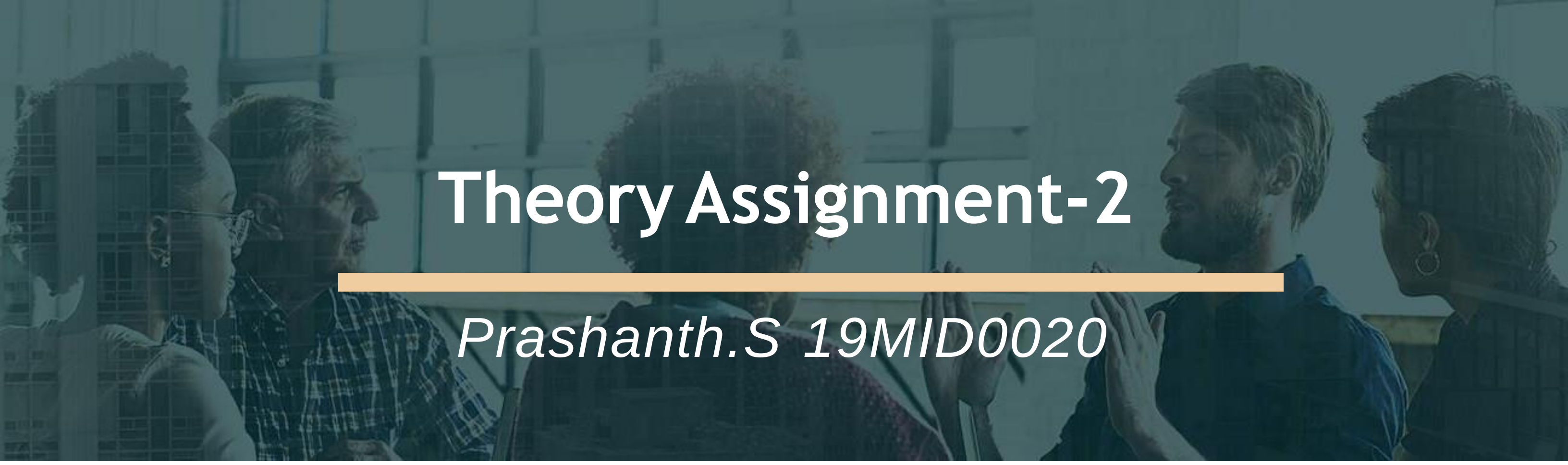


Theory Assignment-2

Prashanth.S 19MID0020

Knowledge Acquisition and Elicitation

Knowledge Acquisition:

Knowledge acquisition is the process of extracting, organizing, and organizing information from a single source, usually human specialists, for usage in software like an ES. This is frequently the most difficult part of creating an ES.

All ES projects must take into account three primary issue areas that are crucial to knowledge acquisition. First, the domain must be assessed to see if the domain's type of knowledge is appropriate for an ES. Second, the source of expertise must be identified and assessed to ensure that the project's specific degree of knowledge is met. Third, if a person is the primary source of expertise, particular knowledge acquisition procedures and participants must be determined.

Technique of Knowledge Acquisition:

The interview is at the center of the procedure. The domain's heuristic model is often derived through a series of in-depth, systematic interviews conducted over several months. It's important to note that the expert and the knowledge engineer aren't the same individual. It's better if the expert and the knowledge engineer aren't the same person, because the more information an expert has, the less able they are to describe their logic. Experts also have a tendency to rationalize their expertise in order to describe their techniques, which can be misleading**.**

1. Watch the person solve real-life problems.

2. Identify the sorts of data, expertise, and processes needed to solve various types of problems through conversations.

3. Collaborate with the expert to create scenarios that can be linked to various problem kinds.

4. Have the expert speak solve a series of problems while you inquire about the reasoning behind each step.

5. Create rules based on the interviews and use them to solve challenges.

6. Review the guidelines and the basic problem-solving approach with the expert.

7. Compare outside experts responses to a set of scenarios developed by the project's expert and the ES.

Prashanth.S 19MID0020

Knowledge Elicitation :

Assuming we've carefully analyzed our domain of interest and specified the expert system's bounds, the first and most important stage is knowledge collection.
The process of extracting information from an expert's head or from a chosen source and representing it in the form required by the expert system is known as knowledge acquisition.

As a result, we may distinguish two parts of this process: knowledge elicitation, in which the expert's knowledge is extracted, and knowledge representation, in which the expert's knowledge is encoded in the expert system.

The knowledge engineer is unlikely to be a specialist in the field.
As a result, the engineer's initial responsibility is to familiarize himself with the area by speaking with domain experts and reading important background material.

Technique of Knowledge Elicitation:

The interview can capture qualitative information, which is crucial for knowledge elicitation, and so serves as a major method for knowledge acquisition. There are several different sorts of interviews, each of which can be used to elicit different types of data.

Prashanth.S 19MID0020

Meta Knowledge

In the domain of AI, meta-knowledge is a piece of data that describes the knowledge.

It is not limited to a single area, but rather seeks to characterize the organization of data in knowledge systems.

Meta-knowledge is primarily used to comprehend and improve the nature of user interface components, as well as to maintain knowledge bases that are utilized in conjunction with inference engines.

With the rising complexity of problems in our environment, it is likely that significant usage of meta-knowledge will become commonplace in the near future.

Many things in a system are critical for autonomous operations to perform well, including control systems, support subsystems, and the architecture.

Some authors divide meta-knowledge into orders:

• Zero order meta-knowledge is knowledge whose domain is not knowledge (and so zero order meta-knowledge is not meta-knowledge per se)

• First order meta-knowledge is knowledge whose domain is zero order meta-knowledge

• Knowledge whose domain is first order meta-knowledge is known as second order meta-knowledge.

• Knowledge whose domain is n order meta-knowledge is known as n+1 order meta-knowledge.

Knowledge engineering, knowledge management, and other research and scientific areas that deal with the study and operations on knowledge, regard meta-knowledge as a unified object/entities abstracted from local conceptualizations and terminologies as a basic conceptual instrument.

Methods of planning, modelling, labelling, learning, and every alteration of a domain knowledge are examples of first-level individual meta-knowledge.

Indeed, for the structuring of meta-levels of individual meta-knowledge, universal meta-knowledge frameworks must be legitimate.

Examples of the first-level individual meta-knowledge are methods of planning, modeling, tagging, learning and every modification of a domain knowledge. Indeed, universal meta-knowledge frameworks have to be valid for the organization of meta-levels of individual meta-knowledge.

Typical Expert System

An expert system is a computer software that can handle complex issues and make decisions in the same way as a human expert can. It does so by pulling knowledge from its knowledge base based on the user's queries, employing reasoning and inference procedures. An expert system's performance is determined on the knowledge stored in its knowledge base by the expert. The more knowledge that is stored in the KB, the better the system performs.

When typing in the Google search box, one of the most common examples of an ES is a suggestion of spelling problems.

Characteristics of Expert Systems

- They are easy to understand
- They are completely reliable
- They are highly responsive
- They have high-performance levels

Components/ Architecture of Expert Systems

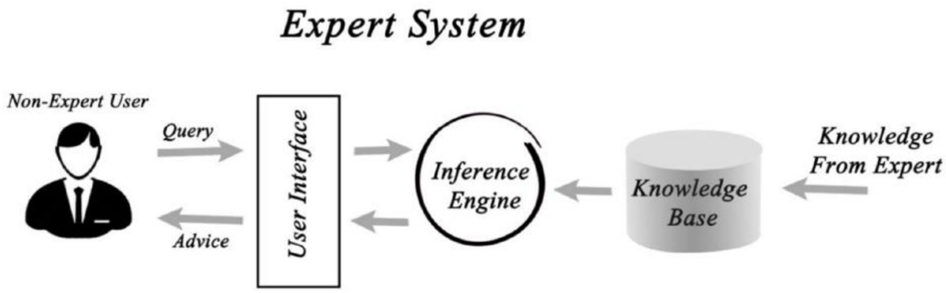
There are 5 Components of expert systems:

- Knowledge Base
- Inference Engine
- Knowledge acquisition and learning module
- User Interface
- Explanation module

Capabilities of Expert Systems

The expert systems are capable of a number of actions including:

- Advising
- Assistance in human decision making
- Demonstrations and instructions
- Deriving solutions
- Diagnosis
- Interpreting inputs and providing relevant outputs
- Predicting results
- Justification of conclusions
- Suggestions for alternative solutions to a problem



MYCIN

MYCIN is a blood infection treatment method that uses an early expert system, or artificial intelligence (AI) technology. MYCIN was first developed at Stanford University in California in 1972. MYCIN would try to diagnose patients based on their symptoms and the results of medical tests.

To arrive at a likely diagnosis, the programmer could ask for further information about the patient and advise additional laboratory testing, after which it would offer a treatment plan. MYCIN would explain the logic behind its diagnosis and suggestion if asked. MYCIN used about 500 production rules to function at a level of competence comparable to human blood infection specialists and somewhat better than general practitioners.

MYCIN worked using a 600-rule knowledge base and a rather modest inference engine. It would ask a long series of simple yes/no or textual questions to the physician who was running the programmers. Finally, it provided a list of possible culprit bacteria ranked from high to low based on the likelihood of each diagnosis, its confidence in each diagnosis' likelihood, the reasoning behind each diagnosis (that is, MYCIN would also list the questions and rules that led it to rank a diagnosis a certain way), and its recommended drug treatment course.

The usage of MYCIN's ad hoc, yet principled, uncertainty framework known as "certainty factors" has aroused dispute. MYCIN's performance was minimally affected by perturbations in the uncertainty metrics associated with individual rules, implying that the system's power was related more to its knowledge representation and reasoning scheme than to the details of its numerical uncertainty model, according to the researchers. Classic Bayesian statistics, according to some observers, should have been possible.

According to MYCIN's creators, this would necessitate either unreasonable probabilistic independence assumptions or specialists providing estimates for an unfeasibly huge number of conditional probabilities. Later research revealed that the certainty factor model may certainly be read in a probabilistic manner, as well as flaws in the model's associated assumptions. The system's modular structure, on the other hand, would be a huge success, leading to the development of graphical models like Bayesian networks.