Knowledge-based and Expert Systems - 1

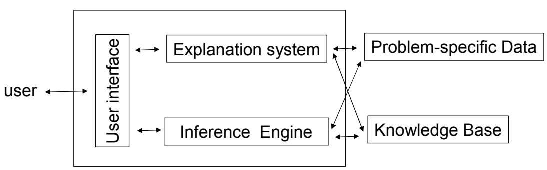
- Expert systems (ES) provide expert quality advice, diagnosis or recommendations.
- ES solve real problems which normally would require a human expert.
- An example situation – using an ES in diagnosing rare diseases.

Knowledge-based and Expert Systems - 2

- Two basic steps in building an expert system:
  1. Extracting the relevant knowledge from the human expert by a *knowledge engineer*.
  2. S/he then develops the *knowledge base* of the ES.
- *Knowledge acquisition* generally involve interviewing the expert.
- An ES should be
  1. easily inspected and modified
  2. able to explain its reasoning

Knowledge-based and Expert Systems - 3

- General architecture of expert system:



Knowledge-based and Expert Systems - 4

- The User Interface
  1. Enable the system to pose questions to the users
  2. Provide explanations about why a particular question is asked
  3. Allow user queries
  4. Displaying the derived results

Knowledge-based and Expert Systems - 5

- The Problem-Specific Database
  1. All info about the current problem.
  2. All conclusion that the system has been able to derive.
- The Knowledge Base
  1. Contains all of the relevant, domain-specific, problem-solving knowledge.
  2. Two perspecitves: Nature and Format

Knowledge-based and Expert Systems - 6

- Inference Engine
  1. Interpreter of the knowledge stored in the knowlede base.
  2. Find connections between the problem features and solutions.
- The Explanation Facilities
  1. Justify 'why' a question was asked and 'how' it reached some conclusion.

Logical Inferences

- The process of reasoning involves *making inferences* from known facts.
- Given a set of premises known (or thought to be true) and a reasoning method, certain conclusions can be inferred to also be true.
- Making inferences involves the derivation of new facts from a set of true facts.
- Predicate logic provides a set of sound rules of inference with which we can perform logical inferences.

Logical Inferences – Modus Ponens

- The best known of these is *modus ponens*
- If statements p and (p → q) are known to be true, then we can infer that q is true.
- The basis for rule-based reasoning
- Example:

  If someone has flu then he has high temperature

  ∀X (has_flu(X) → high_temperature(X))

Logical Inferences – Modus Ponens

If the statement

has_flu(peter)

is found in the database,

Then through modus ponens, we can infer

high_temperature(peter)

Logical Inferences – Modus Tolens

- If the statement (p → q) is known to be true, and q is false, then p is false.
- If the relationship ∀X (has_flu(X) → high_temperature(X)) is true and if peter has no high temperature, then he doesn't have flu. That is if

  ¬ high_temperature(peter)

  which implies, through modus tolens,

  ¬ has_flu(peter)

Deduction - 1

- The Oxford dictionary defines deduction as

  "the process of using information you have in order to understand a particular situation or to find the answer to a problem"

- Logically correct inference, i.e deduction from true premises is guaranteed to result in true conclusions.
- The most accepted, understood method.
- The basis of both propositional and predicate logics.

Deduction - 2

- For example,

  IF Object A is larger than Object B
  AND Object B is larger than Object C
  THEN Object A is larger than Object C

- In predicate logic, this is represented as

  ∀A∀B∀C (larger(A,B) ∧ larger(B,C) → larger(A,C))

Deduction - 3

- If the following axioms exist:

```
larger(house, car)
larger(car, cat)
```

- Through deductive reasoning

```
larger(house,cat)
```

can be derived.

Abduction - 1

- The reasoning method commonly used for generating explanations.
- Unlike deduction, it does not guarantee a true conclusion.
- While abductive inference is unsound, it is a quite useful technique and we use it often in our daily lives.
- Assuming the following rule

```
∀X (has_flu(X) → high_temperature(X))
```


Abduction - 2

- Assuming the following axiom exists

```
high_temperature(john)
```

- Abduction concludes

```
has_flu(john)
```

- There could be other reasons why john has high temperature.

Abduction - 3

- Given the following:

```
( A → B )
B is true
```

- Abduction allows us to say

```
A is possibly true
```


Induction - 1

- Reasoning from particular facts or individual cases to a general conclusion.
- The basis of scientific discovery.
- The most common form is:

```
P(A) is true
P(B) is true
```

- Then by induction we conclude

```
∀X, P(X) is true.
```


Induction - 2

- Observing john over a period of time and noted that whenever he had high temperature, it turned out that he had flu.
- One could induce that

```
∀X, high_temperature(X) → has_flu(X)
```

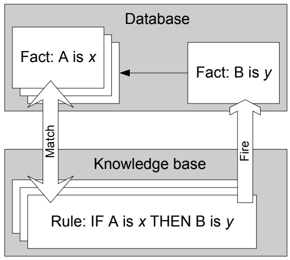
- Obviously this is not always true.

Inferences in Rule-based Systems - 1

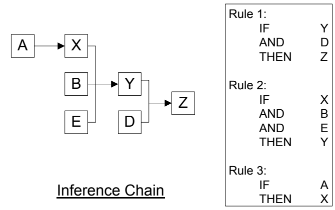
- In rule-based ES

  Domain knowledge is represented as IF-THEN rules.

  Data are facts about the current situation.



Inferences in Rule-based Systems - 2



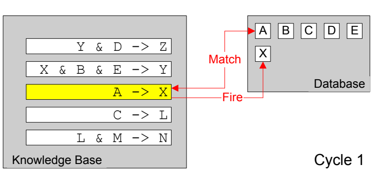
Forward Chaining (Data – Driven)

```
Rule 1:    Y & D → Z

Rule 2:    X & B & E → Y

Rule 3:    A → X

Rule 4:    C → L

Rule 5:    L & M → N
```

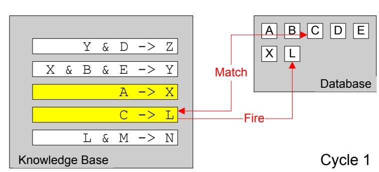

Forward Chaining (Data – Driven)

- Reasoning starts from data and proceeds forward with that data.
- Each time only the top most rule is executed.
- When fired, the rule addes a new fact in the database.
- Any rule can be executed only once.
- The match-fire cycle stops when no further rule can be fired.

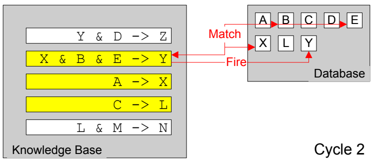
Forward Chaining (Data – Driven)



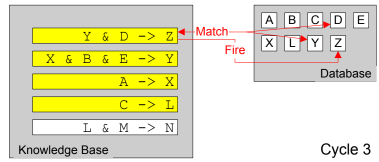
Forward Chaining (Data – Driven)



Forward Chaining (Data – Driven)



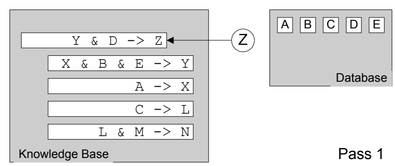
Forward Chaining (Data – Driven)



Backward Chaining (Goal – Driven)

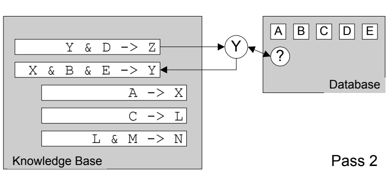
- The ES has the goal (a hypothetical solution) and the inference engine attempts to prove it.
- Find rules that might have the desired solution in the THEN parts.
- Subgoal(s) might be needed in proving the IF parts
- Search rules that can prove the subgoals.
- The process repeats.

Backward Chaining (Goal – Driven)



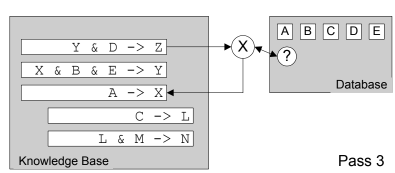
Goal: Z

Backward Chaining (Goal – Driven)



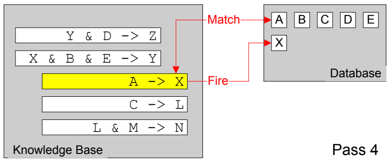
Sub -goal: Y

Backward Chaining (Goal – Driven)



Sub-goal: X

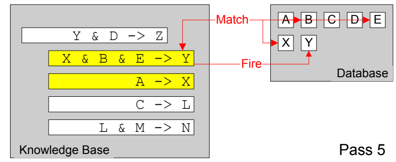
Backward Chaining (Goal – Driven)



Pass 4

Sub-goal: X

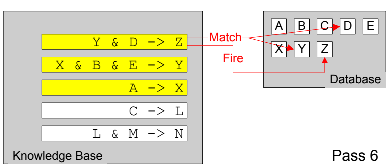
Backward Chaining (Goal – Driven)



Pass 5

Sub-goal: Y

Backward Chaining (Goal – Driven)



Pass 6

Goal: Z

Forward or Backward?

- Forward Chaining – for analysis and interpretation.

  For example, DENDRAL determines the molecular structure of unknown soil based on its mass spectral data.

- Backward Chaining – for diagnosis

  For example, MYCIN diagnoses infectious blood diseases.

- Combining forward and backward chaining?

Readings

1. P.47 – 78 of Luger (2002)
2. P.247 – 267 of Luger (2002)
3. Section 5.1 – 5.6 of Negnevitsky, M. (2002) *Artificial Intelligence, A Guide to Intelligent Systems*. Addison Wesley.

   On Frame-based Expert Systems (Essential Reading)
4. Section 3.1 – 3.5 of Negnevitsky, M. (2002) *Artificial Intelligence, A Guide to Intelligent Systems*. Addison Wesley.

   On Handling Uncertainty in Rule-based Expert Systems (Highly Recommended Reading)