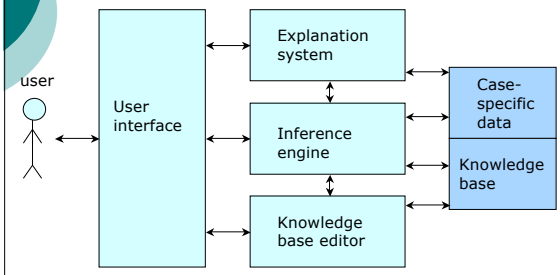
Expert Systems (ES) Week 2

- Expert System architecture
- Knowledge acquisition
- Uncertainty
- Certainty factors

Expert System Architecture



Architecture: Knowledge Base

Consists of facts and rules:

- **Facts**: short term knowledge relating to a particular consultation. They can change rapidly.
- **Rules**: longer term knowledge that generate facts during a consultation.
- Generally represented in the IF..THEN format (Lisp uses (… => …).
- Can be represented using semantic nets, frames, etc., depending upon the implementation.

Architecture: Inference Engine

The part of an Exp Sys which handles rule searching. It:

- Decides on which question to ask next.
- Attempts to draw conclusions (infer knowledge) using the rules in the knowledge base.
- Gives advice on its conclusions.
- Makes use of forward/backward chaining to achieve its aims.
- Needs to be able to resolve rule conflicts (see later).

Architecture: Inference Engine (cont.)

The inference engine also:

- Deals with uncertainty
  - Data for a consultation may be incomplete or inaccurate.
  - Knowledge base may be incomplete or inaccurate as experts make mistakes/don't know everything.
  - Employs an uncertainty strategy e.g. fuzzy logic or certainty factors. Strategy depends on implementation.
- Uses forward/backward chaining, but may use a combination of both.

Architecture: Explanation System

- The part of an Expert System that:
- Justifies the conclusions drawn by the Expert System.
- Explains the reasoning of the system:
  - How a conclusion was reached.
  - Why a question is being asked or why a piece of knowledge is required.

Architecture: Knowledge Base Editor

- Permits addition/deletion/modification Knowledge Base's rules and facts.
- ☑ Useful to remove redundant rules/facts or add new ones.
- ☒ Modification of Knowledge Base can cause problems with existing rules by introducing conflicts or making a goal unattainable: beware!

Architecture: User interface

- Main point of contact between user and Expert System.
- Must be able to communicate with the explanation system, inference engine, and knowledge base editor.
- Needs to be able to represent information in a logical/sensible order.
- Should be able to present requests for information in a suitable format.
- Can be graphical or text-based.

Architecture: User interface (cont.)

- Need to consider usability engineering when designing an interface:
- Learnability: how easy is it to learn how to use the interface?
- Efficiency: does it make entering and displaying information easy?
- Memorability: how easy is it to remember what does what, and where it's located?
- Satisfaction: is it (subjectively) worth using the system again?

Many books on user interface design e.g by Jakob Nielsen

Expert System Shell

- Consists of all of the "processing" parts of an Expert System i.e. not the Knowledge Base or case-specific data.
- ☑ Make it easier to construct an Expert System: just provide the knowledge.
- ☒ Inflexible: difficult to change the way a shell works with its knowledge.

Knowledge Acquisition

People involved are:

- Domain expert:
  - Person with expertise in the problem area (domain).
- Knowledge engineer:
  - Person who questions the domain expert to elicit information for the Knowledge Base.
  - Performs a task analogous to a systems analyst gathering requirements information for a project.

Knowledge Acquisition (cont.)

Acquisition problems

- Much expert knowledge is tacit i.e. the expert assumes "surely everyone knows that".
- Experts need to be asked the right questions.
- Experts may be reluctant to give out information.
- Knowledge engineer must know the Expert System software being used and have some knowledge of the problem domain. Must also have good people skills.

Knowledge Acquisition (cont. again)

Techniques

- Ask about particular situations to determine the expert's general knowledge.
- Ask about pairs of situations which look identical but are handled differently. This helps with learning the expert's vocabulary.

Uncertainty

- Have assumed all knowledge is certain, but it is not always the case:
- If you are coughing and have a sore throat you probably have a cold, but then again this could be a result of you spending your time in a dingy pub.
- Most Expert Systems require some way of saying that something is possibly true, but not necessarily true. Or that some symptoms are usually but not always associated with a disease.

Uncertainty (cont.)

Correct facts + absolute inference rules
= guaranteed conclusions

Uncertain evidence + rough inference rules
= expert advice

Uncertainty sources/corrections

Sources

- Missing or erroneous data and/or rules.
- Inaccurate model.
- Subjective data.

Corrections

- Redundant data or rules.
- Expert's heuristics.
- Cautions strategy.

Uncertainty methods

- Probabilities.
- Bayesianism.
- Certainty factors.
- Fuzzy logic.

AI course covers certainty factors.

Certainty factors

- Can associate CFs with facts and rules.
- Values are in range -1 to 1 inclusive.
- May be referred to as CVs (certainty values).

Value meanings
-1: something is definitely not true.
0: have no knowledge about the truth value.
1: something is definitely true.

Certainty factors (cont.)

Can assign names to certain values:

- Unknown 0.2
- Maybe 0.4
- Probably 0.6
- Almost certainly 0.8
- Definitely 1.0

CFs aren't the same as probabilities, but are related – probabilities range 0…1 inclusive

Certainty factor propagation

For every rule we get a CF value for each of its antecedents (premises) (E) as well as an overall rule CF (rule).

Example
Given the rule
(IF X THEN Y) and CF(rule) = 0.8

Calculate the certainty of consequent (or hypothesis) Y given CF for antecedent X is 0.3

Certainty factor calculations

For single antecedent (IF X THEN Y) rules use:
CF(H,E) = CF(E) x CF(Rule)

For conjunctive multiple antecedent rules (IF A AND B AND … THEN Y) rules use:

AND

CF(H,E1 AND H,E2 …) =
min{CF(E1), CF(E2), …} x CF(Rule)

For disjunctive multiple antecedent rules (IF A OR B OR … THEN Y) rules use:

OR

CF(H,E1 OR H,E2 …) =
max{CF(E1), CF(E2), …} x CF(Rule)

Certainty factor calculations

If two rules give different certainty factors for the same consequent, they can be combined using one of the following:

Both certainty factors (CF1, CF2) positive:
CF1 + CF2 - (CF1 * CF2)

Both certainty factors (CF1, CF2) negative:
CF1 + CF2 + (CF1 * CF2)

Certainty factor calculations

- One certainty factor positive, the other negative:

$$\frac{\text{CF1 + CF2}}{\text{1 - min(abs(CF1), abs(CF2))}}$$

Note

- It doesn't matter which of CF1 and CF2 are positive and negative.
- abs(x) = |x|, so for example
  abs(-5) = |-5| = 5

Certainty factor simplifications

- The formulae for AND and OR are the same except AND uses min and OR uses max.
- Think of truth tables for AND and OR:
  - AND generally results in 0 (false), the minimum value.
  - OR generally results in 1 (true), the maximum value.
- The formulae for both +ve and both –ve are the same except for the change in sign in the middle of the formulae: the sign is the opposite of what both certainty factors are.

A certainty factor problem to try

R1: IF A AND B THEN C	CF(R1) = 0.6
R2: IF D OR G THEN F	CF(R2) = 0.8
R3: IF H THEN C	CF(R3) = 0.5
R4: IF S AND T THEN F	CF(R4) = 0.7

Calculate the individual and combined CFs of C and F given the following values

CF(A)=0.3, CF(B)=-0.5, CF(D)=0.6, CF(G)=0.4, CF(H)=0.7, CF(S)=-0.5 and CF(T)=0.2

Certainty factor answers

Certainty factors

R1. IF A AND B THEN C
CF(A)=0.3, CF(B)=-0.5, CF(R1)=0.6
CF(C) = min(CF(A), CF(B)) * CF(R1)
= min(0.3, -0.5) * 0.6
= -0.5 * 0.6 = **-0.3**

R2. IF D OR G THEN F
CF(D)=0.6, CF(G)=0.4, CF(R2)=0.8
CF(F) = max(CF(D), CF(G)) * CF(R2)
= max(0.6, 0.4) * 0.8
= 0.6 * 0.8 = **0.48**

Certainty factors (cont.)

R3. IF H THEN C
CF(H)=0.7, CF(R3)=0.5
CF(C) = CF(H) * CF(R3)
= 0.7 * 0.5 = **0.35**

R4. IF S AND T THEN F
CF(S)=-0.5, CF(T)=0.2, CF(R4)=0.7
CF(F) = min(CF(S), CF(T)) * CF(R4)
= min(-0.5, 0.2) * 0.7
= -0.5 * 0.7 = **-0.35**

Question: what about expressions with both AND and OR in them?

Certainty factors (cont.)

From R1 CF(C)=-0.3
From R3 CF(C)= 0.35

$$\text{Combined CF(C)} = \frac{(CF1 + CF2)}{(1-\min(\text{abs}(CF1), \text{abs}(CF2)))}$$

$$= \frac{(-0.3 + 0.35)}{(1-\min(0.3, 0.35))}$$

$$= \frac{0.05}{1 - 0.3}$$

= 0.05/0.7 = **0.07**

Certainty factors (cont.)

From R2 CF(F)= 0.48
From R4 CF(F)=-0.35

$$\text{Combined CF(C)} = \frac{(CF1 + CF2)}{(1-\min(\text{abs}(CF1), \text{abs}(CF2)))}$$

$$= \frac{(0.48 + (-0.35))}{(1-\min(0.48, 0.35))}$$

$$= \frac{0.13}{1 - 0.35}$$

= 0.13/0.65 = **0.2**

Certainty factors (cont.)

- For calculations with only ANDs or only ORs
  E.g. R1. IF A AND B AND C THEN D
  Use the appropriate formula for AND or OR:

  CF(D) = min(CF(A), CF(B), CF(C)) * CF(R1)

- For calculations with a mix of ANDs and Ors
  E.g. R2. IF E AND F OR G THEN H
  Use a mixture of formulae for AND or OR:

  CF(H) = min(CF(E), max(CF(F), CF(G)) * CF(R2)