COMP210: Artificial Intelligence

Expert Systems and MYCIN

Trevor Bench-Capon
Room 215, Ashton Building

http://www.csc.liv.ac.uk/~tbc/COMP210

Overview

- We have looked at Rules and Structured Objects for Knowledge Representation
- Expert Systems were a key application of the Rule Based paradigm
- Rule-based expert systems were the big AI success story in the 80s.
- Some remain in use: e.g. Retirement Pension Forecast Adviser
- No longer a research topic, but still of some practical importance

Overview

- We will look at:
- General Features and Architecture of Expert Systems:
- One of the most important expert systems, MYCIN
  - Architecture
  - Rules and certainty factors
  - How MYCIN works
  - What a consultation looks like.
  - How MYCIN explains itself.
- General problems with rule-based expert systems;

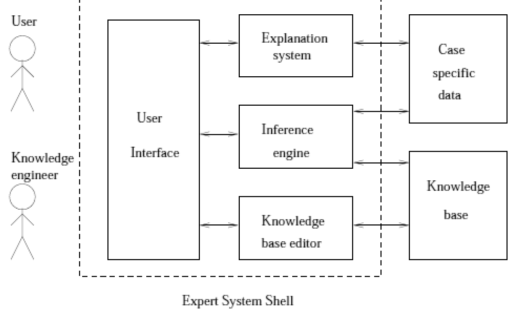
What is an Expert System?

- A expert system is a computing system that is capable of expressing and reasoning about some domain of specialist knowledge.
- Typical domains are:
  - medicine (INTERNIST, MYCIN, . . .)
  - geology (PROSPECTOR)
  - chemical analysis (DENDRAL)
  - configuration of computers (R1)
  - law (British Nationality Act)
- The purpose of the expert system is to be able to solve problems or offer advice in that domain.

What Makes ES Special

- Expert systems can be distinguished from other kinds of AI program because:
  - They deal with subjects of considerable complexit: subjects that normally require a good deal of human expertise.
    - provide expert-level solutions to complex problems.
  - They must be fast and reliable;
  - They must be capable of explaining and justifying solutions
    - be understandable to experts and users
  - They must be sufficiently flexible that new information may be easily accommodated.
    - Declarative representation

Architecture of an Expert System



Architecture of an Expert System

- The knowledge base holds the expertise that the
- system can deploy.
- The knowledge-base is constructed by the knowledge engineer in consultation with the domain expert.
- For most expert systems knowledge representation is through the use of rules (could also be frames, semantic nets etc.). Most expert systems use backward chaining.
- In use, some facts and goals are added to the working memory which represent observations about the domain. Typically the user supplies these in response to questions.
- The inference engine makes inferences from the case specific data and the knowledge in the knowledge base. This leads to more questions as sub-goals are added to working memory
- Backward chaining identifies what the system needs from the user.

Legal Expert System

- citizen(X) :- bornIn(X,uk).
- citizen(X) :- father(X,Y), bornIn(Y,uk).
  - User: citizen(me)?
  - System: where were you born.
  - User: Malta.
    - Add bornIn(me,malta). First rule fails.
  - System: who is your father?
  - User: colin
  - System: where was Colin born.
  - User: London.
    - Add bornIn(colin,uk), using fact in(london,uk). Succeeds
  - System: Yes.

Rules in MYCIN

- Internally held in a Lisp-like syntax, but of the form

```
IF
1. The gram stain of the organism is gramneg,
   and
2. The morphology of the organism is rod, and
3. The aerobicity of the organism is anaerobic
THEN
there is suggestive evidence (0.6) that the
identity of the organism is bacteroides.
```

Certainty factor expressing confidence in the conclusion

Another Example

- Antecedent can contain both AND and OR conditions

```
IF
1. The identity of the organism is not known
   with certainty, and
2. The gram stain of the organism is gramneg,
   and
3. The morphology of the organism is rod, and
4. The aerobicity of the organism is aerobic
THEN
there is strongly suggestive evidence (0.8)
that the identity of the organism is
enterobactericeae.
```

A Rule with OR Conditions

- A rule about treatment

```
IF
1. The therapy under consideration is:
      cephalothin, or
      clindamycin, or
      erythromycin, or
      lincomycin, or
      vancomycin
   and
2. Meningitis is a diagnosis for the patient
THEN
It is definite that the therapy under
consideration is not a potential therapy.
```

OR in the Consequent of a Rule

```
IF
The identity of the organism is bacteroides
THEN
I recommend therapy chosen from among
the following drugs:

1. clindamycin
2. chloramphenicol
3. erythromycin
4. tetracycline
5. carbenecillin
```

Certainty Factors

- Since there is little certain knowledge in medical diagnosis, MYCIN need a way to handle uncertainty
- MYCIN uses certainty factors (CFs), values between +1 and −1 to show how certain its conclusions are, a positive value showing suggestive evidence for the conclusion and a negative value suggestive evidence against the conclusion.
- For example, data of a particular organism relating to its Gram stain, morphology and aerobicity may be as follows.

```
GRAM = (GRMNEG 1.0)
MORPH = (ROD 0.8)
AIR = (ANAEROBIC 0.7)
```

Certainty Factors

- Each statement and rule has a certainty factor
- Statement AND statement – use the minimum of the two CFs
- Statement OR statement – use the maximum of the two CFs
- Rule: CF(antecedent) * CF(rule)
  - Combining conclusions from two rules: (CF(conclusion1)+CF(conclusion2)) minus (CF(conclusion1)*CF(conclusion2))
- Does NOT correspond to probability theory, but simple and tractable. Initial numbers not probabilities anyway!

Example

- R1: If P (0.4) and Q (0.7) then R (0.8)
- R2: If S (0.2) or T (0.5) then R (0.9).
- P and Q = 0.4
- S or T = 0.5
- R1 gives R= 0.32
- R2 gives R = 0.45
- Overall: 0.77 – 0.164 = 0.606

How MYCIN Works

- MYCIN has a four stage task:
  - decide which organisms, if any, are causing significant disease.
  - determine the likely identity of the significant organisms.
  - decide which drugs are potentially useful.
  - select the best drug, or combination of drugs.
- Rules for each stage are in different partitions
- The control strategy for doing this is coded as meta-knowledge.

Main Rule

```
IF
1. There is an organism which requires
   therapy, and
2. Consideration has been given to possible
   other organisms which require therapy
THEN
1. Compile a list of possible therapies, and
2. Determine the best therapy.
ELSE
Indicate that the patient does not require
therapy.
```

Mail Rule gives series of subgoals

- Starts with some clinical observations: blood test results
- Backward chains until it needs more information
- Asks user (who may need to do further tests)
- When a subgoal is complete, e.g, the organism is identified, moves on to next subgoal

Process

- So MYCIN starts by trying to apply the control rule, and this generates *sub-goals.*
- The first of these is to determine if there is an organism which needs to be treated.
- This generates another sub-goal; whether the organism is significant.
- This provokes a question to the user.
- The answer allows other rules to be fired, and these lead to further questions.
- Eventually the IF part of the control rule is satisfied, and the THEN part compiles a list of drugs, and chooses from it.

A consultation with MYCIN

- Systems like MYCIN are often called ***consultative*** expert systems
- Using MYCIN is thus an interactive process:
  - 1. MYCIN starts running.
  - 2. MYCIN asks a question.
  - 3. The user answers it.
  - 4. MYCIN asks another question.
- Only ***relevant*** questions are asked

Example

```
1) Patient's name: (first-last)
** FRED BRAUN
2) Sex
** M
3) Age
** 55
4) Are there any illnesses for Fred Braun
   which may be related to the present illness,
   and from which organisms have been grown
   in the microbiology laboratory?
** Y
-------- CULTURE-1 ----------
5) From what site was the specimen for
   CULTURE-1 taken?
** BLOD
= BLOOD
6) Please give the date and time
   when CULTURE-1 was obtained.
** JUN 20 1977
```

Example continued

```
The first organism isolated from the blood
culture of 20-JUN-77 (CULTURE-1) will be
referred to as:

-------- ORGANISM-1 ---------

7) Enter the laboratory-
reported identity of
ORGANISM-1
** UNKNOWN
8) The stain (gram or Ziehl-
Neelsen acid-fast) of
ORGANISM-1
** NEG
9) Is ORGANISM-1 rod or coccus
** ROD
10) What is the form of the
individual organisms?
** FUSIFORM
...
```

Example continued

```
...
It is important to cover for the
following probable infections and
associated organisms:

INFECTION-1 is BACTEREMIA

<ITEM-1> E.COLI
<ITEM-2> KLEBSIELLA
<ITEM-3> ENTEROBACTER
<ITEM-4> KLEBSIELLA-PNEUMONIA
```

Explanation in MYCIN

- An important feature of any expert system is that it can explain how it comes up with conclusions.
- This allows users to understand what the system is doing, and so see that it is doing sensible things.
- One of the advantages of rules is that they make it possible to generate good explanations simply.
- Essentially the explanation is just a presentation of the *proof tree.*
- We can illustrate this with an example explanation dialogue from MYCIN.

Example Explanation

```
34) Was penicillinase added to CULTURE-1?
** Why?

[3.0] This will aid in determining whether
 ORGANISM-1 is a contaminant.
 It has already been established that:
 [3.1] the site of  CULTURE-1 is blood, and
 [3.2] the gram stain of ORGANISM-1 is grampos

Therefore if
 [3.3] penicillinase was added to this
       blood culture
then
 there is weakly suggestive evidence
   that ORGANISM-1 is a contaminant [RULE 39]
```

Explanation continued

```
** Why?

[4.0] ... in order to determine whether there
 is significant disease associated with
 ORGANISM-1.

 It has already been established that:
  [4.1] this blood culture was
    taken from a sterile source

 Therefore if
  [4.2] ORGANISM-1 is not a contaminant
 then
  there is strongly suggestive evidence
  that there is significant disease
  associated with ORGANISM-1
```

How Explanation Works

- When asked “WHY”, MYCIN uses the current rule.
- If asked “WHY” again, MYCIN uses the rule which caused the current rule to fire
- Remember that MYCIN is chaining backwards from the overall goal of showing significant disease.
- This explanation can be continued by asking more “WHY”s until user supplied information is reached.

Evaluation of MYCIN

- Evaluated by comparing its performance to 8 members of Stanford medical school: 5 faculty members, one research fellow in infectious diseases, one physician and one student.
- They were given 10 randomly selected case histories and asked to come up with diagnoses and recommendations.
- MYCIN performed as well as any of the Stanford medical team and considerably better than the physician or student.

Use of MYCIN

- MYCIN has never been used in clinical practice due to:
  - expense of computing power required at the time
  - the amount of time and typing required for a session
  - incompleteness of the knowledge base.
  - issues relating to professional responsibility – what if it was wrong?
- MYCIN was *enormously influential:*
  - Almost all practical expert systems of the 80s and 90s used ideas from MYCIN
  - Fielded in domains where complete knowledge was available, responsibility issues less (not life critical, advice), and as computer power became cheap (micro computers).

Other Kinds of Expert System

- Most expert systems are rule-based like MYCIN.
- However, there is no reason why expert systems cannot be based on (or use) other forms of knowledge representation:
  - Frames (Internist)
  - Semantic networks (Grebe)
  - Bayesian networks (Prospector)
- What makes a program an “expert system” is not its use of rules, but its expert level performance. However, the MYCIN style proved easiest to implement in practice.

How Expert Systems are Used

- As we saw, MYCIN was intended as an adviser.
- The idea behind many expert systems was similar:
  - Capture expertise known to a few
  - Make it available to many.
- During the "expert systems boom" many ESs were built which tried to *replace experts.*
- Such systems often ran into difficulties:
  - how to construct them?
  - how to field them?
  - Can we trust them?
  - Are they cost effective?
- Also used in education and training to give practice in problem solving and to explain expert reasoning.

Advantages of Expert Systems

- Declarative information – programming an expert system involves capturing the expert's knowledge, its not like programming a conventional system;
  - failure to draw a conclusion - missing knowledge;
  - drawing the wrong conclusion - a faulty statement;
- Interface issues – user-friendly as inferences drawn by the expert system are intended to be similar to inferences drawn by the experts themselves;
  - often users can maintain the system;
  - missing knowledge can be acquired in a natural way;
- Explanation - the ability to explain their conclusions.
- Can be easy to extend and maintain – provided the domain does not change
  - Knowledge of the human body grows, but (for example) laws change quite frequently.

Problems with Expert Systems

- Problems with construction:
  - Knowledge acquisition bottleneck.
  - Machine learning.
- Problems with representation:
  - What does "significant evidence" mean?
  - Handling uncertainty.
- Problems with acceptance
  - Operational issues.
  - Legal issues.
- Problems with domain
  - Brittleness.
  - Common sense knowledge and CYC.