Who Taught the Machine?

AI isn't magic — it learns from examples people choose, so people are in charge of it.

Upper primary · Years 3 to 6

Free preview — C4Sight

In one breath: This is the free preview pack for Who Taught the Machine?, our AI incursion for Years 3 to 6. It gives you one classroom activity you can run yourself today, for nothing, plus what the full visit adds, the curriculum mapping and the prices. Paid visits start at $950, and the first 30-minute Taster is free.

What this series is

Who Taught the Machine? is C4Sight's flagship AI incursion series for Years 3 to 6. In one session, your class trains a real machine-learning tool, watches it get an answer confidently wrong, and works out why. They leave holding an idea most adults haven't been given: an AI doesn't "know" things — it learned patterns from examples people chose, and those people are responsible for what it does.

Your students already use tools with AI inside them, and most of what they've picked up about it is either "it's magic" or "it's scary". This session gives them a working picture of how it actually works, built from what they watch happen in the room.

The session storyline is fixed, and everything in the session serves it:

1. **It knows nothing.** A blank classifier goes up on the big screen. It can't tell a cat from a dog, because nobody has taught it anything yet.
2. **We teach it.** Students hold picture cards up to a camera while the room calls out the categories. Confidence bars move as it learns, live. That's supervised learning, happening in front of them — no jargon needed.
3. **We break it.** We show it something in-between on purpose. It answers "I think this is a CAT, 96% sure" — and it's wrong. The room laughs, then goes quiet.
4. **Why did it break?** Because it only knows what we fed it. It matches patterns from the examples we chose — and the class chose them.
5. **So who's the boss?** If a system is taught unfairly or incompletely, it behaves that way — and that's on the people who taught it, not the machine. The class is in charge, so the class gets the job that comes with it: ask questions, and check its work.

The one big idea: AI isn't magic and doesn't "know" things. It learns patterns from examples people give it, so it's only as good, and only as fair, as what it's taught. People are in charge of it, not the other way around.

The wrong-answer moment — the machine being *confidently wrong* — teaches a second thing your students will keep: something can sound very sure of itself and still be mistaken. A ten-year-old who has watched that happen, and can say why, is harder to fool. The fairness step then makes it real with

true, age-appropriate examples: a voice assistant trained mostly on some accents struggles to understand others; a system taught only one breed of dog may not recognise the rest.

Who is in the room. Two presenters. One is a working AI practitioner — he runs a Perth studio building AI automation for commercial clients, studies law at UWA, and produces C4Sight, a public AI explainer series. He drives the tool and gives the honest answers. The other is a TRBWA-registered teacher who runs the hands-on activities, sets the pace, and keeps every student in the room with it.

The safety promise, in plain words. Every C4Sight session runs this way, no exceptions:

- No student devices. Everything hands-on is unplugged (no screens) or driven by the presenter.
- No student data is entered into any system. Ever.
- No student images are captured, saved, or shown.
- A school staff member is present throughout.
- The camera frames the objects and cards the class holds up — never students' faces.
- The tool runs on the presenter's own device and hotspot, never the school network, and the classifier runs fully offline.

Run this yourself: Guess the Rule

You don't need us in the room to give your class the heart of this series. Here is one complete activity, using nothing but voices and (optionally) some picture cards, in 20 to 25 minutes.

It is built around a trap. Your class will train a "computer" on examples that are all true but all narrow, and it will learn the wrong rule with total confidence. That is not a party trick. It is the single most common way real AI systems fail, and your students will have caused it themselves.

Time	20–25 minutes
Group	Whole class, working in pairs
Year levels	3 to 6
What it teaches	A machine learns a rule only from the examples it is shown. Narrow examples teach a wrong rule, the machine is confident anyway, and better examples fix it.

The trap, in one line (read this before you run it)

The secret rule is **"things that can fly."** But every flying example you feed is a **bird**. So the "computer" learns *bird*, not *flies* — because that is all it was ever shown. Then two held-back cards break it in opposite directions:

- **Butterfly** — it flies, so the true answer is Group 1. A guesser who learned "bird" says Group 2. **Wrong.**
- **Penguin** — it cannot fly, so the true answer is Group 2. A guesser who learned "bird" says Group 1. **Wrong.**

Both mistakes come from the same cause, and neither depends on luck. Get the training examples right and this works every time.

> **The one rule for you:** during training, never feed a flying thing that isn't a bird. No bats, no butterflies, no bees, no aeroplanes. That restraint *is* the lesson.

Materials

Voices alone are enough. Cards make it visible and keep quieter students in the game.

- **Group 1 — the "can fly" pile, all birds on purpose:** parrot, eagle, owl, magpie, cockatoo, seagull.
- **Group 2 — the "cannot fly" pile:** dog, cat, snake, goldfish, kangaroo, echidna.
- **Two test cards, kept out of sight until step 5: butterfly and penguin.**
- One picture and its name per quarter-A4 card, bold and uncluttered. Hand-drawn is fine.

Before you start

Choose a spot where the volunteer can stand out of earshot — back of the room, facing away, is enough if the class whispers. Agree a silent signal for choosing the rule. Keep the two test cards face down under something.

How to run it

1. **Set it up (2 min).** One volunteer becomes "the computer" and moves out of earshot. Everyone else works in pairs. Tell the class the job: *"We are going to teach the computer a rule, and we are never allowed to say the rule. Only examples."*
1. **Show the class the rule, and the trap (2 min).** With the volunteer still away, tell the class the secret rule is **things that can fly**. Then show them the two piles and ask the question that makes them complicit: *"Every flying animal here is a bird. What might the computer think the rule is?"* Take a few answers. Somebody will say "birds". Tell them they are probably right, and that you are going to do it anyway and see what happens. They are now in on it, which is what keeps them watching.
1. **Feed the examples (4–5 min).** Volunteer returns. Each pair agrees on one example and its group before speaking. Call on pairs one at a time, each giving the example **with its group and never the rule**: *"A parrot is in Group 1. A dog is in Group 2. An eagle is in Group 1."* Feed at least eight before any guessing. Say the reason out loud: a computer needs plenty of examples too.
1. **Test it on an easy one (2 min).** Pick a new bird — a cockatoo, say. Before the volunteer answers, the whole room predicts with hands: left for Group 1, right for Group 2. Then the volunteer answers **and says how sure they are out of ten**. They will be right, and they will be very sure. Write the number on the board. That confidence score is the thing to remember.
1. **Now the trap (4 min).** Produce the **butterfly**. Room predicts first, then the volunteer answers and gives a confidence number out of ten. A guesser who learned "bird" says Group 2, usually with a high number. It is wrong — a butterfly flies.

Then produce the **penguin**. Same drill. They will usually say Group 1, and confidently. Also wrong. Let the room enjoy it, then point at the two confidence numbers still on the board. *"It was just as sure as when it was right."*

1. **Work out why — this is the lesson (3 min).** Put it to the pairs for thirty seconds, then take hands: *"Why did it get those wrong?"* Steer until someone says it: every single flying example was a bird, so it learned "bird". Then land it: *"It could not have learned anything else. We never gave it the chance."*

1. **Fix it (3–4 min).** This is the part that makes the activity worth running. Ask: *"What is the smallest number of new cards that would fix it?"* Take suggestions. The answer they are hunting is one flying non-bird — the butterfly — and one non-flying bird — the penguin. Feed those two in with their correct groups. Then test with a fresh case, a bat or a bee, and watch the volunteer get it right. Better examples, better machine. Nobody had to open the computer up.

The teaching point — say it out loud, exactly here

> "The computer wasn't broken and it wasn't silly. It learned exactly what we showed it, and we only showed it birds. Real AI has the same problem: it only knows the examples people chose for it, it is confident either way, and when it goes wrong the fix is usually better examples — chosen by people like you."

Debrief questions

Put each to the pairs first for thirty seconds, then take hands, so everyone has an answer before anyone speaks.

- "When it got the butterfly wrong, was it unsure? What does that tell us about being confident?"
- "Whose fault was the mistake — the computer's, or ours?"
- "We fixed it with two cards. What does that say about who is in charge?"
- "Real AI tools learned from examples too. Who chose them, and what should those people be careful about?"

The last question is where the full series picks up. Give the class a minute to sit with it.

If it wobbles

- **The volunteer says "flies" straight away.** Rare, but it happens with a sharp guesser. Say so honestly: *"This one worked it out with less. Real systems often don't."* Then run step 7 in reverse — ask the class which examples would have *fooled* it, and have them build the trap themselves. Same lesson, arrived at backwards.
- **The class leaks the rule.** Reframe instantly: *"Now it was told the answer. That is copying, not learning. Let's do it properly."*
- **A struggling guesser.** Offer a frame — *"I think Group 1 means ___ because ___"* — and let them try it quietly on the nearest pair first. Then feed more examples. More examples make a better model, which is the lesson anyway.
- **Short on time.** Steps 1, 3, 5 and 6 are the spine. Drop step 4 and step 7.

Run only this, and your class has genuinely met supervised learning, biased training data, the confident mistake, and the repair. It's yours to keep, whether or not you ever book us.

What the full incursion adds

Our own presenters run the activity above mid-session — we're giving it away because it works. Here's what changes when C4Sight comes to your school.

The machine, live. In the incursion, your class trains a genuine image classifier on the big screen instead of a volunteer. Students hold cards to a camera, the room calls the categories, and confidence bars sharpen in real time as it learns. Then it meets a deliberately in-between object and gets it confidently wrong, with a big sure number attached. Everything your class felt in Guess the Rule happens again in front of them, done by a real machine. The best minute of the session is the quiet after the laugh, when a ten-year-old explains bias back to the room in their own words.

Two presenters, working as a pair. A working AI practitioner drives the tool and takes the hard questions honestly — including "how does it really work?" and "should we trust it?". A registered teacher runs the activities, the pace and the differentiation, and watches for who is and isn't following. Together they hold a whole class for an hour.

The same session, adapted to fit your day. The core 60-minute lesson, a 45-minute rotation version for whole-year-level days, and a 90-minute deep-dive that adds the Fairness Sort — a second hands-on activity that turns bias into something students sort with their hands — plus a vetted, framed short video and a "make it fairer" discussion. Differentiation is built in across Years 3–4 and 5–6, with support stems for students who need them and extension questions for the ones who race ahead.

The teacher resource pack. Booked sessions include a pack you keep: a pre-session primer with the handful of words worth introducing first (data, examples, training, fair), a teacher-facing run sheet, a no-technology follow-up activity for afterwards, the vetted video links, and the full descriptor-by-descriptor curriculum alignment mapping.

The safety promise, built in. The safety rules are built into how the tool works, not bolted on afterwards: the camera frames objects only, nothing is recorded or saved, and the classifier runs fully offline on the presenter's own device and hotspot.

Curriculum connections (Australian Curriculum v9)

An incursion enriches the curriculum; it doesn't deliver it, and we won't pretend it does. A one-hour session can't teach a content descriptor to the achievement standard. What it gives you is a vivid, correctly-modelled first meeting with the ideas, for your own teaching to build on. The strength ratings below are deliberately conservative, and every code is checkable. (These are the same codes SCSA implements in WA.)

Years 3–4

Code	Descriptor (AC v9, verbatim)	How the series connects	Strength
AC9TDI4K03	recognise different types of data and explore how the same data can be represented differently depending on the purpose	The heart of the fit: a held-up card becomes image data to the camera, then a pattern, then a confidence number, then a category label — the same object represented four ways, live.	Direct
AC9TDI4K01	explore and describe a range of digital systems and their peripherals for a variety of purposes	The webcam is a textbook peripheral: students watch input hardware feed a system and can describe its job in the pipeline.	Enriches
AC9TDI4P02	follow and describe algorithms involving sequencing, comparison operators (branching) and iteration	Guess the Rule — in this pack and in the session — is a comparison and branching algorithm run by humans, across repeated examples.	Enriches
AC9TDI4P05	discuss how existing and student solutions satisfy the design criteria and user stories	The "why did it break? is it right?" segment is a genuine evaluation: the class judges whether the tool met its job and pinpoints why it did not.	Enriches

Years 5–6

Code	Descriptor (AC v9, verbatim)	How the series connects	Strength
AC9TDI6K03	explain how digital systems represent all data using numbers	The tool turns what the camera sees into numbers and patterns; the confidence bars are the data as numbers. Students see, rather than get told, that the machine's world is numbers.	Direct
AC9TDI6P06	evaluate existing and student solutions against the design criteria and user stories and their broader community impact	The fairness step is exactly this: the class evaluates the tool not only on "did it work" but on who it might treat unfairly and why — broader community impact, made concrete.	Direct
AC9TDI6K04	explore how data can be represented by off and on states (zeros and ones in binary)	The session establishes that everything the machine perceives becomes numbers — the exact hook a teacher needs to then teach binary. The incursion sets it up; your classroom lands it.	Enriches
AC9TDI6P02	design algorithms involving multiple alternatives (branching) and iteration	Guess the Rule embodies a branching rule refined over repeated examples, and "how could we make it fairer?" pushes students to modify it.	Enriches

The heaviest curriculum work sits in three general capabilities, which carry no descriptor codes in v9. **Critical and Creative Thinking**: the wrong-answer moment is set up so the questions — "wait, is it

right?", "why is it so sure?" — come from the students. **Ethical Understanding**: fairness, responsibility and bias at an age-appropriate level — the machine wasn't unfair, the people who taught it were. **Digital Literacy**: a source being sure of itself isn't the same as being right. The session also models the **Australian Framework for Generative AI in Schools** — its transparency, fairness, accountability and privacy principles are visible in how the session runs — and we're happy to talk that through with your leadership team.

The free activity in this pack does honest curriculum work on its own — Guess the Rule is AC9TDI4P02 and AC9TDI6P02 in action. The full descriptor-by-descriptor mapping, including the general-capability continua, ships in the teacher resource pack with a booking.

Where we stand on AI

You are about to put this in front of children, so you are entitled to know what the people who wrote it actually think.

We are not AI evangelists. Caleb uses these tools most working days and still would not call himself a fan. It is a drill: you pick it up when it suits the job and put it down when it doesn't. The moment it starts doing your thinking instead of your work, it has stopped being useful. That is the view underneath every one of these packs, and it is why the activities are built to make students sceptical rather than impressed.

Two things we think schools should keep well away from it: **creativity and judgement.** Drafting is labour and a machine can carry some of it. Deciding is different, and a student who never sits in the not-knowing does not build the muscle that hard problems need later. The struggle was the point of the lesson.

What that means for a C4Sight session, without exception:

- No student devices. Everything hands-on is unplugged, or driven by the presenter.
- No student data is entered into any system, ever.
- No student images are captured, saved or shown. The camera frames only the objects students hold up.
- A school staff member is present throughout.
- The tool runs on the presenter's own device and hotspot, never your network, and the classifier runs fully offline.

And the thing schools least enjoy hearing from us: AI detectors do not reliably work. They flag honest work as cheating and wave real AI writing straight through, and the students who cop it worst are often the ones who speak English as a second language. We will not sell you one, and we will say so in front of your staff. What does work is changing how the task is set, and that is teachable.

All of it is published, dated, and open to being checked: **c4studios.com.au/how-we-use-ai**

If your school would rather keep AI outside the fence entirely, that is a fair position and we will not argue you out of it. We would rather you knew where we stood before you booked us than after.

Bring Who Taught the Machine? to your school

The series comes to you as an incursion, sized to fit how your school runs. All prices exclude GST. Perth metro; travel is free within 30 km of the Perth CBD.

Product	Format	Capacity	Price
Taster	30 min, one class or assembly	Any	**Free**
Single Year Group	2 × 60 min sessions, one presenter	Up to 60	**$950**
Half Day	3 sessions	Up to 90	**$1,750**
Full Day	5 sessions + free 45 min after-school staff briefing	Up to 150	**$2,850**
Term Program	6 weekly sessions with one year group, ends in a showcase	One year group	**$7,500**

Single Year Group, Half Day and Full Day bookings include the teacher resource pack — the pre-session primer, post-session extension activities, and the curriculum mapping. A Half Day covers one year level in rotation; a Full Day covers a whole primary school's upper years, and the free after-school staff briefing means your teachers get something out of the day too.

The free Taster, and how it works. We'll bring the tool and run the real thing for one class or an assembly, free, for 30 minutes. Two honest rules, so you know exactly what you're agreeing to: it's **one per school, ever**, and it **only runs with a decision-maker in the room** — Principal, Deputy, or your Digital Technologies coordinator. The Taster ends with us booking a conversation about whether a paid visit makes sense for your school. If it doesn't, no hard feelings — your students still got the lesson.

Teaching Pre-primary to Year 2? The lower-primary series is **The Computer Puppy** — its free preview pack is a separate download.

Book, or ask us anything:

> → **caleb@c4studios.com.au · c4studios.com.au**

C4Sight · Perth, Western Australia