Lecture 6: Expert Systems
Basic Concepts

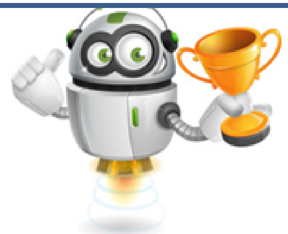
Brunel University London

Lecture Outline

- Concepts of expert systems
  - Knowledge base
  - Inference engine
  - Reasoning process
- Structure of expert systems
  - Components
  - Rule-based knowledge representation
- Some examples

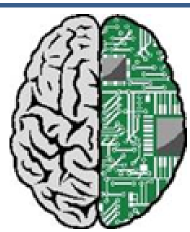
Brunel University London

Humans vs Computers



Computing wins

- Input and output
- Information processing and memory



Closely matched

- Complex movement
- Vision
- Language
- Structured problem solving

Brain still wins

- Creativity
- Emotion and Empathy
- Planning and Executive Function
- Consciousness

Humans (experts) have experience and knowledge

Expert System is way of encoding knowledge

Brunel University London

DECEMBER 7, 2015

BLOG

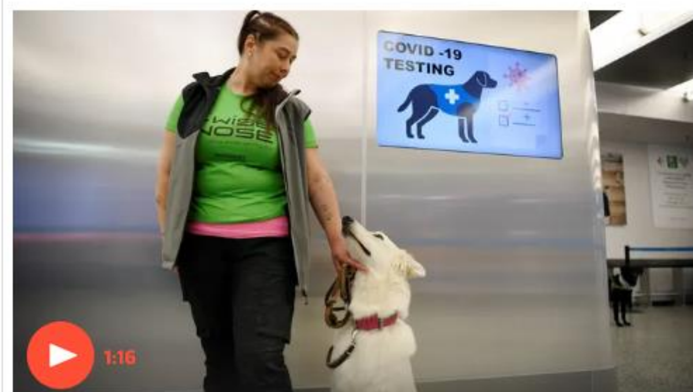
Birdbrains? Pigeons capable of recognizing malignant tumors in breast tissue

The Guardian

'Close to 100% accuracy': Helsinki airport uses sniffer dogs to detect Covid

Researchers running Helsinki pilot scheme say dogs can identify virus in seconds

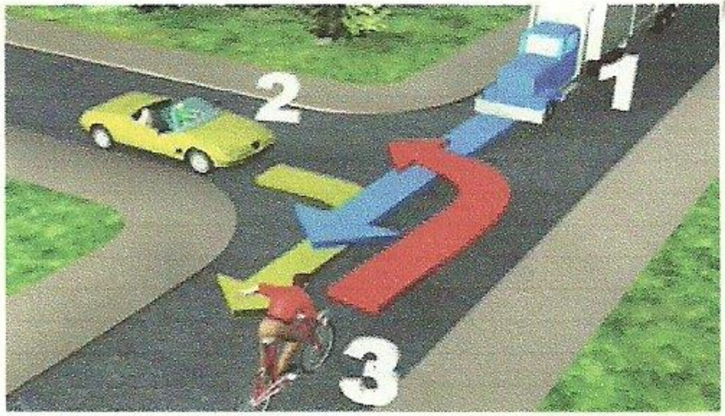
- Coronavirus - latest updates
- See all our coronavirus coverage



▲ Sniffer dogs used to detect coronavirus in Helsinki airport as part of trial – video

Four Covid-19 sniffer dogs have begun work at Helsinki airport in a state-funded pilot scheme that Finnish researchers hope will provide a cheap, fast and effective alternative method of testing people for the virus.

Expertise: Cycling



Imagine you are cycling and come to a junction.

In this case, you are an ***expert*** in cycling.

There are the basic ***rules*** for ***reasoning***. Your ***knowledge*** can be formulated as the following statements (knowledge representation):

IF the "traffic light" is green

THEN the action is go

IF the "traffic light" is red

THEN the action is stop

Brunel University London

Basic concepts: Reasoning

Common sense reasoning:

If I pull out in front of a speeding car then I will get hit

I don’t want to die

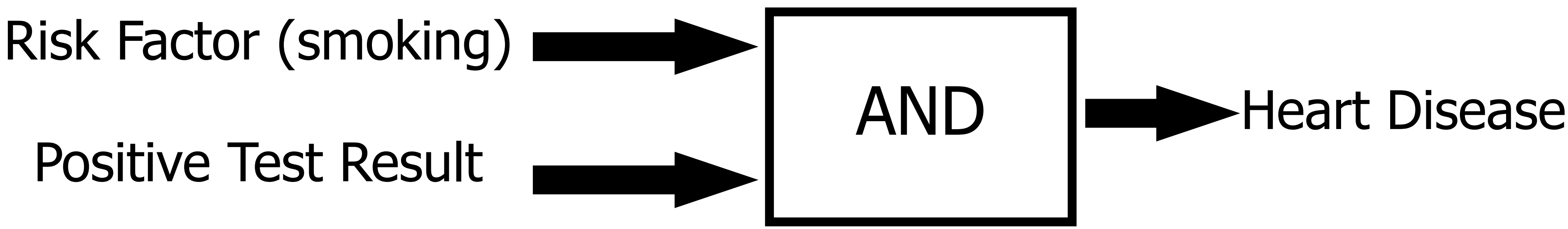
- **IF** there is a fast car driving from the left, **THEN** I’ll wait to pull out ← The rule!
- There **IS** a fast car ← The fact!

Conclusion: I will wait

Basic concepts: Knowledge

What is knowledge?

Knowledge is a theoretical or practical understanding of a ***subject*** or a ***domain***.



Knowledge: sum of what is currently known

Basic concepts: Experts

Who are the experts?

Those who possess knowledge are called **experts**.

An expert should have:

- Deep knowledge (eg. medical diagnosis)
- Strong practical experience (e.g. driving)

Questions: How do you represent how experts think? How do you represent knowledge?

Basic concepts: Experts

ledge are called **experts**.

Deep knowledg

- Strong practica

Questions: How do you re
think? How do you repr

Brunel University London

Knowledge Representation - I

2 possible ways to represent knowledge:

Programming language:

- Procedural knowledge
  - "how to"
  - Knowledge about how to perform some task
- Declarative knowledge
  - "what is"

Production Rules:

- IF-THEN expressions
- IF some condition(s) exists THEN perform some action(s)
- Test-Action rules
- Rule-Based System

Knowledge Representation - II

IF <antecedent> THEN <consequent>

- Rule Part 1
  - Antecedent (premise or condition)
  - IF part
  - Test part
- Rule Part 2
  - Consequent (conclusion or action)
  - THEN part
  - Action part
- A rule **fires** when its condition part is satisfied and its action part is executed

Knowledge Representation - III

- Rule can have multiple antecedents
  - Conjunction AND
  - Disjunction OR
  - Or a combination of both
  - Good habit to avoid mixing conjunction and disjunction in one rule
  - IF <antecedent1> AND <antecedent2> THEN <consequent>

- Consequent can have multiple clauses
  - IF <antecedent> THEN <consequent1>, …, <consequent$_n$>

Knowledge Representation - III

- Rule can have multiple antecedents
  - Conjunction AND
  - Disjunction OR
  - Or a combination of both
  - Good habit to avoid mixing conjunction and disjunction in one rule
  - IF <antecedent1> AND <antecedent2> THEN <consequent>

- Consequent can have multiple clauses
  - IF <antecedent> THEN <consequent1>, …, <consequentn>

- Google car: At a T-junction, if you want to turn right, what rules do you have to follow?

Brunel University London

Knowledge Representation - IV

What can rules represent?

1. Relations
   IF fuel tank is empty then car is dead

2. Recommendations
   If the season is autumn
   And the sky is cloudy
   And the forecast is drizzle
   Then the advice is take an umbrella

3. Directive
   If the car is dead
   And the fuel tank is empty
   Then the action is refuel the car

4. Strategies

If the car is dead, Then the action is check the fuel tank. Step 1 is complete

If step1 is complete, And the fuel tank is empty, Then check the battery. Step 2 is complete

5. Heuristics

If the spill is liquid, And the spill ph is < 0, And the spill smell is vinegar

Then the spill material is acetic acid

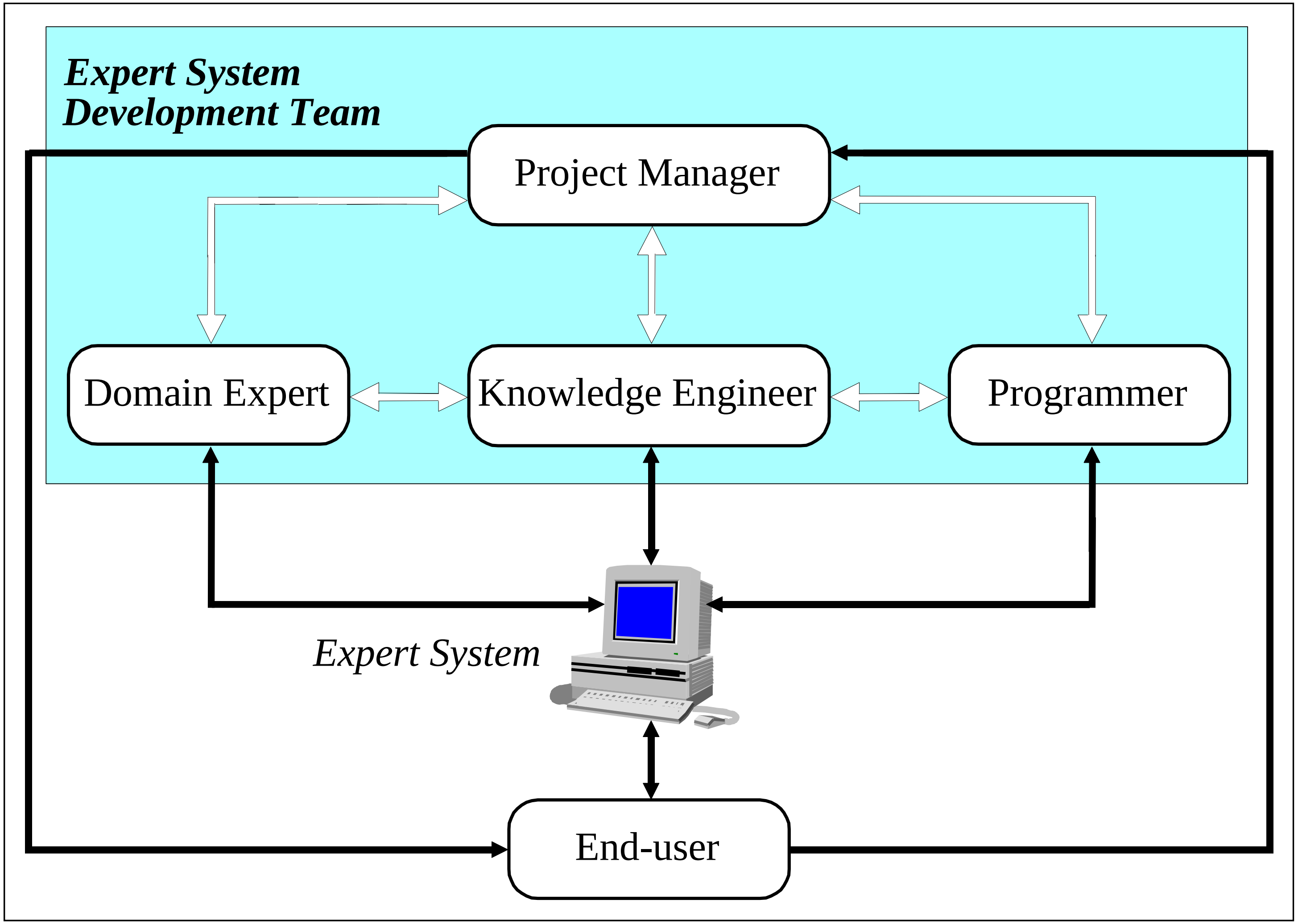
Expert system – I: The definition

Expert systems are software packages designed to assist humans in situations in which an expert in a specific area is required.

Major tasks:

- obtain the required knowledge from an expert.
- express knowledge as a collection of rules in the form of logical implications (knowledge base).
- extract conclusions (reasoning)

Building an Expert System



Expert Systems-II:

knowledge base, inference engine, and reasoning

Knowledge base:

A set of rules describing knowledge of a specific domain. Rules can have multiple antecedents joined by keywords AND (conjunction), OR (disjunction) or a combination of both.

Inference engine:

carries out the reasoning whereby the expert system reaches a solution. It links the rules given in the knowledge base with the facts provided in the database.

Reasoning: a mechanism for selecting the relevant facts and extracting conclusions from them in a logical way.

Example: Given that the rule 'If A then B' and the fact A are both in the knowledge base, then the fact B can be concluded

Expert Systems-III: Some comments

Knowledge representation	Good
Uncertainty tolerance	Good
Imprecision tolerance	Bad
Adaptability	Bad
Learning ability	Bad
Explanation ability	Good
Knowledge discovery and data mining	Bad
Maintainability	Bad

Machine learning complement with expert systems very well!

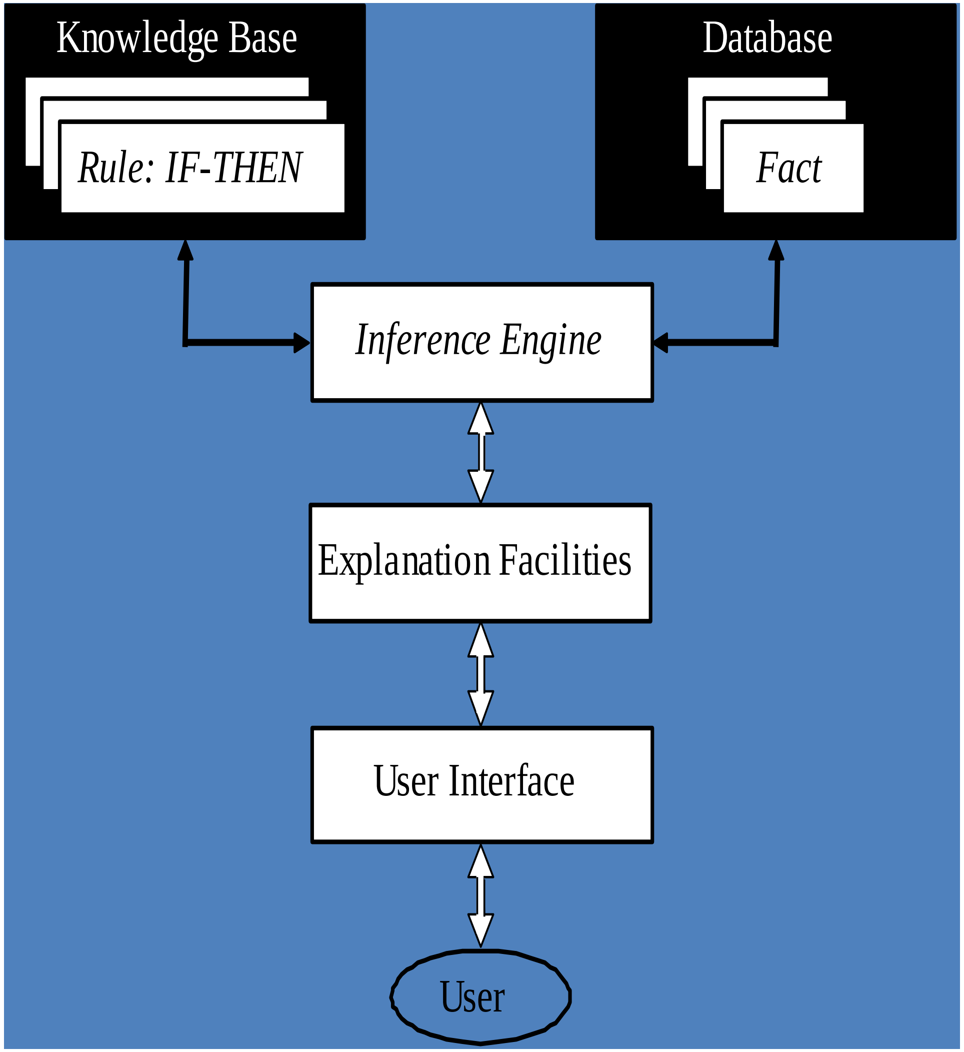
Expert Systems-IV
Example 1: Car maintenance

```
Rule 1: IF a car has
         A good battery
         Good spark plugs
         Fuel
         Good tires
        THEN the car can move
Rule 2: IF the battery is good
        THEN there is electricity
Rule 3: IF there is electricity
       and the spark plugs are good
       THEN the spark plugs will fire
Rule 4: IF the spark plugs fire
       and there is gas
       THEN the engine will run
Rule 5: IF the engine runs
       and there are good tires
       THEN the car will move
```

B = battery is good
E = there is electricity
G = there is fuel
S = spark plugs are good
C = car will move
F = spark plugs will fire
R = engine will run
T = there are good tires

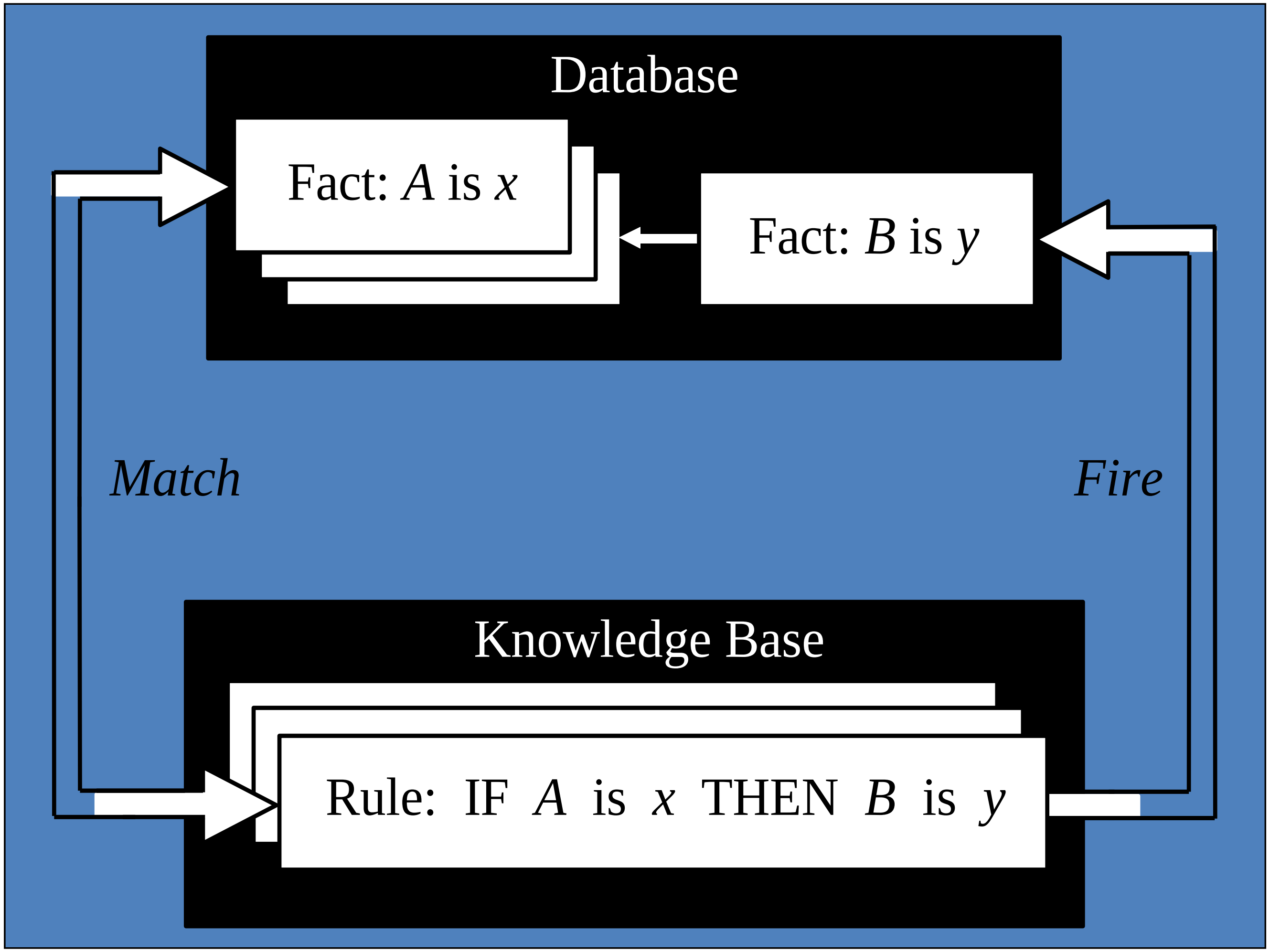
Rule 1: $B \wedge S \wedge G \wedge T \rightarrow C$
Rule 2: $B \rightarrow E$
Rule 3: $E \wedge S \rightarrow F$
Rule 4: $F \wedge G \rightarrow R$
Rule 5: $R \wedge T \rightarrow C$

Basic structure of a rule-based expert system



- In a rule-based expert system, the domain knowledge is represented by a set of IF-THEN production rules and data is represented by a set of facts about the current situation.
- The inference engine compares each rule stored in the knowledge base with facts contained in the database.
- When the IF (condition) part of the rule matches a fact, the rule is fired and its THEN (action) part is executed.
- The matching of the rule IF parts to the facts produces **inference chains.**
- The inference chain indicates how an expert system applies the rules to reach a conclusion.

Inference engine cycles via a match-fire procedure



An example of an inference chain

Rule 1:	IF	*Y* is true
	AND	*D* is true
	THEN	*Z* is true
Rule 2:	IF	*X* is true
	AND	*B* is true
	AND	*E* is true
	THEN	*Y* is true
Rule 3:	IF	*A* is true
	THEN	*X* is true

Facts : A, B, E, D

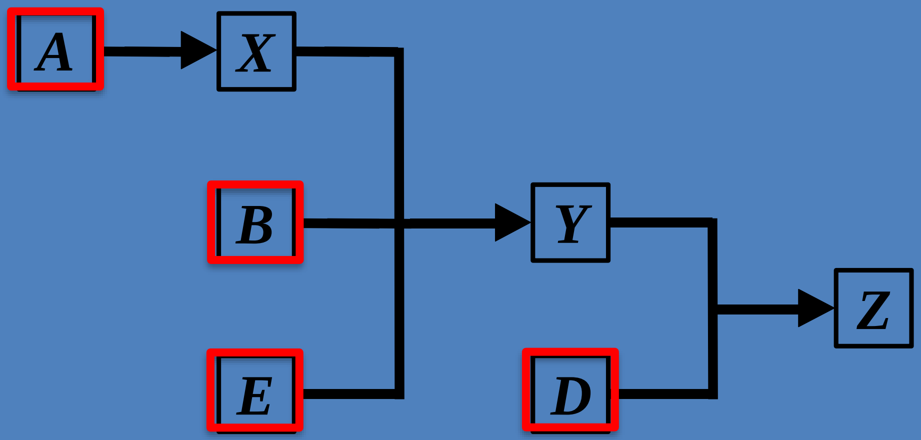
An example of an inference chain

Rule 1: IF *Y* is true
AND *D* is true
THEN *Z* is true

Rule 2: IF *X* is true
AND *B* is true
AND *E* is true
THEN *Y* is true

Rule 3: IF *A* is true
THEN *X* is true

Facts : A, B, E, D



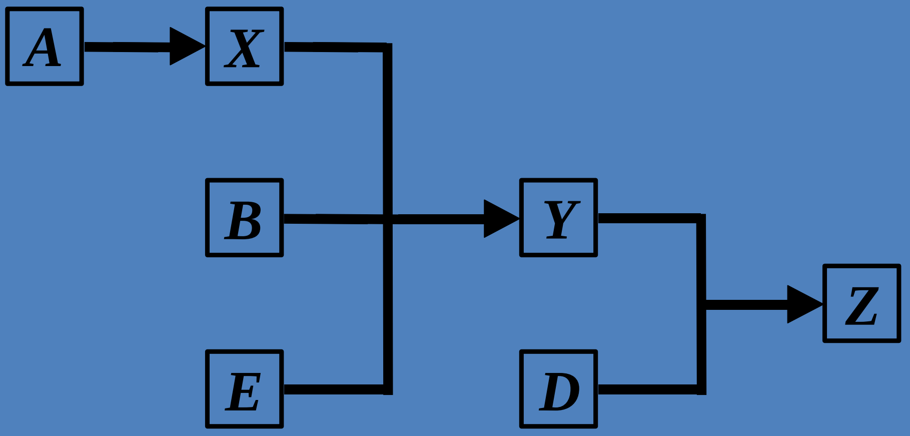
An example of an inference chain

Rule 1: IF Y is true
AND D is true
THEN Z is true

Rule 2: IF X is true
AND B is true
AND E is true
THEN Y is true

Rule 3: IF A is true
THEN X is true

What is the ultimate goal?

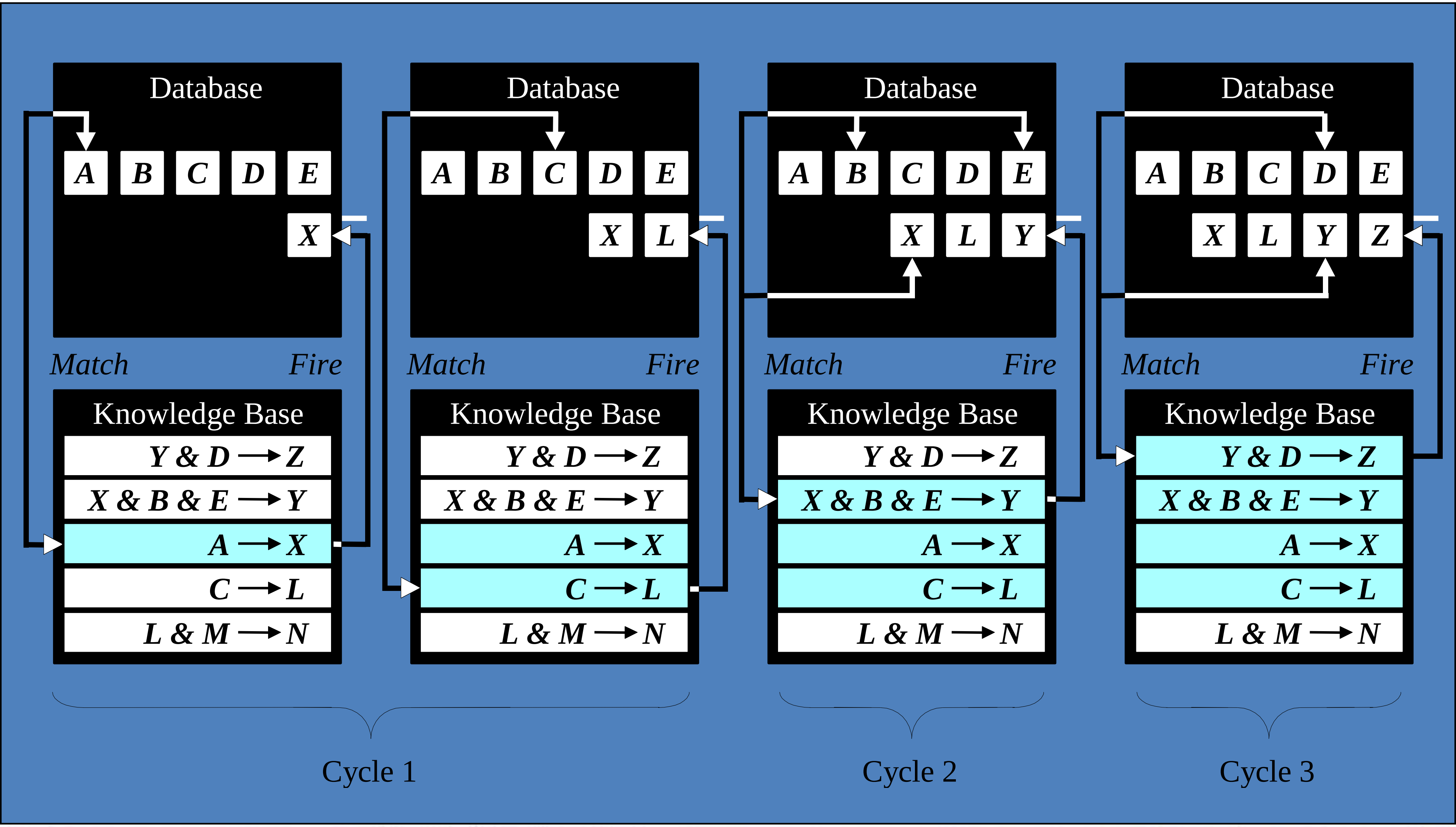


Forward chaining: I

- Forward chaining is *data-driven reasoning*.
- The reasoning starts from the known data and proceeds forward with that data.
- Each time only the topmost rule is executed.
- When fired, the rule adds a new fact in the database.
- Any rule can be executed only once.
- The match-fire cycle stops when no further rules can be fired.

Brunel University London

Forward chaining: II



Forward chaining: III

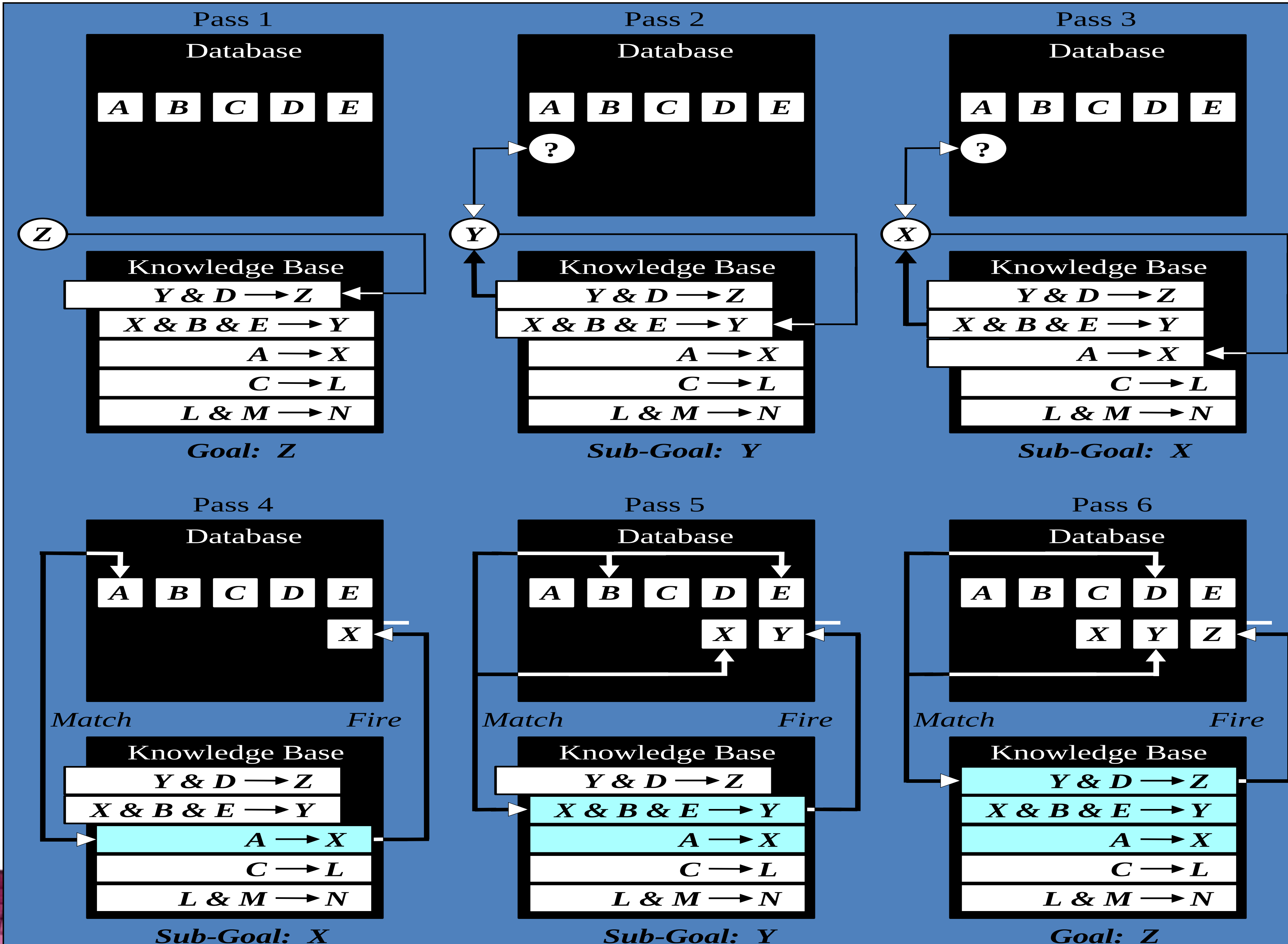
- Forward chaining is a technique for gathering information and then inferring from it whatever can be inferred.
- However, in forward chaining, many rules may be executed that have nothing to do with the established goal.
- Therefore, if our goal is to infer only one particular fact, the forward chaining inference technique would not be efficient.

Brunel University London

Backward chaining: I

- Backward chaining is the *goal-driven reasoning*.
- In backward chaining, an expert system has a goal, and the inference engine attempts to find the evidence to prove it.
- First, the knowledge base is searched to find rules that might have the desired solution. Such rules must have the goal in their THEN (action) parts.
- If such a rule is found and its IF (condition) part matches data in the database, then the rule is fired and the goal is proved.
- However, this is rarely the case.
- Thus the inference engine puts aside the rule it is working with (the rule is said to *stack*) and sets up a new goal, a *subgoal*, to prove the IF part of this rule.
- The knowledge base is searched again for rules that can prove the subgoal.
- The inference engine repeats the process of stacking the rules until no rules are found in the knowledge base to prove the current subgoal.

Backward chaining: II



Choosing Forward or Backward Chaining

- If an expert first needs to gather some information and then tries to infer from it whatever can be inferred, choose the forward chaining inference engine.
- However, if your expert begins with a hypothetical solution and then attempts to find facts to prove it, choose the backward chaining inference engine.

Conflict resolution

- Earlier we saw two simple rules for a road junction. Let us now add third rule:

  Rule 1:

  IF the **'traffic light' is green**

  THEN **the action is go**

  Rule 2:

  IF the 'traffic light' is red

  THEN the action is stop

  Rule 3:

  IF the **'traffic light' is green**

  THEN **stop AND check for pedestrians**

- Rule 2 and Rule 3 have the same IF parts, and can be set to fire when the condition part is satisfied. These rules represent a conflict set.

- Inference engine must determine which rule to fire from such a set. Method for choosing this is called ***conflict resolution.***
- In forward chaining, BOTH rules would be fired. Rule 1 is fired first (topmost), and as a result, its THEN part is executed giving the action value "go".
- However, Rule 3 is also fired ('traffic light' is green is still in the database). As a consequence, action then takes new value "stop".

Methods used for conflict resolution

- **Fire the rule with the highest priority.** In simple applications, the priority can be established by placing the rules in an appropriate order in the knowledge base. Usually this strategy works well for expert systems with around 100 rules.

- **Fire the most specific rule.** It is based on the assumption that a specific rule processes more information than a general one.

- **Fire the rule that uses the data most recently entered in the database.** This method relies on time tags attached to each fact in the database. In the conflict set, the expert system first fires the rule whose antecedent uses the data most recently added to the database.

Brunel University London

Metaknowledge and Metarules

- Metaknowledge can be simply defined as knowledge about knowledge. Metaknowledge is knowledge about the use and control of domain knowledge in an expert system.

- In rule-based expert systems, metaknowledge is represented by metarules. A metarule determines a strategy for the use of task-specific rules in the expert system.

- **Metarule 1:**

  Rules supplied by experts have higher priorities than rules supplied by novices.

- **Metarule 2:**

  Rules governing the rescue of human lives have higher priorities than rules concerned with clearing overloads on power system equipment.

Brunel University London

Metaknowledge and Metarules

3 Rules of Robotics Issac Asimov:

I. A robot may not injure a human being or, through inaction, allow a human being to come to harm.
II. A robot must obey the orders given it by human beings except where such orders would conflict with the First Law.
III. A robot must protect its own existence as long as such protection does not conflict with the First or Second Laws

- **Metarule 1:**

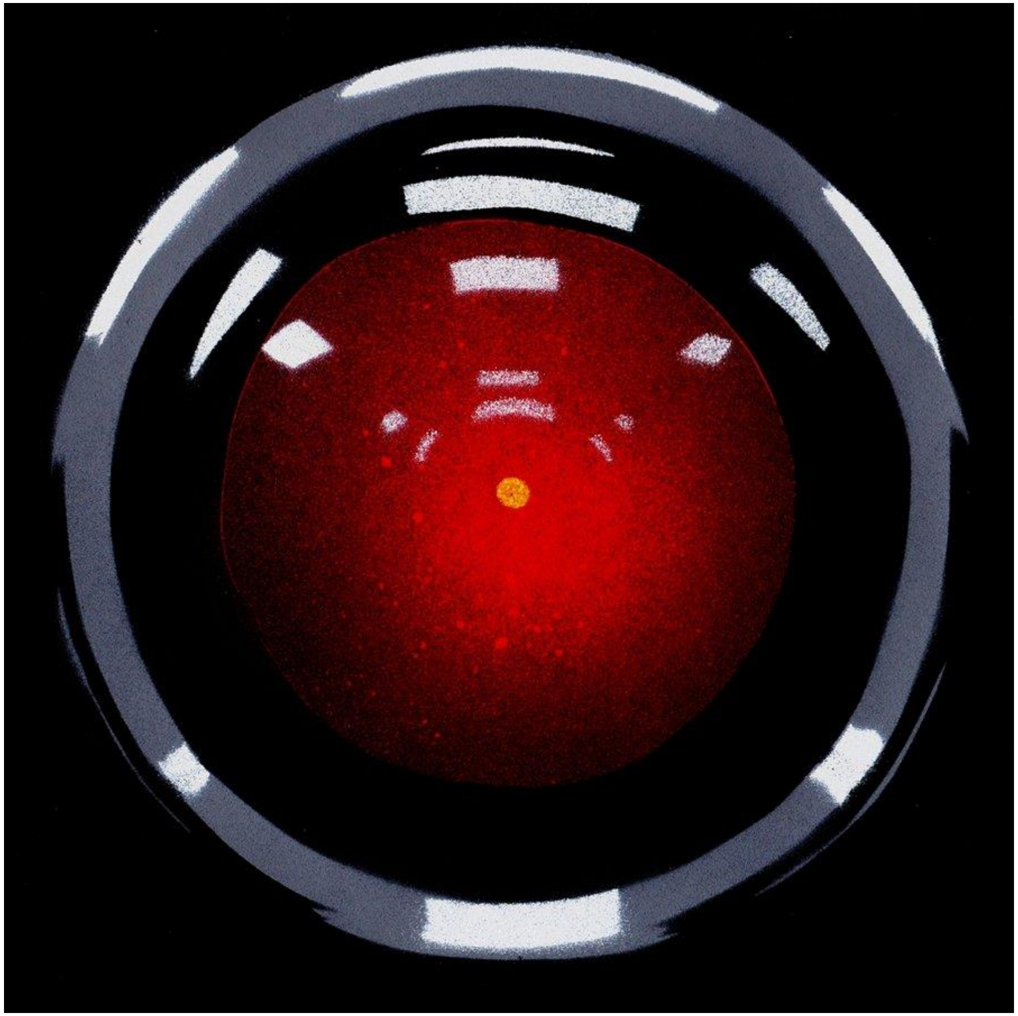
  Rules supplied by experts have higher priorities than rules supplied by novices.

- **Metarule 2:**

  Rules governing the rescue of human lives have higher priorities than rules concerned with clearing overloads on power system equipment.

Brunel University London

Metaknowledge and Metarules



- **Metarule 1:**

  Rules supplied by experts have higher priorities than rules supplied by novices.

- **Metarule 2:**

  Rules governing the rescue of human lives have higher priorities than rules concerned with clearing overloads on power system equipment.

https://www.youtube.com/watch?v=dSIKBliboIo

Brunel University London

Back on Earth …

BBC | Sign in | News | Sport | Weather | iPlayer | Sounds

NEWS

Home | UK | World | Business | Politics | Tech | Science | Health | Family & Education

Technology

Tesla Model 3: Autopilot engaged during fatal crash

17 May 2019

Share

Self-driving cars

This article is more than **1 year old**

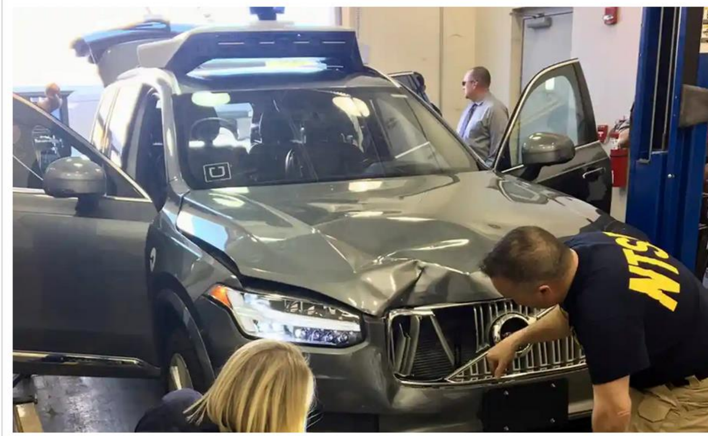
Driver was streaming The Voice when Uber self-driving car crashed, say police

Rafaela Vasquez looked up half a second before Arizona crash that killed woman, report says

Reuters in San Francisco

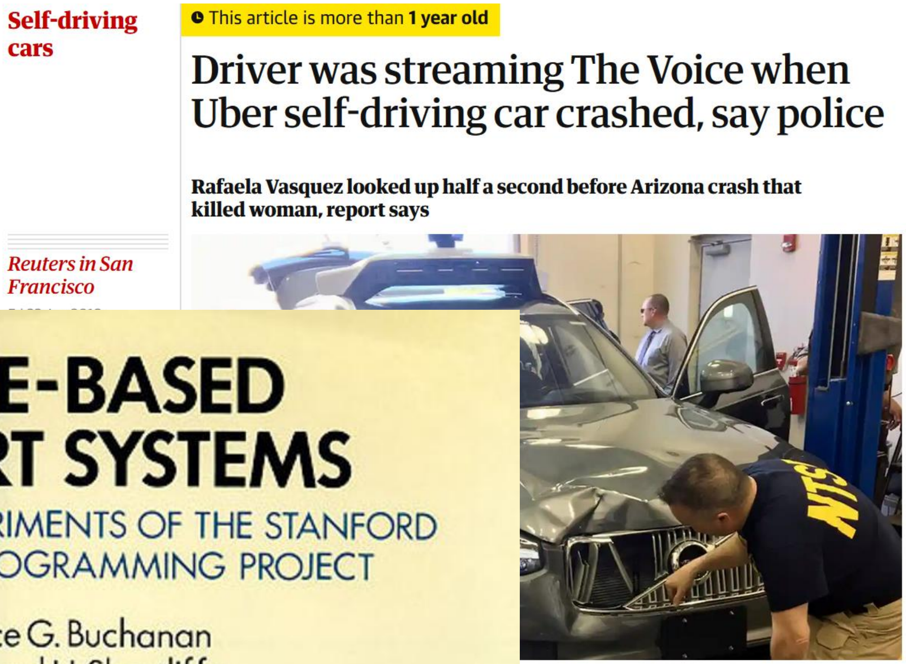
Fri 22 Jun 2018
09.32 BST

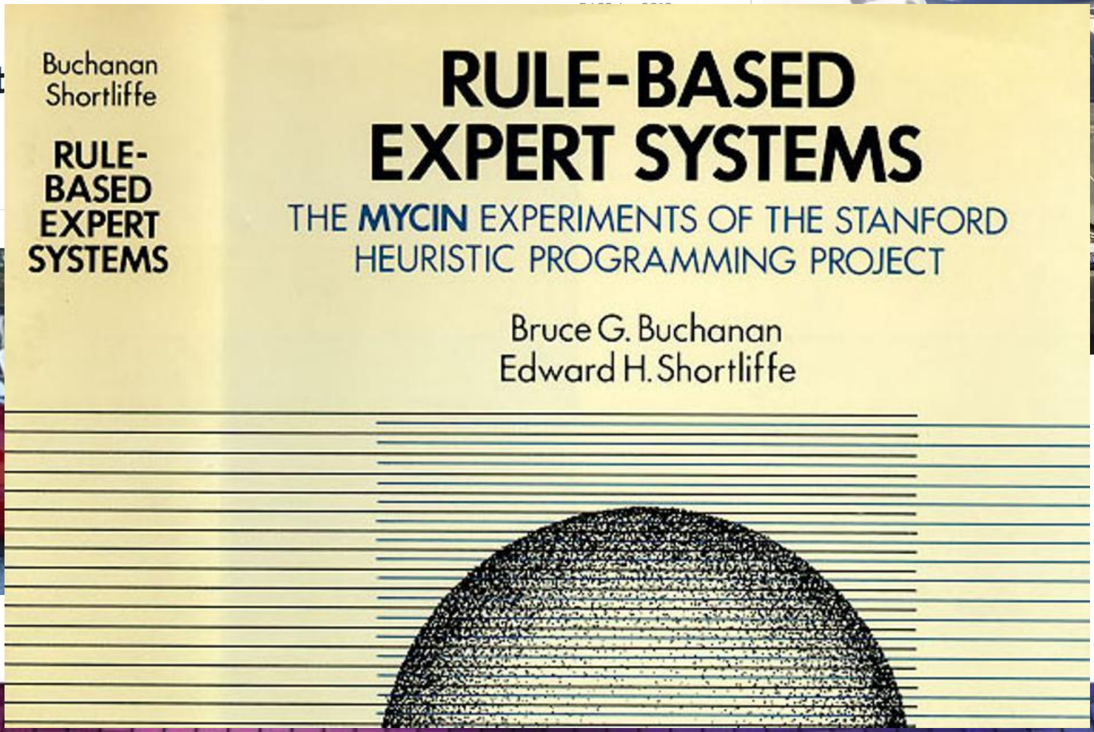
67



Brunel University London

Back on Earth …





MYCIN EXPERT SYSTEM

www.nipunjaswal.com | How Helpful Are Expert Systems In Medical ? | 20, Apr 2013

PRESENTED BY NIPUN JASWAL

MYCIN was an early expert system that used artificial intelligence to identify bacteria causing severe infections, such as bacteremia and meningitis, and to recommend antibiotics, with the dosage adjusted for patient's body weight — the name derived from the antibiotics themselves, as many antibiotics have the suffix "-mycin". The Mycin system was also used for the diagnosis of blood clotting diseases.

MYCIN was never actually used in practice. This wasn't because of any weakness in its performance. As mentioned, in tests it outperformed members of the Stanford medical school faculty. Some observers raised ethical and legal issues related to the use of computers in medicine — if a program gives the wrong diagnosis or recommends the wrong therapy, who should be held responsible? However, the greatest problem, and the reason that MYCIN was not used in routine practice, was the state of technologies for system integration, especially at the time it was developed.

Brunel University London

Advantages of rule-based expert systems

- **Natural knowledge representation.** Expert often explains problem-solving with expressions like: "In such-and-such situation, I do so-and-so". These can be represented naturally as IF-THEN production rules.

- **Uniform structure.** Production rules have uniform IF-THEN structure. Each rule is an independent piece of knowledge.

- **Separation of knowledge from its processing.** Expert system structure provides effective separation of knowledge base from inference engine. Possible to develop different applications using same expert system shell.

- **Dealing with incomplete and uncertain knowledge.** Most rule-based expert systems are capable of representing and reasoning with incomplete and uncertain knowledge.

Brunel University London

Disadvantages of rule-based expert systems

- **Opaque relations between rules.** Although individual production rules are relatively simple, their logical interactions within large sets of rules may be opaque. Difficult to observe how individual rules serve the overall strategy.

- **Ineffective search strategy.** Inference engine applies exhaustive search through all production rules during each cycle. Expert systems with large sets of rules (over 100 rules) can be slow - unsuitable for real-time applications.

- **Inability to learn.** In general, rule-based expert systems do not have an ability to learn from the experience. Human experts know when to “break the rules”. Expert systems cannot automatically modify its knowledge base / adjust or add rules. Knowledge engineer is still responsible for maintaining the system.

Review

In this lecture you have learned about:

- Definition of Expert Systems
- Why we need them
- Basic concepts and structure
- Representation of knowledge

Reading: Sections 2.1-2.5 of Chapter 2, “Artificial Intelligence" by A. Negnevitsky

Viva Deadline TODAY

Last chance for Clustering Assessment!

Next Lecture (2 weeks)

Deep Learning & PYTHON with Alina Miron