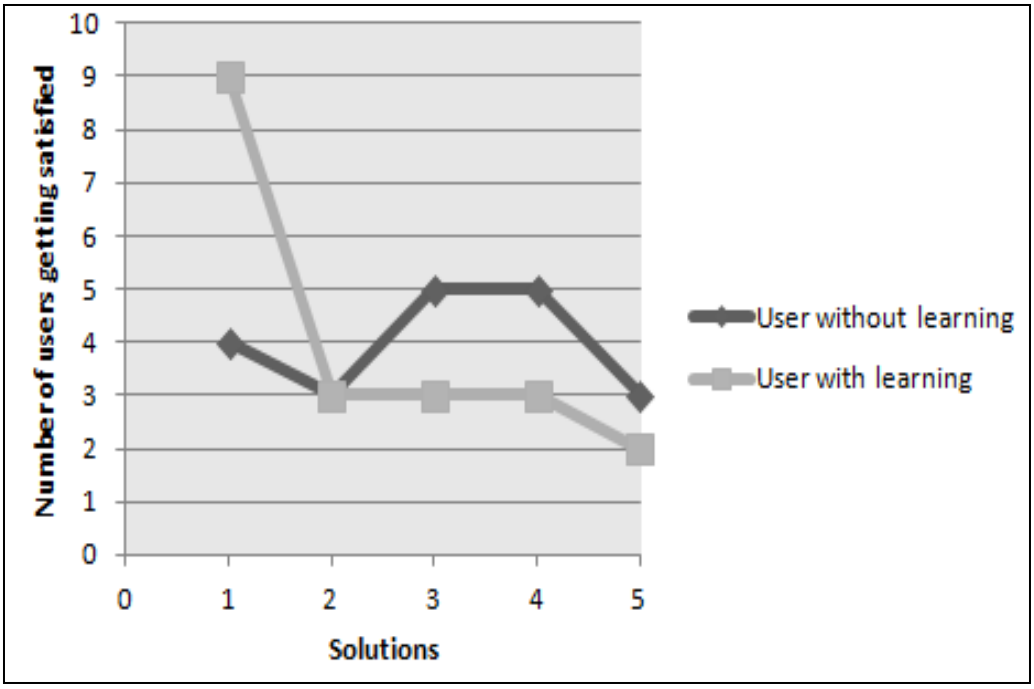

Fig 5.3 Line Graph showing the satisfaction level of users with the system with and without the learning algorithm embedded in it

Chapter 6

Conclusion and Future Scope

In this thesis work a learning system called Self learning Expert PC Troubleshooter (SLEPCT) for troubleshooting and solving PC problems and faults is given. This system helps in troubleshooting and diagnosing hardware and network problems of a PC. It extracts data from the knowledgebase using the inference engine where hybrid learning algorithm is embedded that provides the results based on the acceptance factor. Acceptance factor is calculated using the accuracy of each solution of the query which is based on the probability measures. This learning algorithm has been embedded to enhance the user satisfaction. It is the motive of the system to provide solutions to the users for the faults asked by them in such a sequence that the user gets satisfied with the first provided solution only and accuracy of the system goes on increasing with this iterative process. The system with the learning algorithm embedded is more efficient as it provides the most accepted answer at its first attempt which leads to higher user satisfaction as compared to the system where no learning technique is applied. In addition to this admin interface eases the managing of the system and updating and modifying the knowledgebase is also easy. Furthermore, the system features a mail server that helps admin to get notifications about the new queries generated and then admin can make responses to the new queries using that mail server. Likewise, the new queries from the users helps the system to stay up-to- date with the latest troubleshooting knowledge.

Considering the future work, knowledge base can be improved further with the latest and more number of PC faults covering all the exhaustive cases. More number of learning techniques can be embedded in the system to enhance its performance further. The system can be made available online to server large number of users. It can also be implemented as an android application that can serve the purpose for android operating system too.

References

1. J.C. Giarratano and G.D. Riley, "Introduction to Expert Systems," in Expert Systems Principles and Programming, 4th edition, pp. 1-20.
2. A. Miller, "The structure, history and future of successful AI applications", *IEEE Transactions on Software Engineering*, Vol. 15, No. 8, pp. 12-14, Aug 1989.
3. Artificial Intelligence [online]. Available: http://ai-depot.com/Intro.html
4. A. Iqbal, I. Khan, N.U. Dar and N. He, "A Self-Developing Fuzzy Expert System, Designed for Optimization of Machining Process''. *Proceedings of the World Congress on Engineering, London, U.K.*, Vol. 3, pp. 978-988, 2008.
5. S.M. Duggal, R. Chhabra, "Learning systems and their applications : Future of strategic expert system". *IACIS*, pp. 179-185, 2002.
6. E. Kheirkhah and A. Deraman, "Important factors in selecting Requirement Engineering Techniques", *Proceedings of International Symposium on Information Technology (ITSIM 2008)*, pp.1-5, 2008.
7. L. Jiang, A. Eberlein, B.H. Far, and M. Mousavi, "A Methodology for the selection of Requirement Engineering Techniques", *Software and Systems Modeling*, Vol.7, No.3, pp. 303-328, 2008.
8. J.K. Ang, S.B. Leong, C.F. Lee and U.K. Yusuf, "Requirement Engineering Techniques In Developing Expert Systems", *IEEE symposium on Computers & Informatics*, pp. 640-645, 2011.
9. A. Mavin, P. Wilkinson, A. Harwood, and M. Novak, "EARS (Easy Approach to Requirements Syntax)", *Requirements Engineering Conference, RE '09, 17th IEEE International*, pp. 317 – 322, 2009.
10. C.K. Tyran and J.F. George, "The implementation of expert systems: A survey of Successful Implementation", *ACM SIGMIS Database*, Vol. 24, No. 1, pp. 5-15, 1993.
11. B.H.C. Cheng, J.M. Attle., "Research Directions in Requirement Engineering", *Proceedings of IEEE Future of Software Engineering (FOSE'07)*, doi:0-7695-2829-5/07, 2007.

12. A.Kusiak And S.S. Heragu, “Expert Systems and Optimization”, *IEEE Trans. On Software Engineering*, Vol. 15, No. 8, Aug 1989.
13. F.S. Hillier and G.J. Lieberman, “Introduction to operations research”**,** Tata McGraw-Hill Education, 2001.
14. A. Kusiak, "Artificial intelligence and operations research in Flexible manufacturing systems," *INFOR: Information Systems and Operational Research*, Vol. 25, No. I, pp. 2-12, Jan. 1987.
15. S.N.Islam, S. Kundu, J. Shoran, N. Sabir, K. Sharma, S. Farooqi, R. Singh, H.O. Agarwal, K.K. Chaturvedi, R.K. Sharma and A.K. Sharma, “Selection of wheat (Triticum aestivum) variety from expert system on Wheat” . *Industrial Journal of Agriculture Science* , Vol. 82, No. 1, pp. 39-43, 2012.
16. S.N. Islam, M.A. Khan , R.K. Kakani, K. Kant,O.P. Aishwath & M.M. Anwer , “Expert system for effective selection of seed spices varieties”. *International Journal of Seed Spices,* Vol. 1, pp. 88-94, 2011.
17. Devraj and R. Jain, ” An expert system for the diagnosis and control of diseases in pulse crops”. *Expert Systems with Applications*, Vol. 38, No. 9, pp. 11463-11471, 2011.
18. S.N. Islam., “ShellAg: Expert System Shell for Agricultural Crops”. *In proceedings International Conference on Cloud & Ubiquitous Computing & Emerging Technologies*, pp. 83-86, 2013.
19. M. Bohanec, “DEX: An Expert System Shell for Decision Support”. *Sistemica*, Vol. 1, No. 1, pp. 145–157, 1990.
20. R.S. James, R.W. Michael, and O.P. Marius, “AGNESS: A generalized network-based expert system shell”. *Proceedings of the 5th National Conference on Artificial Intelligence. Philadelphia, PA*, Vol. 2, pp. 996-1002, 1996.
21. K. Bilal and S. Mohsin, “ Muhadith : A Cloud based Distributed Expert System for Classification of Ahadith”. *In proceedings 10th International Conference on Frontiers of Information Technology*, pp. 73-78, 2012.
22. Y. Zeng, Y. Cai, P. Jia and H. Jee , “Development of a web based decision support system for supporting integrated water resources management in Daegu

city, South Korea". *Expert Systems with Applications,* Vol. 39, No. 11, pp. 10091-10102, 2012.

23. 23. M.B. Jain, A. Jain , M.B. Srinivas . "A Web based Expert System Shell for Fault Diagnosis and Control of Power System Equipment". *International Conference on Condition Monitoring and Diagnosis, Beijing, China,* pp. 1310-1313, 2008.
24. 24. L.J. Liu, Y.D. Wang, M.Z. Guo . "The research and application of the self-learning expert system based on BP network". *Proceedings of the Fourth International Conference on Machine Learning and Cybernetics, Guangzhou*, Vol. 7, pp. 4153- 4157, 2005.
25. 25. M. Minsky. "A Framework for Representing Knowledge". *Reprinted in The Psychology of Computer Vision, P. Winston (Ed.), McGraw-Hill,* 1975.
26. M. Negnevitsky, "Artificial Intelligence: A Guide to Intelligent Systems". *Addison Wesley, Harlow, England,* 2002.
27. R. Fikes and T. KehLer, "The Role of frame-based representation in reasoning". *Communications of the ACM,* Vol. 28, No. 9, pp. 904-920, 1985.
28. G. Luo and V.K. Chaudhri, "Implementing OKBC Knowledge Model Using Object Relational Capabilities of Oracle 8, Technical Report", pp. 1-22, 2013.
29. C. Rattanaprateep and S. Chittayasothorn, "A frame-based object relational expert database system". *In AFRICON* , pp. 1-7, 2007.
30. Y.S. Chen, H.J.V. Zuylen, R. Lee, "A self-learning-process based decision support system for Beijing Traffic Management". *Advanced OR and AI Methods in Transportation*, pp. 579-586, 2005.
31. C.S. Xu and Y.Q. Zhang, "Fault Diagnostic Expert System Based on Two Knowledge of Coal-Cutter". *Chinese Mine Automation*, Vol. 3, pp. l-4, 1994.
32. P. Yusong et al., "A Simulation Based Expert System for Process Diagnosis". *In Proceedings of EUROSIS 4th International Industrial Simulation Conference (ISC 2006), Italy*, pp. 393-398, 2006.
33. S. Xu and Y. Jing, "Expert Control System for Material Layer Depth of Chain Grate Based on Q_learning Algorithm", *Computer Science and Service System (CSSS), 2011 International Conference on*. IEEE, pp. 2241-2244, 2011.

34. 34. S. Muqeem , A. Bin Idrus ,M.F. Khamidi , Y.K. Siah, M. Saqib." Application of Fuzzy Expert Systems for Construction Labor Productivity Estimation". *International Conference on Computer & Information Science (ICCIS)* ,Vol. 1, pp. 506-5011, 2012.
35. H. Mao, "Estimating labour productivity using fuzzy set theory", Master Thesis, Civil and Environmental Engineering Dept., University of Alberta, Canada, pp. 128, 1999.
36. A. Fayek and A.O. Oduba, "Predicting industrial construction productivity using fuzzy expert systems". *Journal of Construction and Engineering Management,* Vol. 131, No. 8, pp. 938–941, 2005.
37. X. Wang, H. Qu, P. Liu & Y. Cheng, "A self-learning expert system for diagnosis in traditional Chinese medicine". *Expert Systems with Applications,* Vol. 26, pp. 557–566, 2004.
38. V. Ghaemi , "Knowledge-based troubleshooting and diagnosis of electronic devices". *Instrumentation and Measurement Technology Conference, IEEE IMTC-89*, pp. 299-302, 1989.
39. K. Gupta, "Knowledge-Based systems for troubleshooting complex equipment". *International Journal of Information and Computing Sciences,* Vol. 1, No. 1, pp. 30-31, 1998.
40. I. Sidek, "PC diagnosis and troubleshooting expert system based on computer response during power-on-self-test (POST) – PCDIASHOOT". *TENCON Proceedings,* Vol. 3, pp. 458-463, 2000.
41. Verma et al., "An Approach towards designing of Car Troubleshooting Expert System". *International Journal of Computer Applications,* Vol. 1, No. 23, pp. 107-114, 2010.
42. J. Romeo, G. Pajares, M. Montalv et al., " A new Expert System for greenness identification in agricultural images", *Expert Systems with Applications*, Vol.40, No.6, pp. 2275-2286, 2013.
43. T. Frantti, & M. Majanen, " An expert system for real-time traffic management in wireless local area networks", *Expert Systems with Applications,*Vol.41, No.10, pp.4996-5008, 2014.

LIST OF PUBLICATIONS

1. M. Agarwal, S. Goel, "Expert System and it's Requirement Engineering Process", In proceedings of IEEE International Conference on Recent Advances and Innovations in Engineering (ICRAIE-2014), May 09-11, 2014, Jaipur India, pp. 46, 2014 (is to be published).
2. M. Agarwal, S. Goel, "Self Learning Expert Systems: A Survey and Design", communicated (March 2014) in International Journal of Expert System with Application, Indexed in SCIE.