UCS1524 – Logic Programming

Foundations of an Expert System

Session Meta Data

Author	**Dr. D. Thenmozhi**
Reviewer	
Version Number	1.2
Release Date	17 October 2022

Session Objectives

- Understanding the foundations of an expert system
- Learn about the functions, features and structure of an expert system

Session Outcomes

- At the end of this session, participants will be able to
  - explain expert system, its features, functions and structure.

Agenda

- Introduction to expert system
- Features
- Functions
- Structure

EXPERT SYSTEMS

- An expert system behaves like an expert in some narrow area of expertise

- An expert system is a computer system that emulates, or acts in all respects, with the decision-making capabilities of a human expert

- Typical examples:
  - Diagnosis in an area of medicine
  - Adjusting between economy class and business class seats for future flights
  - Diagnosing paper problems in rotary printing

- Very fashionable area of AI in period 1985-95

SOME FAMOUS EXPERT SYSTEMS

- MYCIN (Shortliffe, Feigenbaum, late seventies)
  Diagnosis of infections

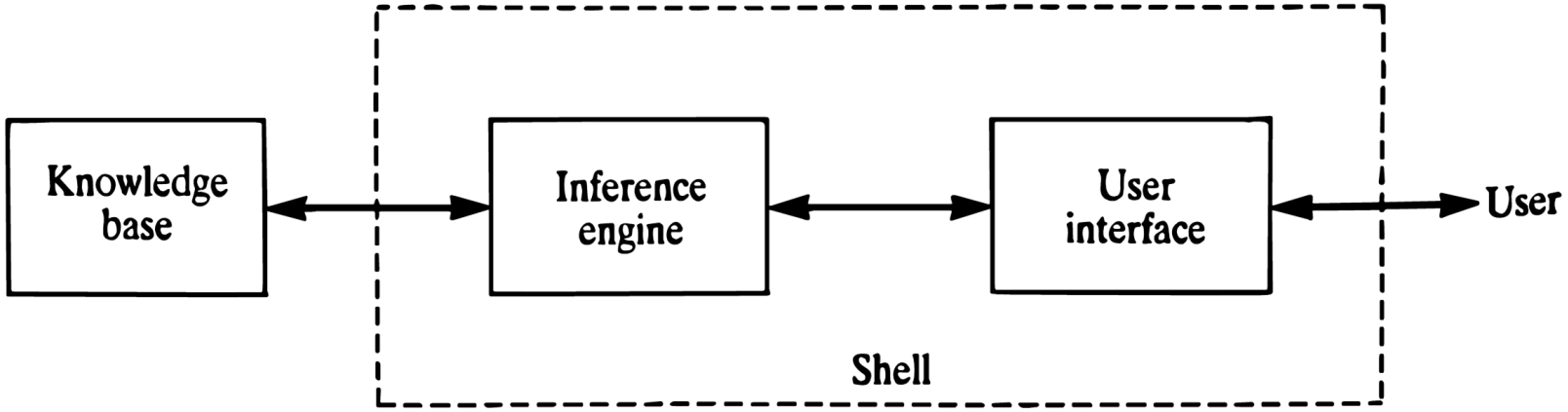
- PROSPECTOR (Buchanan, et al. ~1980)
  Ore and oil exploration

- INTERNIST (Popple, mid eighties)
  Internal medicine

Functions of an expert system

- To build an expert system we have to develop the following functions:
  - problem-solving function capable of using domain-specific knowledge. This may require dealing with uncertainty
  - user-interaction function, which includes explanation of the system's intentions and decisions during and after the problem-solving process

Structure of expert systems



Structure of an expert system

- A knowledge-base
  - Comprises the knowledge that is specific to the domain of application, including such things as simple facts about the domain, rules that describe relations or phenomena in the domain, and possibly also methods, heuristics and ideas for solving problems in this domain.
- An inference engine
  - knows how to actively use the knowledge in the base.
- A user interface
  - caters for smooth communication between the user and the system, also providing the user with an insight into the problem-solving process carried out by the inference engine.
- It is convenient to view the inference engine and the interface as one module, usually called an expert system shell, or simply a shell for brevity.

FEATURES OF EXPERT SYSTEMS

- Problem solving in the area of expertise
- Interaction with user during and after problem solving
- Relying heavily on domain knowledge
- Explanation: ability of explain results to user

Important characteristic of Expert System

- **The Highest Level of Expertise:**
  - The expert system offers the highest level of expertise. It provides efficiency, accuracy and imaginative problem-solving.
- **Right on Time Reaction:**
  - An ES interacts in a very reasonable period of time with the user. The total time must be less than the time taken by an expert to get the most accurate solution for the same problem.
- **Good Reliability:**
  - The ES needs to be reliable, and it must not make any a mistake.
- **Flexible:**
  - It is vital that it remains flexible to handle different types of users.
- **Effective Mechanism:**
  - ES must have an efficient mechanism to administer the compilation of the existing knowledge in it.
- **Capable of handling challenging decision & problems:**
  - An expert system is capable of handling challenging decision problems and delivering solutions.

REPRESENTING KNOWLEDGE WITH IF-THEN RULES

- Traditionally the most popular form of knowledge representation in expert systems

- Examples of rules:

  - if condition P then conclusion C
  - if situation S then action A
  - if conditions C1 and C2 hold then condition C does not hold

DESIRABLE FEATURES OF RULES

- Modularity: each rule defines a relatively independent piece of knowledge
- Incrementability: new rules added (relatively independently) of other rules
- Modifiability (as a consequence of modularity): old rules can be changed relatively independently of other rules.
- Support explanation
- Can represent uncertainty

TYPICAL TYPES OF EXPLANATION

If-then rules facilitate answering the following basic types of user's question

- How explanation

  Answers user’s questions of form: How did you reach this conclusion?

- Why explanation

  Answers users questions: Why do you need this information?

RULES CAN ALSO HANDLE UNCERTAINTY

- If condition A then conclusion B follows with certainty F

- Examples of knowledge-based systems:

  - MYCIN: for identifying various bacteria that could cause acute infections. It could also recommend drugs based on the patient's weight.
  - AL/X: for diagnosing equipment failures
  - AL3: for problem solving in chess
  - DENDRAL: for chemical analysis to predict molecular structure.
  - PXDES: to predict the degree and type of lung cancer
  - CaDet: identify cancer at early stages

EXAMPLE RULE FROM MYCIN

if

1 the infection is primary bacteremia, and

2 the site of the culture is one of the sterilesites, and

3 the suspected portal of entry of the organism is the gastrointestinal tract

then

there is suggestive evidence (0.7) that the identity of the organism is bacteroides.

EXAMPLE RULES FROM AL/X

- Diagnosing equipment failure on oil platforms, Reiter 1980

if

the pressure in V-01 reached relief valve lift pressure

then

the relief valve on V-01 has lifted [*N* = 0.005, *S* = 400]

if

NOT the pressure in V-01 reached relief valve lift pressure,
and the relief valve on V-01 has lifted

then

the V-01 relief valve opened early (the set pressure has drifted)
[*N* = 0.001, *S* = 2000]

EXAMPLE RULE FROM AL3
Game playing, Bratko 1982

if

1 there is a hypothesis, *H*, that a plan *P* succeeds, and

2 there are two hypotheses,

*H*1, that a plan *R*1 refutes plan *P*, and

*H*2, that a plan *R*2 refutes plan *P*, and

3 there are facts: *H*1 *is false*, and *H2 is false*

then

1 generate the hypothesis, *H*3, that the combined plan '*R*1 *or R*2' refutes plan *P*, and

2 generate the fact: *H*3 *implies not*(*H*)

Development of an Expert System

- Steps
  - Consult actual experts for that domain and learn a great deal about it yourself.
  - Extracting some understanding of the domain from experts and literature and moulding this understanding in to a chosen knowledge-representation formalism is called the art of knowledge engineering
- Example:
  - Consider a knowledge base.
  - It consists of simple rules that help identify animals from their basic characteristics assuming that the identification problem is limited just to a small number of animals.
  - Rules in this knowledge base are of the form:
    - RuleName : if Condition then Conclusion

Summary

- Introduction to expert system
- Features
- Functions
- Structure
- Role of if-then rule
- Rules of MYCIN, AL/X and AL3

Check your understanding

- What is shell in ES?
- How will you represent knowledge?