Knowledge Management and Expert Systems

MS (AI & DS) | Types, development, and the role of AI and neural networks | 60 minutes

Anurag Srivastava (DoES, ABV-IIITM Gwalior)

Plan for the 60 minutes

Segment	Minutes	What we cover
Introduction	0–3	Why knowledge is the real asset
Knowledge management	3–14	DIKW, tacit and explicit, SECI, KM cycle, Indian cases
Expert systems	14–28	Architecture, representation, inference, MYCIN, XCON
Types of expert systems	28–35	By approach and by task; matching exercise
Development	35–43	Lifecycle, knowledge acquisition, tools
Role of AI and ANN	43–56	ML, neural networks, hybrids, Indian and global cases
Activity and wrap-up	56–60	Design a mini expert system; takeaways

Learning outcomes

1

Explain

Knowledge management and how organisations capture knowledge.

2

Describe

Expert system components and how they reason.

3

Classify

Types of expert systems and how they are developed.

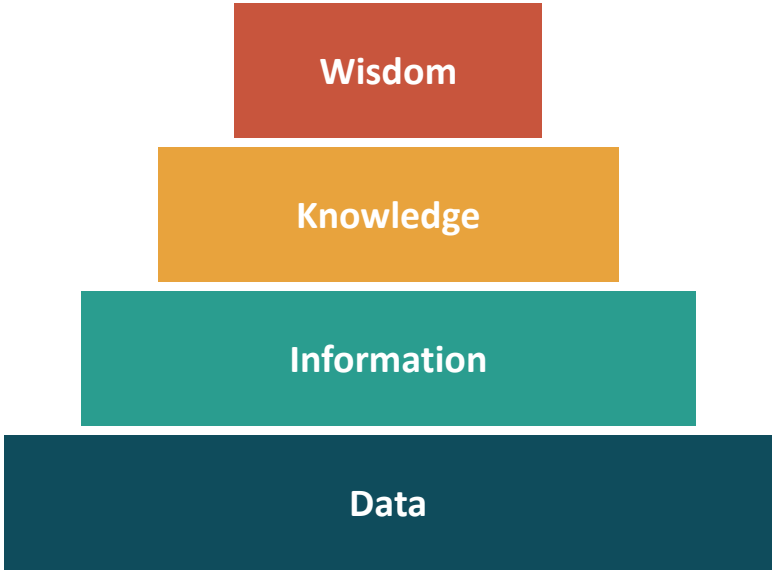
4

Evaluate

How AI and ANN help, and where they fall short.

By the end: you can decide when to use rules, when to use neural networks, and when to combine them.

From data to wisdom (DIKW)



Decide: test, treat or admit, weighing risk and local context

Fever + rash + monsoon in an endemic area suggests dengue

The patient has a fever with a rash

38.9 °C; rash; July

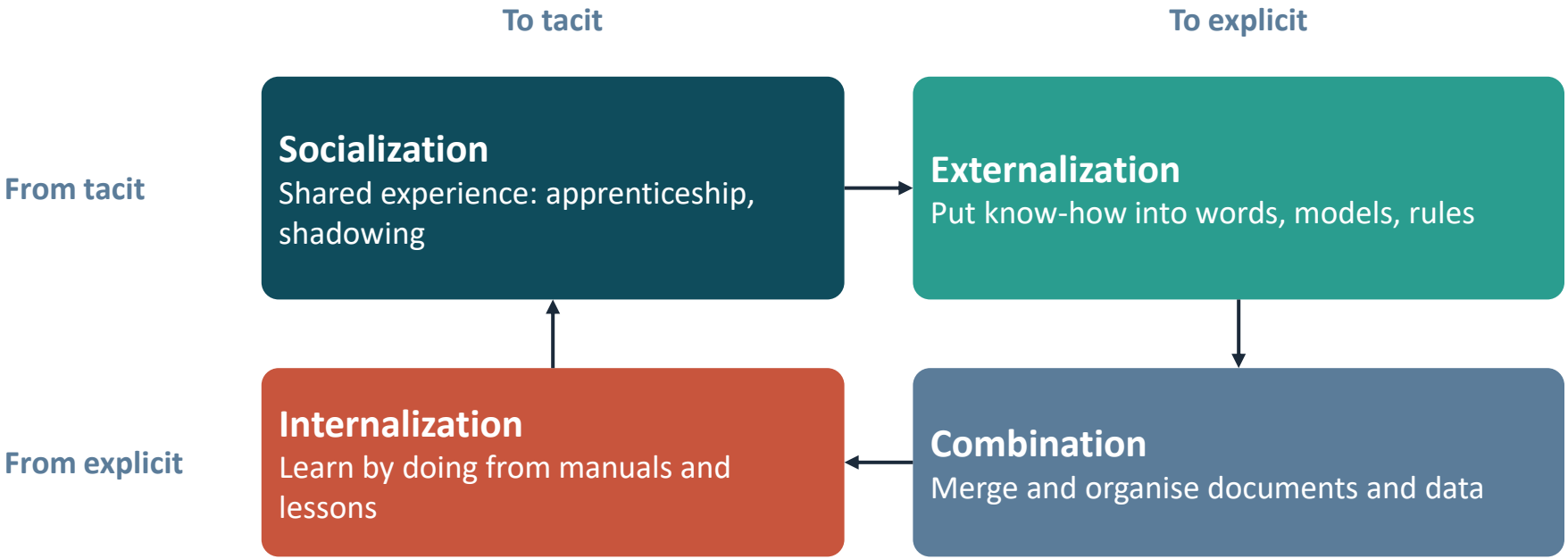
Idea: each step adds context and meaning. Knowledge is what lets an expert turn information into a good decision.

Tacit vs explicit knowledge

	Explicit knowledge	Tacit knowledge
Form	Written, codified	In people's heads and hands
Examples	Manuals, SOPs, code, databases	A mechanic's feel for a failing engine
Sharing	Easy: copy and send	Hard: needs apprenticeship, talk
Risk	Becomes outdated	Leaves when people leave

Risk: much valuable expertise is tacit, so when experienced people leave, the organisation can lose it.

SECI model: how knowledge is created



Case: Matsushita's home bread maker

1

Socialization

Developer Ikuko Tanaka trained under the head baker of a leading Osaka hotel to learn kneading.

2

Externalization

She described the baker's stretch-and-twist technique in words the engineers could use.

3

Combination

Engineers combined that with engineering data to set machine specifications.

4

Internalization

The team tested prototypes, adding ribs inside the machine to copy the technique.

Lesson: the hardest knowledge to capture was tacit. It moved through people before it reached documents and design.

The KM cycle and its technologies



Note: most KM failures are people and incentive problems, not technology problems.

Indian KM cases

Kisan Call Centre

Toll-free helpline launched by the Ministry of Agriculture in 2004. Agricultural graduates answer farmer queries; answers are logged and reused.

Digital Green

Farmer-to-farmer videos spread practices in local languages. It began at Microsoft Research India.

Infosys

Widely cited Indian KM pioneer with a company-wide knowledge portal and a culture of reuse.

TCS mKRISHI

Mobile advisory for farmers, delivering local-language advice from expert knowledge.

Pattern: scarce expert knowledge becomes a shared, reusable service reaching many people.

What is an expert system?

A computer program that captures the knowledge of human experts in a narrow domain and uses reasoning to give expert-level advice, with explanations.

1. **Narrow domain**
   One task done well
2. **Knowledge-based**
   Rules and facts, not just code
3. **Explains itself**
   Answers why and how

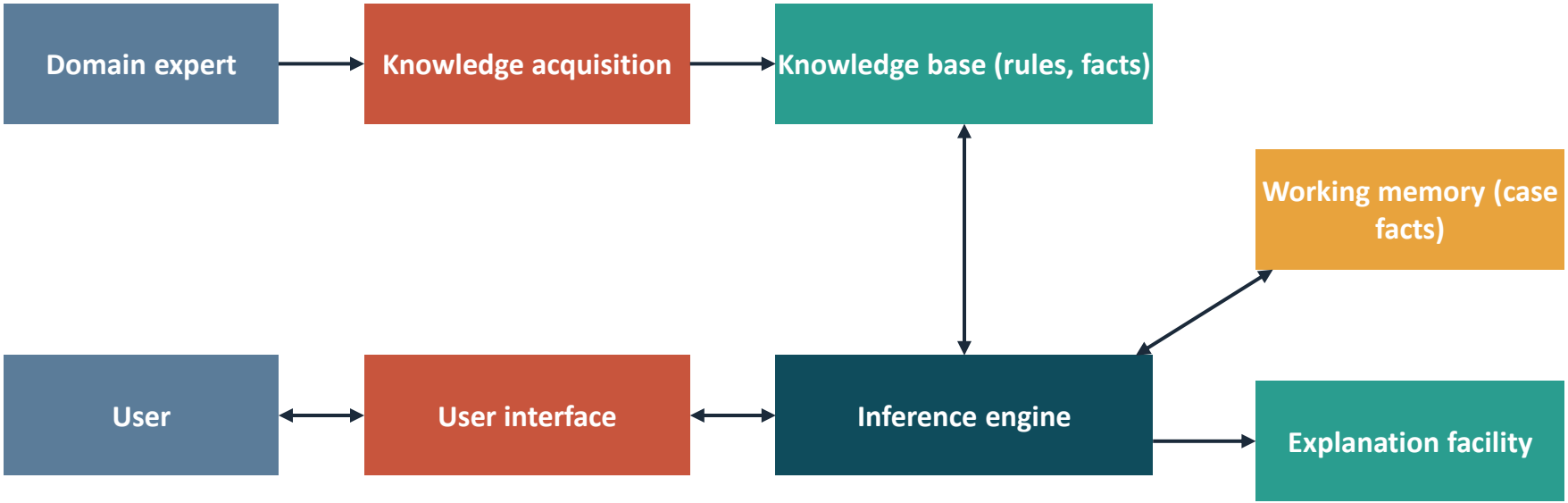
Link to KM: KM captures knowledge for people; an expert system makes captured knowledge executable.

Expert system vs program vs expert

Feature	Conventional program	Expert system	Human expert
Knowledge	Buried in code	Separate knowledge base	Experience
Reasoning	Fixed algorithm	Inference engine	Judgement
Explains?	Rarely	Yes: why and how	Yes
Uncertainty	Poorly	Certainty factors, fuzzy	Naturally
Availability	Always	Always	Limited, tires

Key point: the knowledge base is separate from the engine, so rules can change without rewriting the program.

Expert system architecture



Flow: the engine matches case facts against rules in the knowledge base and can explain how it reached advice.

Knowledge representation

Rules

IF-THEN statements.

IF spots=spindle AND humid THEN blast

Frames

Objects with slots and values.

Crop: name=paddy, stage=tillering

Semantic networks

Nodes and labelled links.

Blast is-a fungal disease; affects paddy

Cases

Stored past problems and fixes.

Ticket: router restarts, fix: update firmware

Choice: pick the representation that matches how the expert naturally explains the knowledge.

Inference: forward and backward chaining

```
R1 IF spots=spindle AND centre=grey AND weather=humid THEN disease=blast
R2 IF disease=blast THEN advice=see agri officer, fungicide, cut nitrogen
```

Forward chaining (data-driven)

- Start with facts: spindle spots, grey centre, humid
- R1 fires, then R2: advice

Backward chaining (goal-driven)

- Start with goal: is it blast?
- Ask for missing facts until R1 is proven or fails

Use: forward chaining for monitoring and configuration; backward chaining for diagnosis.

Handling uncertainty

Certainty factors (MYCIN)

Belief from −1 (false) to +1 (true). Two rules back 'blast' with CF 0.6 and 0.5.

Combined = 0.6 + 0.5 × (1 − 0.6) = 0.80

Fuzzy logic

Replaces yes/no with degrees of truth. 'Humid' may be 0.7 true at 80% relative humidity.

Used in washing machines, air conditioners, metro control

Why: real expert knowledge is rarely certain, so the system must reason with degrees of belief.

Case: MYCIN

What

Stanford system that diagnosed blood infections and recommended antibiotics.

How

Hundreds of IF-THEN rules, backward chaining, certainty factors and WHY/HOW explanations.

Outcome

Matched specialists in evaluations but was never used routinely: legal, ethical and integration barriers.

```
IF   stain = gram-negative AND shape = rod AND aerobic = yes
THEN organism = Enterobacteriaceae (CF 0.8)   [illustrative rule]
```

Legacy: certainty factors, rule-based reasoning, explanations and the EMYCIN shell shaped later systems.

Case: XCON, DENDRAL and PROSPECTOR

XCON / R1

Digital Equipment Corporation, from 1980. Configured VAX computer orders; grew to thousands of rules with large reported savings; hard to maintain.

DENDRAL

Stanford. Worked out molecular structures from mass-spectrometry data. Widely regarded as the first expert system.

PROSPECTOR

SRI. Advised on mineral exploration; widely cited for pointing to a molybdenum deposit in Washington state.

Lesson: narrow domains and clear payoffs work; the hidden cost is maintaining thousands of rules.

Types by knowledge approach

Type	How it reasons	Example
Rule-based	IF-THEN rules	MYCIN, eligibility advisors
Frame-based	Objects with slots	Product and equipment configuration
Case-based	Retrieve and adapt similar cases	Help-desk fault finders
Fuzzy	Degrees of truth	Appliances, Sendai subway control
Neural	Patterns learned from data	Retina and X-ray screening
Hybrid	Combines approaches	Neuro-fuzzy controllers

Trend: modern systems are hybrids: learning for perception, rules for reasoning.

Types by task

Task	What it does	Example
Interpretation	Infers meaning from data	DENDRAL
Diagnosis	Finds the cause of a fault	MYCIN, equipment troubleshooting
Configuration	Assembles parts under constraints	XCON
Prediction	Forecasts outcomes	Crop and weather advisories
Monitoring	Compares behaviour with norms	Plant alarm systems
Advice	Guides users to a decision	Tax and agri-advisory

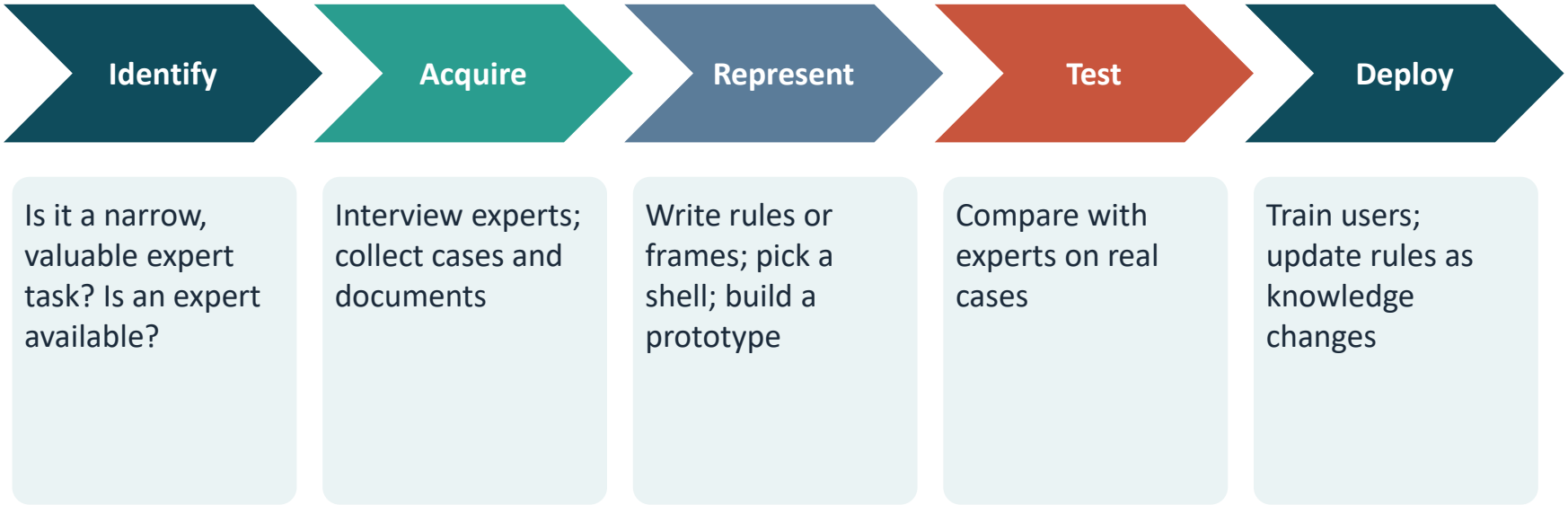
Tip: knowing the task type tells you which reasoning method and tools to use.

Case: match the scenario to the type

Scenario	Best type	Why
Eligibility rules for a government scheme	Rule-based	Explicit, auditable rules
IT help desk using past tickets	Case-based	Similar past cases solve new ones
AC adjusts cooling smoothly	Fuzzy	Gradual degrees, no sharp cut-offs
Spot plant disease from a leaf photo	Neural	Learns visual patterns from data
Triage with an image model and rules	Hybrid	Needs perception and explanation

Try first: cover the last two columns and name the best type for each scenario.

Expert system development life cycle



Iterate: prototype early, show the expert, refine. Maintenance never ends.

Knowledge acquisition: the bottleneck

Technique	How it works	Good for
Interviews	Structured and open questions	Overview and main rules
Observation, think-aloud	Watch and record the expert	Tacit steps
Case review	Walk through real past cases	Exceptions and edge cases
Card sorting, repertory grid	Expert groups and compares concepts	Structure of concepts
Document analysis	Mine manuals and guidelines	Explicit knowledge

Bottleneck: extracting tacit expertise from busy experts is usually the most expensive step.

Tools: building the paddy-disease example

```
(defrule blast
  (spots spindle)
  (centre grey)
  (weather humid)
  =>
  (assert (disease blast)))

(defrule advise
  (disease blast)
  =>
  (printout t "Consult agri officer" crlf))
```

Common tools

- CLIPS (NASA): rule shell
- Jess: Java rule engine
- Prolog: logic programming
- Drools and business rule engines

Reading: when spots, centre and weather facts exist, the first rule asserts blast and the second gives advice.

Role of AI in building expert systems

Learn rules from data

Decision-tree and rule induction turn records into rules.

e.g. ID3, C4.5

Read knowledge from text

NLP extracts facts and rules from manuals and notes.

Knowledge graphs

Link entities for reasoning and search.

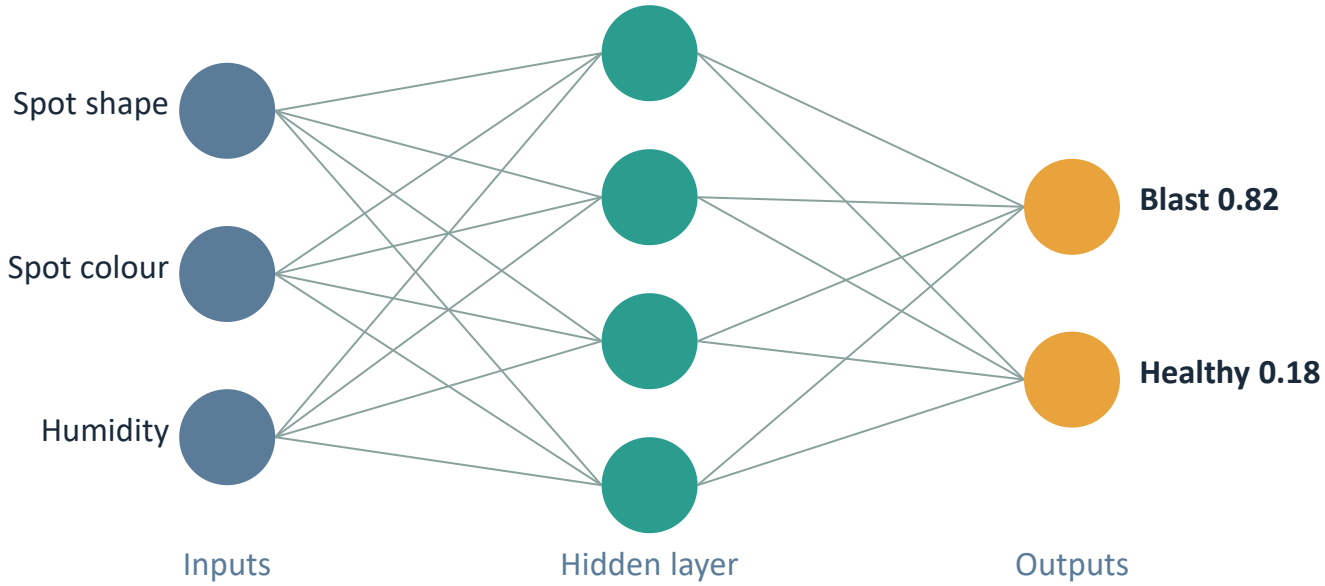
LLM interfaces

Retrieval over guidelines plus plain-language Q&A.

needs guardrails

Idea: AI reduces the knowledge-acquisition bottleneck, but verified rules still anchor trust.

Artificial neural networks in brief



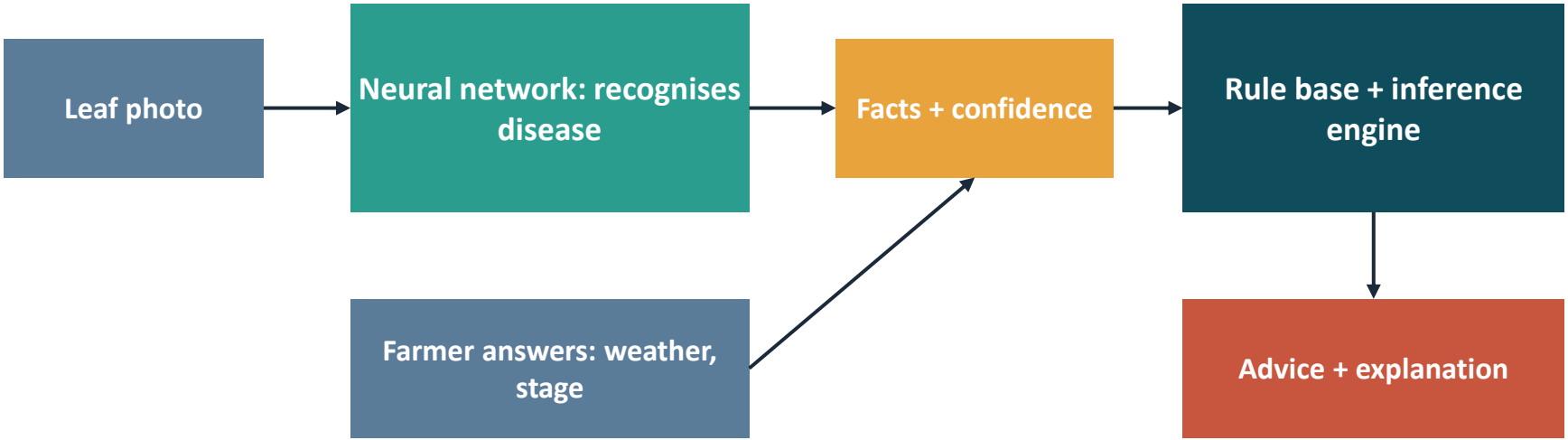
Learning: weights adjust by backpropagation to reduce error on labelled examples.

Rule-based vs ANN-based systems

Aspect	Rule-based ES	ANN-based system
Knowledge source	Experts, written as rules	Data, stored as weights
Explanation	Clear: why and how	Limited: black box
Uncertainty	Certainty factors, fuzzy	Probabilistic outputs
Data needed	Little	A lot
Maintenance	Edit rules by hand	Retrain on new data
Best for	Clear rules, audit	Perception, patterns

Choice: rules when knowledge is explicit and auditable; networks when patterns are in the data.

Hybrid: neural perception plus rules



Idea: neural network for perception, rules for reasoning and explanation: the best of both.

Case: AI eye screening in India

Problem

Diabetic retinopathy can cause blindness if missed. India has a large diabetic population and few retina specialists.

Solution

Google researchers trained a deep neural network on retinal images graded by ophthalmologists. Pilots ran with Aravind Eye Care System and Sankara Nethralaya.

Lesson

Accuracy in studies was high, but field work showed image quality, connectivity and clinic workflow matter as much as the model.

Takeaway: ANN screening extends scarce expertise but needs local validation and workflow design.

Indian examples: AI-assisted expert systems

Qure.ai

Mumbai start-up. AI reads chest X-rays and CT scans to flag conditions such as TB for triage.

Niramai

Bengaluru start-up. Thermal imaging and AI to screen for breast cancer without radiation.

Kisan e-Mitra

Government of India chatbot answering farmers' queries on PM-KISAN in several languages.

Pattern: narrow domain + scarce expertise + AI for scale.

Case: a cautionary tale, Watson for Oncology

What

IBM's system, trained with oncologists at Memorial Sloan Kettering, was adopted by hospitals including Manipal in India.

What happened

Studies reported varying agreement with local tumour boards; media reports raised concerns about some advice.

Lessons

Validate against local practice, be transparent about training data, and avoid hype.

Lesson: an expert system encodes one setting's practice. Always test it in yours.

Responsible use and limits

Explainability

Users need reasons, not just scores.

Bias and drift

Performance falls when population or equipment changes.

Accountability

A named human owns each decision.

Privacy and law

India's DPDP Act 2023 and ICMR 2023 AI guidelines for health.

Rule: keep a human in the loop wherever decisions can harm people.

Class activity (3 minutes, in pairs)

1

Choose

A domain with a clear expert task.

2

Write

Five rules, each with a certainty factor.

3

Decide

Rules, ANN or hybrid? Justify.

4

Source

Who is the expert? Is the knowledge tacit or explicit?

Pick one: crop disease advisor, hostel mess-complaint triage, or loan pre-screening.

Key takeaways

1. Knowledge management captures and shares what organisations know, especially tacit knowledge
2. Expert systems make expert knowledge executable: knowledge base, inference engine, explanation
3. Choose the type by approach and task: rule-based, case-based, fuzzy, neural or hybrid
4. Development is iterative; knowledge acquisition is the bottleneck
5. ANNs add perception and learning; rules add explanation. Hybrids with human oversight work best

Backup: KM challenges and fixes

Challenge	Why it happens	What helps
Knowledge hoarding	Knowledge is power	Reward sharing; recognise contributors
Stale content	No owner, no review	Owners and review dates
Low use	Outside daily workflow	Embed in tools people already use
Security and IP	Sensitive content	Access controls and clear policy

Principle: make sharing easy, visible and rewarded.

Backup: validation and maintenance

Activity	Question	How
Verification	Did we build the system right?	Check rules for conflicts, gaps, loops
Validation	Did we build the right system?	Test on real cases against experts
Maintenance	Does knowledge stay current?	Review rules; track changes; retire old ones
Monitoring	Is it still accurate?	Track errors and user feedback

Lesson: XCON-style rule bases needed continuous, costly maintenance to stay current.