Contents

Introduction

Artificial intelligence is a branch of computer science that deals with intelligent machines and software. It mainly concerns with deep study and development phases of intelligent machines. Some authors also define Artificial Intelligence as "The study and design of intelligent agents, where intelligent agent is a system that perceives its environment and takes actions that maximize its chances of success". An expert system is also called knowledge based system. It is a computer program that contains some of the subject specific knowledge of one or more human experts.

The main components of expert system are knowledge base and inference engine. Knowledge base contains the domain knowledge needed to solve the problems in the form of rules. The rules are a popular paradigm for representing knowledge. Inference engine is the code at the core of the system which derives conclusions from knowledge base through inference or reasoning. The major features of expert system are user interface, data representation, inference, explanations, coping with uncertainty and advantages of expert system are fast response, increased reliability, reduced cost, reducing errors, multiple expertise, intelligent database, reduced danger.

Artificial intelligence systems are used in economics, medicines and military. These systems also have many software applications like video games, computer chess etc. The goals of artificial intelligence research are reasoning, knowledge, planning, learning, communication, perception and ability to manipulate objects Artificial intelligence is divided into two categories. These two categories are conventional artificial intelligence and computational intelligence. Conventional artificial intelligence includes machine learning and stastical analysis. Computational intelligence includes neural networks and fuzzy systems .The other applications of artificial intelligence are Automation, Computer Vision, Artificial Creativity, Expert System and Knowledge Management.

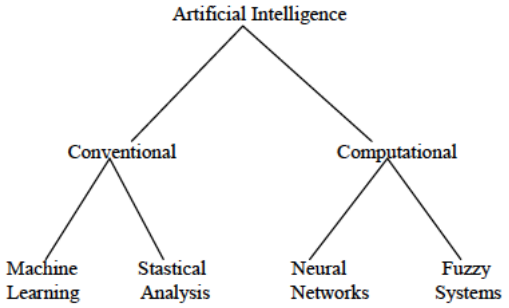

Figure 1: Components of Artificial intelligence

Expert System is one of the most common applications of artificial intelligence. It is a set of programs that manipulate knowledge to solve problems in a specialized domain that requires human expertise. An expert system represents information and searches for patterns in that information. They are known as expert systems because they model how a human expert analyzes a particular situation by applying rules to the facts (or compares the current case with similar cases) in order to reach a conclusion (Jimmy Singla, 2014). There are many applications of expert systems such as diagnosis, design, planning, financial decision making etc. Most applications of expert systems in medicine involve predicting, diagnosing and treating a particular disease (R.A. Soltan, 2013).

A large number of expert systems are medical. The main aim of any medical expert system is diagnosis and treatment of diseases. A medical expert system is built up of programs and medical knowledge base. The information obtained from medical expert system is similar to the information given by doctor or expert in that particular area (Jimmy Singla, 2014). Medical Knowledge of specialized doctor is vital for the growth of medical expert system. This knowledge is composed in two phases. In the first phase, the medical conditions of diseases are recorded during the formation of personal meeting with doctors and patients.

In the second phase, a deposit of rules is formed where each rule contains IF part that has the symptoms and THEN part that has the disease that should be realized (R.A. Soltan, 2013). Lungs are among the most important organs of the body, since they are vital for the process of breathing and respiration. Here, it is important to define and distinguish between breathing and respiration. Compact Oxford Dictionary defines respiration as: "(Biology) a process in living organisms involving the production of energy, typically with the intake of oxygen and the release of carbon dioxide from the oxidation of complex organic substances."

The definition of breathing, as given by Compact Oxford dictionary is: "air taken into or expelled from the lungs." Lung diseases basically hinder with the proper functioning of the mechanism of breathing. They might cause the person to become breathless in cases when the lungs get obstructed by mucus or other obstructive material and the person has to exert extra force in order to breathe. In some cases, the alveoli (air sacs) in the lungs get collapsed causing the patient to wheeze.

Patients who need to be treated by medical experts for the disease related to lung need to be physically present on the medical center and contact the doctor, and checked by physicians. But there are few numbers of specialist and medical centers along the globe, especially in developing countries. It is better to develop an intermediary tool that avoid/ trace the gap between shortage of experts and large number of patients who need expertise advise and treatment, beside wide spread establishment of medical center in the country. Here, Medical expert system that can treat lung disease of patients. Based on symptoms of different types of disease of lung can be treated by this expert system.

Design

This medical expert system is used to diagnose the main lung diseases among the patients. The diagnosis is made taking into account the symptoms that can be seen or felt. This medical expert system helps the doctor or expert in making the appropriate diagnosis of the patient. The lung diseases have many common symptoms and some of them are very much alike. This creates many difficulties for the lungs doctor to reach at a right decision or diagnosis. This expert system can remove these difficulties . This expert system is implemented in Prolog and java.

The main aim of any medical expert system is diagnosis and treatment of diseases. A medical expert system is built up of programs and medical knowledge base. The information obtained from medical expert system is similar to the information given by doctor or expert in that particular area. This medical expert system has ten lung diseases in its knowledge base. The user or patient is asked to answer with YES or NO, If a particular

symptom appears or not. In the end, based on user's or patient's answers, the name of the disease is displayed on the screen. A limitation of this medical expert system is that only symptoms entered by the programmer in the knowledge base are available. It does not think and learn by itself. Therefore the knowledge base can be updated any time with new symptoms and new diseases.

The main objective of this section was to identify, characterize, and define the problems the system will be expected to solve. The main problems identified include: Shortage of specialist obstetricians; the other medical staff in the Division needed expert knowledge and guidance, from the specialist obstetricians, on diagnosis and treatment of Lung patients. The users needed a system that could: Automate the medical protocols for diagnosis and treatment of lung patients; Offer training facility on the diagnosis and treatment for lung patients (J. Gudu, 2012).

The following system hardware, software, and human (end user) skills requirements are recommended so as to get the fastest, most reliable and upgradeable computer system that can diagnose lung patients efficiently and reliably (R.A. Soltan, 2013).. This was categorized as: Hardware requirements (a computer); Software requirements (Operating and application); what is expected of the user to have in order to use the system (such as medical skills and knowledge and basic computer skills). The system architecture diagram enables us to model the applications of a system graphically. It shows the interaction of users and system response, including the system development, development of knowledge question (J. Gudu, 2012).

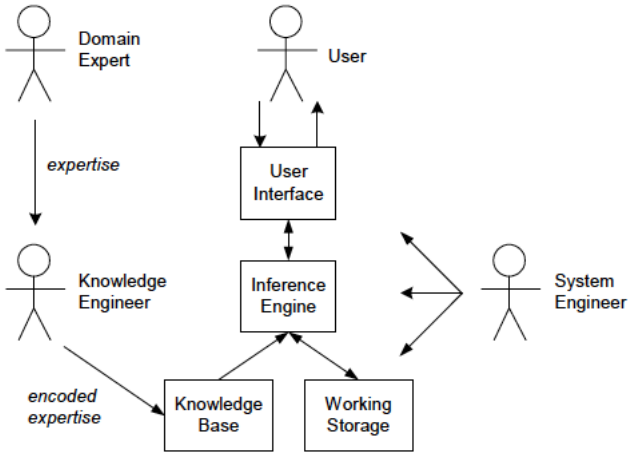

Figure 2: System architecture

The proposed rule based expert system has the following components- contains information about thirty-two lung diseases which are represented as a set of if-then production rules. It is analogue to the long term human memory. The total ordering of production rules is done in the knowledge base.

Working memory:

The working memory has conclusions reached before they are displayed to the user. The Fact Base contains facts which are used to match against the antecedent part of rules stored in the knowledge base. The fact base is analogue to the short term human memory.

The Inference Engine

As for the inference Engine, the user provides information about the problem to be solved and the system then attempts to provide insights derived (or inferred) from the knowledge base by examining the facts in the knowledge base . The system uses the forward chaining inference method (the process of starting with the facts and working forward to the conclusions [2, sited on 10]. The main task of Inference Engine is to carry out the reasoning by linking the rules with facts and deducing new facts.

The User Interface is used to communicate between user and expert system.

Knowledge base A declarative representation of the expertise, often in IF THEN rules.

To understand expert system design, it is also necessary to understand the major roles of individuals who interact with the system. These are:

Domain expert The individual or individuals who currently are experts solving the problems the system is intended to solve. **Knowledge engineer** The individual who encodes the expert's knowledge in a declarative form that can be used by the expert system.

User The individual who will be consulting with the system to get advice that would have been provided by the domain expert.

System engineer The individual who builds the user interface, designs the declarative format of the knowledge base, and implements the inference engine.

Implementation

The Knowledge Base contains information about ten lung diseases which are represented as a set of if-then production rules. The knowledge base is analogue to the long term human memory. The total ordering of production rules is done in the knowledge base, for examples Tuberculosis is a lung disease whose symptoms are persistent cough, constant fatigue, weight loss, loss of appetite, fever, coughing up blood, night sweats. So it will be stored in knowledge base in the form of a rule which is as follow:

Rules and Hypothesis

Disease (Patient, tuberculosis):-

Symptom (Patient, persistent_cough),

Symptom (Patient, constant_fatigue),

Symptom (Patient, weight_loss),

Symptom (Patient, loss_of_appetite),

Symptom (Patient, fever),

Symptom (Patient, coughing_up_blood),

Symptom (Patient, night_sweats).

Pneumonia is a disease whose symptoms are cough, fever, shaking chills, shortness of breath. So it will be stored in knowledge base as follow:-

Disease (Patient, pneumonia):-

Symptom (Patient, cough),

Symptom (Patient, fever),

Symptom (Patient, shaking_chills),

Symptom (Patient, shortness_of_breath).

Byssinosis is a disease whose symptoms are chest tightness, cough, wheezing. So it will be stored in knowledge base as follow:-

Disease (Patient, byssinosis):-

Symptom (Patient, chest_tightness),

Symptom (Patient, cough),

Symptom (Patient, wheezing).

Pertusis is a disease whose symptoms are runny nose and mild fever. So it will be stored in the knowledge base as follow:-

Disease(Patient, pertusis):-

Symptom(Patient, runny$_n ose$),

$Symptom(Patient, mild_f ever)$.

Pneumoconiosis is a disease whose symptoms are chronic cough and shortness of breath. So it will be stored in the knowledge base as follow:-

Disease (Patient, pneumoconiosis):-

Symptom (Patient, chronic$_c ough$),

$Symptom(Patient, shortness_o f_b reath)$.

Sarcoidosis is a disease whose symptoms are dry cough, shortness of breath, mild chest pain, scaly rash, fever, red bumps on legs, sore eyes and swollen ankles. So it will be stored in knowledge base as follow:-

Disease (Patient, sarcoidosis):-

Symptom (Patient, dry$_c ough$),

$Symptom(Patient, shortness_o f_b reath)$,

$Symptom(Patient, mild_c hest_p ain)$,

$Symptom(Patient, scaly_r ash)$,

$Symptom(Patient, fever)$,

$Symptom(Patient, red_b umps\_on\_legs)$,

$Symptom(Patient, sore\_eyes)$,

$Symptom(Patient, swollen\_ankles)$.

Figure below depicts the simulation of the diagnosis application. In the figure, S1 D1 denotes the first symptom of first disease. In general Si Dj denotes the "i" symptom of "j" disease. If the program has a positive answer to the symptom, it goes on with the symptoms from that disease. If only one symptom from that disease is negative, it jumps to the first symptom from the next disease.

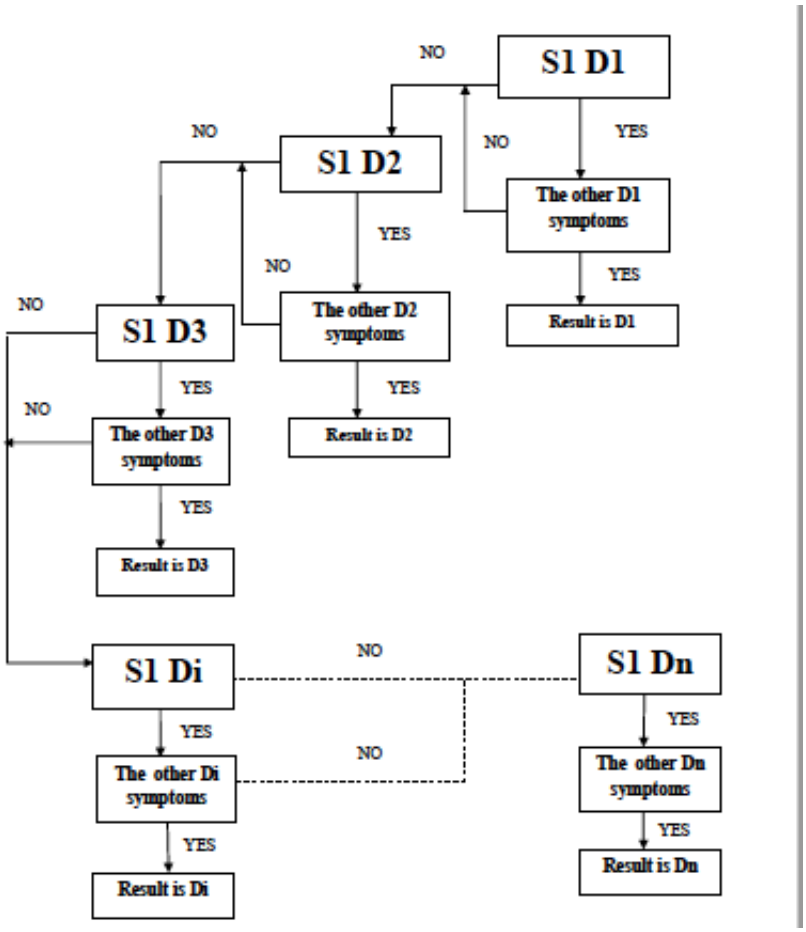

Figure 3: Rule based expert system

Conclusion

This project designed and implemented on the basis of medical expert system, Artificial intelligence. It diagnoses diseases related to lung. This medical expert system is dealing with person's health and an approximate diagnosis of a certain disease is established, this system has a great risk. In reality, there may be more lung diseases which are not considered in the system's knowledge base. So, this knowledge base is incomplete but it can be updated any time with new symptoms and diseases. Symptoms already available in knowledge base

are not 100 percent correct because different doctors have different opinions and there are anomalies in medicines.

Source

```
%To Start the system type start.
% Name : - Thamsanqa Dube

:- use_module(library(jpl)).
start :-sleep(0.4),
write('-----------------------------------------------------------------'),nl,
sleep(0.4),
write('*****************************************************************'),nl,
sleep(0.2),
write("###################||| LUNG DISEASE EXPERT SYSTEM |||#########################"),nl,
sleep(0.4),
write('*****************************************************************'),nl,
sleep(0.4),
write('-----------------------------------------------------------------'),nl,nl,nl,


        /*write("Hi. How are you? First of all tell me your name Please : "),
        read(Patient),*/


interface2.


%hypothesises

disease(Patient,tuberculosis):-
```

```
symptom(Patient, persistent_cough),

symptom(Patient, constant_fatigue),

symptom(Patient, weight_loss),

symptom(Patient, lack_of_appetite),

symptom(Patient, fever),

symptom(Patient, coughing_blood),

symptom(Patient, night_sweats).


disease(Patient,pneumonia):-

symptom(Patient, cough),

symptom(Patient, fever),

symptom(Patient, shaking_chills),

symptom(Patient, shortness_of_breath).


disease(Patient,byssinosis):-

symptom(Patient, chest_tightness),

symptom(Patient, cough),

symptom(Patient, wheezing).


disease(Patient,pertusis):-

symptom(Patient, runny_nose),

symptom(Patient, mild_fever).


disease(Patient,pneumoconiosis):-

symptom(Patient,chronic_cough),

symptom(Patient,shortness_of_breath).


disease(Patient,sarcoidosis):-

symptom(Patient, dry_cough),

symptom(Patient, shortness_of_breath),

symptom(Patient, mild_chest_pain),
```

```
symptom(Patient, scaly_rash),
symptom(Patient, fever),
symptom(Patient, red_bumps_on_legs),
symptom(Patient, sore_eyes),
symptom(Patient, swollen_ankles).

disease(Patient,asbestosis):-
symptom(Patient, chest_tightness),
symptom(Patient, shortness_of_breath),
symptom(Patient, chest_pain),
symptom(Patient, lack_of_appetite).


disease(Patient,asthma):-
symptom(Patient, wheezing),
symptom(Patient, cough),
symptom(Patient, chest_tightness),
symptom(Patient, shortness_of_breath).

disease(Patient,bronchiolitis):-
symptom(Patient, wheezing),
symptom(Patient, fever),
symptom(Patient, blue_skin),
symptom(Patient, rapid_breath).

disease(Patient,influenza):-
symptom(Patient, headache),
symptom(Patient, fever),
symptom(Patient, shaking_chills),
symptom(Patient, nasal_congestion),
symptom(Patient, runny_nose),
```

```
symptom(Patient, sore_throat).


disease(Patient,lung_cancer):-
symptom(Patient, cough),
symptom(Patient, fever),
symptom(Patient, hoarseness),
symptom(Patient, chest_pain),
symptom(Patient, wheezing),
symptom(Patient, weight_loss),
symptom(Patient, lack_of_appetite),
symptom(Patient, coughing_blood),
symptom(Patient, headache),
symptom(Patient, shortness_of_breath).



/*Ask rules*/


symptom(P, Val):-ask('Does the Patient have',Val).
ask(Obj, Val):-known(Obj, Val, true),!.
ask(Obj, Val):-known(Obj, Val, false),!, fail.
ask(Obj, Val):-nl,write(Obj),write(' '),
write( Val) , write('?(y/n)'), read(Ans), !,
((Ans=y, assert(known(Obj, Val, true)));(assert(known(Obj, Val, false)),fail)).


diagnose:-nl,write('Diagnosing a lung disease.....'),nl,disease(symptom,Disease) ,!,nl,
write('That lung disease could be '), write(Disease).
diagnose:- nl, write('Sorry,we may not be able to diagnose the desease!!').


start:-notice,repeat, abolish(known/3),dynamic(known/3), retractall(known/3), diagnose,nl,nl, writ
nl,write('Bye ! Thanks for using this system'),abolish(known,3) .
interface(X,Y,Z) :-
```

```
atom_concat(Y,X, FAtom),

atom_concat(FAtom,Z,FinalAtom),

jpl_new('javax.swing.JFrame', ['Expert System'], F),

jpl_new('javax.swing.JLabel',['--- LUNG DISEASE EXPERT SYSTEM ---'],LBL),

jpl_new('javax.swing.JPanel',[],Pan),

jpl_call(Pan,add,[LBL],_),

jpl_call(F,add,[Pan],_),

jpl_call(F, setLocation, [400,300], _),

jpl_call(F, setSize, [400,300], _),

jpl_call(F, setVisible, [@(true)], _),

jpl_call(F, toFront, [], _),

jpl_call('javax.swing.JOptionPane', showInputDialog, [F,FinalAtom], N),

jpl_call(F, dispose, [], _),

write(N),nl,

( (N == yes ; N == y)

      ->

       assert(yes(Z)) ;

       assert(no(Z)), fail).


interface2 :-

jpl_new('javax.swing.JFrame', ['Expert System'], F),

jpl_new('javax.swing.JLabel',['--- LUNG DISEASE EXPERT SYSTEM ---'],LBL),

jpl_new('javax.swing.JPanel',[],Pan),

jpl_call(Pan,add,[LBL],_),

jpl_call(F,add,[Pan],_),

jpl_call(F, setLocation, [400,300], _),

jpl_call(F, setSize, [400,300], _),

jpl_call(F, setVisible, [@(true)], _),

jpl_call(F, toFront, [], _),

jpl_call('javax.swing.JOptionPane', showInputDialog, [F,'Hi. How are you? First of all tell me you

jpl_call(F, dispose, [], _),
```

```
/*write(N),nl,*/
( N == @(null)
->write('you cancelled'),interface3('you cancelled. ','Thank you ','for use ','me.'),end,fail
; write("Hi. How are you? First of all tell me your name please : "),write(N),nl,pt(N)
).



interface3(P,W1,D,W2) :-
atom_concat(P,W1, A),
atom_concat(A,D,B),
atom_concat(B,W2,W3),
jpl_new('javax.swing.JFrame', ['Expert System'], F),
jpl_new('javax.swing.JLabel',['--- LUNG DISEASE EXPERT SYSTEM ---'],LBL),
jpl_new('javax.swing.JPanel',[],Pan),
jpl_call(Pan,add,[LBL],_),
jpl_call(F,add,[Pan],_),
jpl_call(F, setLocation, [400,300], _),
jpl_call(F, setSize, [400,300], _),
jpl_call(F, setVisible, [@(true)], _),
jpl_call(F, toFront, [], _),
jpl_call('javax.swing.JOptionPane', showMessageDialog, [F,W3], N),
jpl_call(F, dispose, [], _),
/*write(N),nl,*/
( N == @(void)
->write('')
; write("")
).


help :- write("To start the expert system please type 'start.' and press Enter key").
```