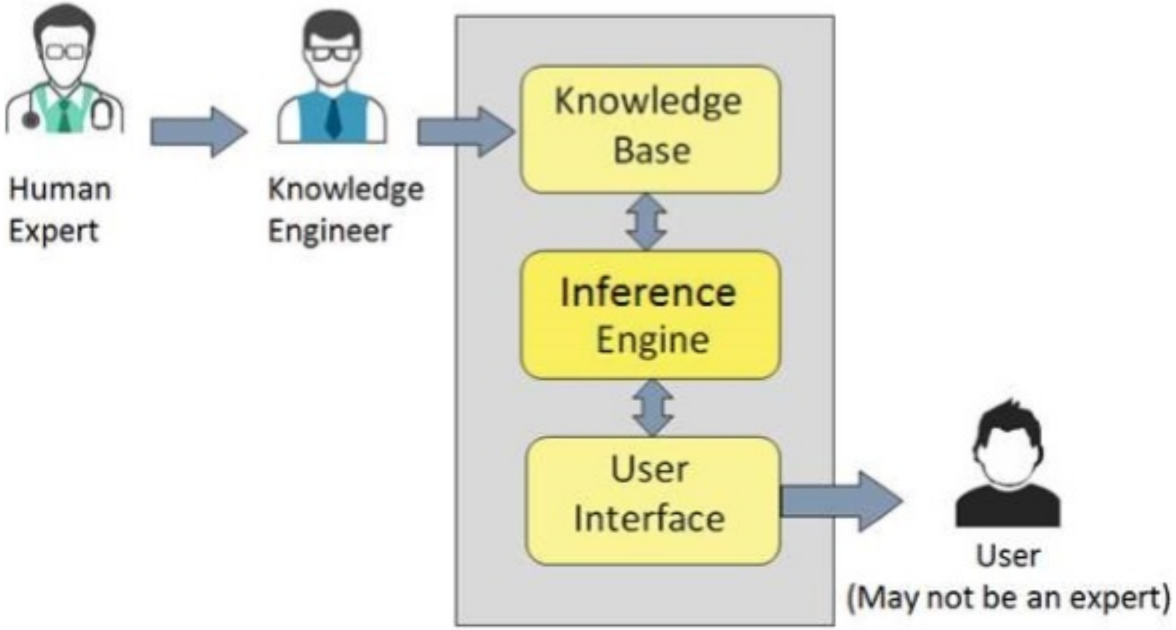
Expert System
Unit-5

- Expert systems (ES) are one of the prominent research domains of AI.
- It is introduced by the researchers at Stanford University, Computer Science Department.
- An expert system is a computer program that is designed to solve **complex problems** and to provide decision-making ability like a human expert.
- These systems are designed for a specific domain, such as **medicine, science,** etc.

Components of Expert Systems

- The components of ES include –
- Knowledge Base
- Inference Engine

User Interface



1.Knowledge Base

- The knowledgebase is a type of storage that stores knowledge acquired from the different experts of the particular domain.
- It is considered as big storage of knowledge.
- It is similar to a database that contains information and rules of a particular domain or subject.

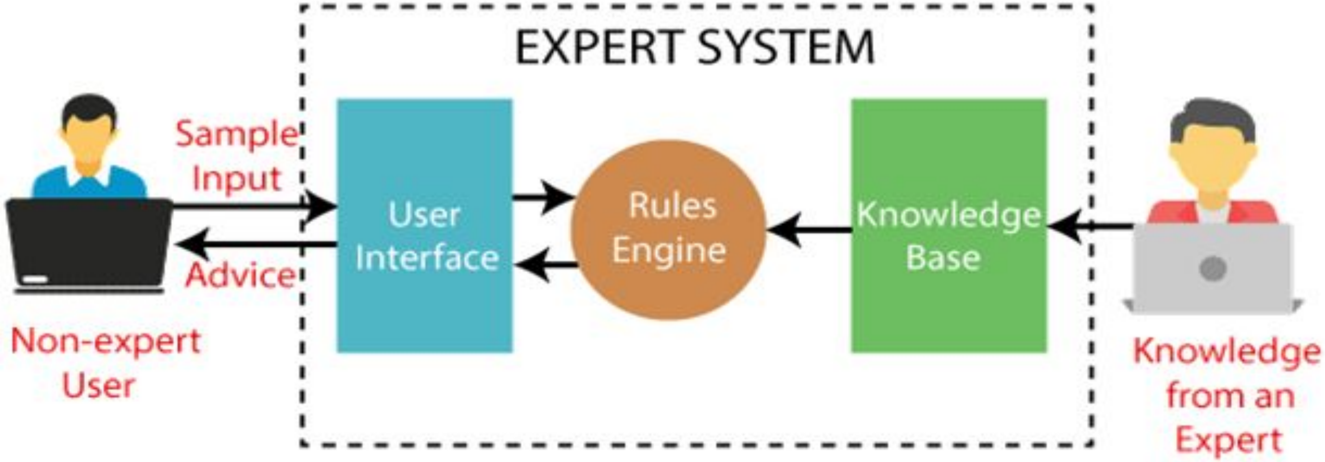
2. **User Interface**

- With the help of a user interface, the expert system interacts with the user, takes queries as an input in a readable format, and passes it to the inference engine. After getting the response from the inference engine, it displays the output to the user.
- it is an interface that helps a non-expert user to communicate with the expert system to find a solution.

3. **Inference Engine(Rules of Engine)**

- The inference engine is known as the brain of the expert system as it is the main processing unit of the system.
- It applies inference rules to the knowledge base to derive a conclusion or deduce new information. It helps in deriving an error-free solution of queries asked by the user.
- With the help of an inference engine, the system extracts the knowledge from the knowledge base.
- There are two types of inference engine:
- **Deterministic Inference engine:** The conclusions drawn from this type of inference engine are assumed to be true. It is based on **facts** and **rules**.
- **Probabilistic Inference engine:** This type of inference engine contains uncertainty in conclusions, and based on the probability.
- Inference engine uses the below modes to derive the solutions:
- **Forward Chaining:** It starts from the known facts and rules, and applies the inference rules to add their conclusion to the known facts.
- **Backward Chaining:** It is a backward reasoning method that starts from the goal and works backward to prove the known facts.

One of the common examples of an ES is a suggestion of spelling errors while typing in the Google search box



examples of the Expert System:

- **DENDRAL:** It was an artificial intelligence project that was made as a chemical analysis expert system.
- It was used in organic chemistry to detect unknown organic molecules with the help of their mass spectra and knowledge base of chemistry.
- **MYCIN:** It was one of the earliest backward chaining expert systems that was designed to find the bacteria causing infections like bacteraemia and meningitis.
- It was also used for the recommendation of antibiotics and the diagnosis of blood clotting diseases.
- **PXDES:** It is an expert system that is used to determine the type and level of lung cancer. To determine the disease, it takes a picture from the upper body, which looks like the shadow. This shadow identifies the type and degree of harm.
- **CaDeT:** The CaDet expert system is a diagnostic support system that can detect cancer at early stages.

Characteristics of Expert System

- **High Performance:** The expert system provides high performance for solving any type of complex problem of a specific domain with high efficiency and accuracy.
- **Understandable:** It responds in a way that can be easily understandable by the user. It can take input in human language and provides the output in the same way.
- **Reliable:** It is much reliable for generating an efficient and accurate output.
- **Highly responsive:** ES provides the result for any complex query within a very short period of time.

Need of the ES:

- **No memory Limitations:** It can store as much data as required and can memorize it at the time of its application. But for human experts, there are some limitations to memorize all things at every time.
- **High Efficiency:** If the knowledge base is updated with the correct knowledge, then it provides a highly efficient output, which may not be possible for a human.
- **Expertise in a domain:** There are lots of human experts in each domain, and they all have different skills, different experiences, and different skills, so it is not easy to get a final output for the query. But if we put the knowledge gained from human experts into the expert system, then it provides an efficient output by mixing all the facts and knowledge
- **Not affected by emotions:** These systems are not affected by human emotions such as fatigue, anger, depression, anxiety, etc.. Hence the performance remains constant.
- **High security:** These systems provide high security to resolve any query.
- **Considers all the facts:** To respond to any query, it checks and considers all the available facts and provides the result accordingly. But it is possible that a human expert may not consider some facts due to any reason.
- **Regular updates improve the performance:** If there is an issue in the result provided by the expert systems, we can improve the performance of the system by updating the knowledge base.

Human Experts vs Expert Systems

Human Experts	**Artificial Intelligence ES**
Unpredictable	Highly consistent
Can get tired	Works without breaks
Knowledge is perishable	Permanent
Expensive	Cost-effective

Advantages

- **Storage:** They have the ability to store large amounts of data easily and add levels of access.
- **Training:** Expert systems can be mapped to help train new employees or incoming employees, thereby saving a lot of money involved in training and preparing employees for their roles.
- **Decision-making:** The ability to implement computing power to derive expert skills and use it to drive decisions is very much possible. This is leaps and bounds ahead from the conventional ‘educated-guess’ approach.
- **Efficiency:** In the process of problem-solving, most of the time, the requirement is time-sensitive. Cutting the process short by making everything more efficient, thereby solving the problem quicker, is a possibility with the use of expert systems.
- They provide a very high speed to respond to a particular query.
- **Error handling:** Expert systems not only reduce human errors but also make sure that errors can be handled in a structured way and solved, which is an important advantage in most of the situations.
- **Wrangling:** With the usage of expert systems, one can look at data or any process and understand it in a way that would’ve been incomprehensible to the naked eye.
- So, what makes an expert system an expert? To understand this, let us take a look at the expert system architecture in Artificial Intelligence.
- They can be used for risky places where the human presence is not safe.

Limitations of Expert System

- The response of the expert system may get wrong if the knowledge base contains the wrong information.
- Like a human being, it cannot produce a creative output for different scenarios.
- Its maintenance and development costs are very high.
- Knowledge acquisition for designing is much difficult.
- For each domain, we require a specific ES, which is one of the big limitations.
- It cannot learn from itself and hence requires manual updates.

Limitations of Expert Systems

- **Creativity:** No matter how good an expert system is designed to be, there are many situations where it lacks finesse and brings out a response that can be deemed as plain and commonplace.
- **Result validation:** If there is a situation where the input data to the knowledge hub is not accurate, the validation engine might not be able to figure this out and, therefore, might furnish inaccurate results.
- **Running cost:** As you might expect, designing, maintaining, and running these expert systems require a lot of finances. But once set up, it will serve for longer durations efficiently than human experts driving in more return on investment overall.
- **Conflict of solutions:** When you consider the human approach to solving a problem, each person can come up with a unique solution. This is not the case when working with expert systems, which forms to be one of the biggest limitations today.

- More expensive
- Taken more time
- Higher Consumption
- Not flexible
- No having common sense
- Having more bugs in its programs
- Not able to adapt to altering environments
- Difficult to maintain
- Having legal and ethical areas
- More expensive in development area
- Having narrow focus
- Required ground verification
- No capable to process for complex automation
- Require update manually
- Development for specific domain

Rule-Based Expert Systems

- Rule-based systems (also known as *production systems* or *expert systems*) are the simplest form of artificial intelligence.
- A rule-based system(RBS) provide automatic problem solving tools for capturing the human expertise and decision making.
- is a system that applies human-made rules to **store**, sort and manipulate data. In doing so, it mimic human intelligence.
- To work, rule-based systems require a set of facts or source of data, and a set of rules for manipulating that data.
- These rules are sometimes referred to as ‘**If statements**’ as they tend to follow the line of ‘IF X happens THEN do Y’.

- The term ***rule*** in AI, which is the most commonly used type of knowledge representation, can be defined as an IF-THEN structure that relates given information or facts in the IF part to some action in the THEN part.
- A rule provides some description of how to solve a problem.
- Rules are relatively easy to create and understand.

- Any rule consists of two parts:
- the IF part, called the *antecedent* (*premise* or *condition*) and the THEN part called the *consequent* (*conclusion* or *action*).
- IF <antecedent>
- THEN <consequent>
- A rule can have multiple antecedents joined by the keywords **AND** (**conjunction**), **OR** (**disjunction**) or
- a combination of both.
- IF <antecedent 1> IF<antecedent 1> AND <ante.cedent 2> OR<ante.cedent 2>
- ..
- AND <antecedent *n*> OR<antecedent *n*>

- THEN <consequent> THEN <consequent>
- The antecedent of a rule incorporates two parts: an ***object*** (*linguistic object*) and its ***value***.
- The object and its value are linked by an ***operator***.
- The operator identifies the object and assigns the value. Operators such as *is*, *are*, *is not*, *are not* are used to assign a **symbolic value** to a linguistic object.
- Expert systems can also use mathematical operators to define an object as numerical and assign it to the **numerical value**.
- IF ‘age of the customer’ < 18 AND ‘cash withdrawal’ > 1000
- THEN‘signature of the parent’ is required
-

-
  Rules can represent relations, recommendations, directives, strategies and heuristics:
- **Relation**
- IF the ‘fuel tank’ is empty THEN the car is dead **Recommendation**
- IF the season is autumn
- AND the sky is cloudy AND the forecast is drizzle
- THEN the advice is ‘take an umbrella’
- **Directive**
- IF the car is dead
- AND the ‘fuel tank’ is empty THEN the action is ‘refuel the car’
-

- **Strategy**
- IF the car is dead
- THEN the action is ‘check the fuel tank’; step1 is complete
- IF step1 is complete AND the ‘fuel tank’ is full
- THEN the action is ‘check the battery’; step2 is complete
- **Heuristic**
- IF the spill is liquid AND the ‘spill pH’ < 6
- AND the ‘spill smell’ is vinegar
- THEN the ‘spill material’ is ‘acetic acid’
-

Frame-based expert systems

- **A frame is a data structure with typical knowledge about a particular object or concept. Frames, first proposed by Marvin Minsky in the 1970s.**

- Each frame has its own name and a set of **attributes** associated with it. *Name*, *weight*, *height* and *age* are slots in the frame *Person*.
- *Model*, *processor*, *memory* and *price* are slots in the frame *Computer*.
- Each attribute or slot has a value attached to it.

- A frame provides a means of organising knowledge in **slots** to describe various attributes and characteristics of the object.
- Frames are an application of **object-oriented programming** for expert systems.
-

- The concept of a frame is defined by a collection of **slots**.
- Each slot describes a particular attribute or operation of the frame.
- Slots are used to store values. A slot may contain a default value or a pointer to another frame, a set of rules or procedure by which the slot value is obtained.