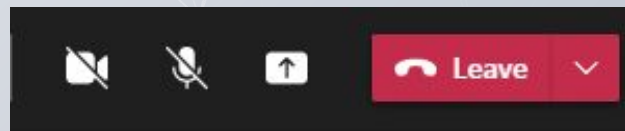
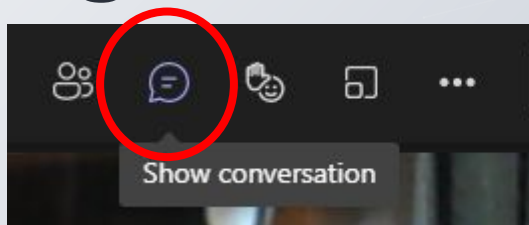


While we wait to get started ...

Open the chat

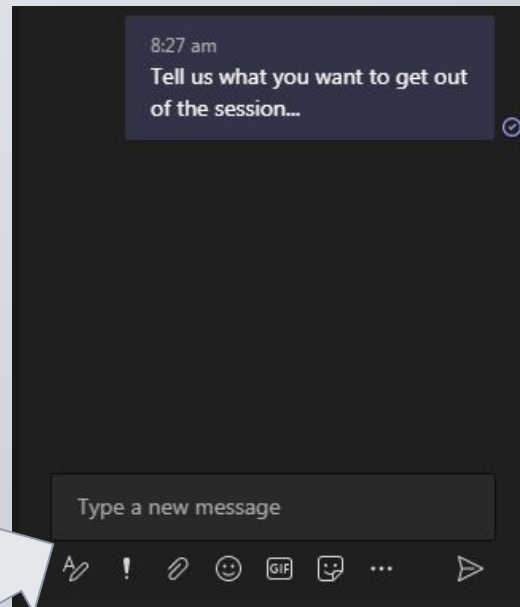


Tell us what you want to get out of the session.

NOTE:

your name will appear with your comment.

The chat won't be part of the recorded version.



Your mic is on mute
... and camera disabled

What context could
you use in your
classroom to induce
creating an AI model?



Discovering Artificial intelligence

Machine learning and creating an AI model



Acknowledgement of Country



ESA acknowledges the Eastern Kulin Nation, Traditional Custodians of the land on which our head office stands, and pays our respects to Elders past and present.

We recognise the Traditional Custodians of Country across Australia and their continuing connection and contribution to lands, waters, communities and learning

By the end of this session...

You should be able to:

Describe how to train an AI and choose a context for students to create their own model.

Develop an understanding of bias and how to limit it in their AI model.

Use an AI model in a Scratch 3.0 project.

Achievement standards: starting point

Achievement Standard

By the end of Year 2, students identify systems (hardware and software) purposes. They use digital system patterns in **data** in different ways.

Students design solutions to simple problems, design and implement digital systems that involve decision-making and how the solutions meet their purposes. They evaluate different **data** when creating solutions. They safely use and manage information systems and their solutions meet needs and consider sustainability. Students manage the creation and communication of ideas and information in collaborative digital projects using validated **data** and agreed protocols.

Achievement Standard

By the end of Year 6, students explain the fundamentals of **digital system components** (hardware, software and networks) and how digital systems are connected to form networks. They explain how digital systems use whole numbers as a basis for representing a variety of **data** types.

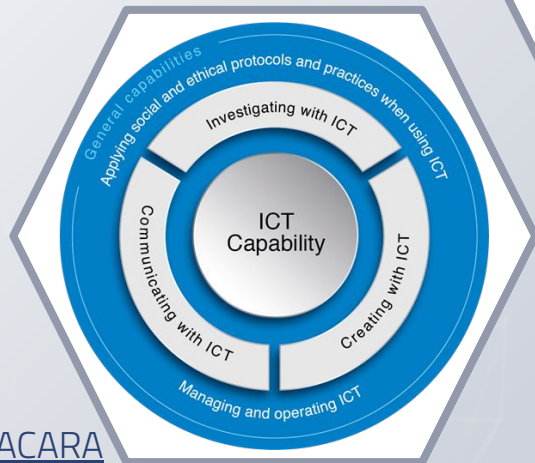
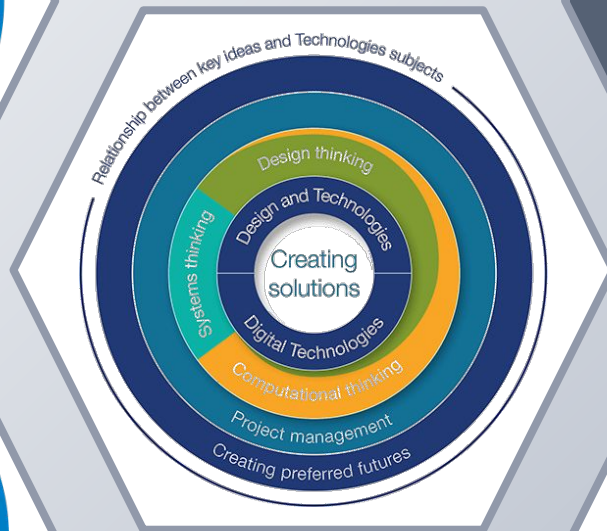
Students define problems in terms of **data** and functional requirements and design solutions by developing algorithms to address the problems. They incorporate decision-making, repetition and **user interface** design into their designs and implement their digital solutions, including a visual program. They explain how information systems and their solutions meet needs and consider sustainability. Students manage the creation and communication of ideas and information in collaborative digital projects using validated **data** and agreed protocols.

Achievement Standard

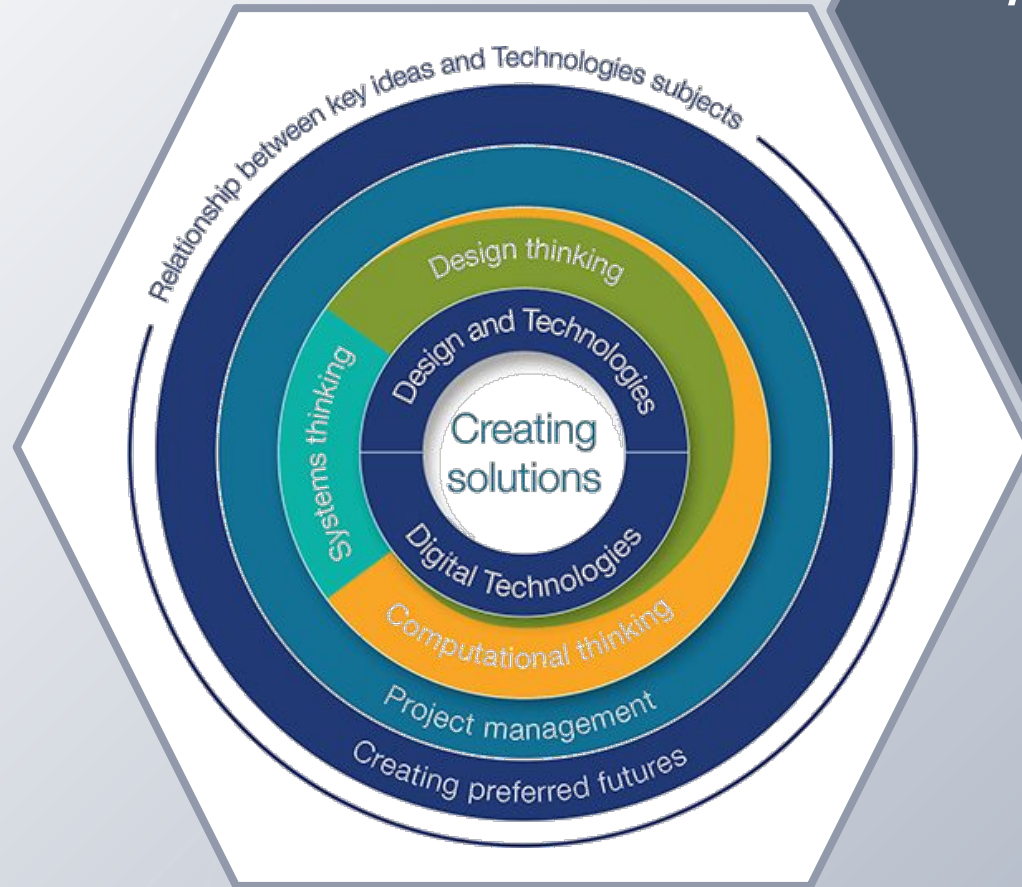
Students describe how a range of digital systems (software) and their peripheral devices are used for different purposes. They explain how the same systems can be used in different ways.

Students define problems, design and implement digital systems that involve decision-making and how the solutions meet their purposes. They evaluate different **data** when creating solutions. They safely use and manage information systems and their solutions meet needs and consider sustainability. Students manage the creation and communication of ideas and information in collaborative digital projects using validated **data** and agreed protocols.

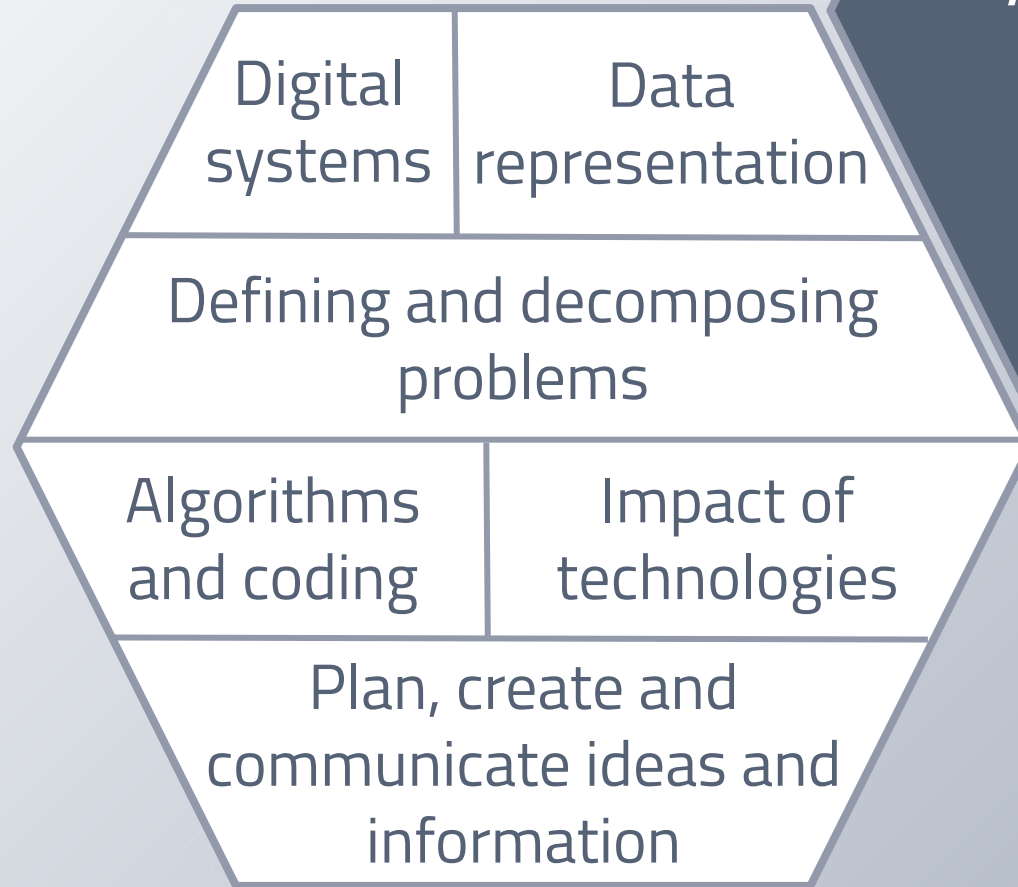
AI topics



AI topics



AI topics



AI topics

Image recognition

Text & speech recognition

Creating & using AI
models (machine learning)

Bias and ethical issues

Digital
systems

Data
representation

Defining and decomposing
problems

Algorithms
and coding

Impact of
technologies

Plan, create and
communicate ideas and
information

Systems Thinking



Design Thinking

Computational Thinking





What is an AI model?

An AI (artificial intelligence) model is a program that has been trained on a set of data (called the training set) to recognize certain types of patterns.

Image CC-BY-SA NDB Photos ([Wikimedia Commons](#))



Learning

So the AI is a program that can learn.

- It needs **learning data** to carry out the learning process.
- And (usually) our **feedback** to get it right
- We use **test data** to check if the learning was successful



Let's make an AI model

LESSON: Can AI guess your emotion?

(Years F-4)

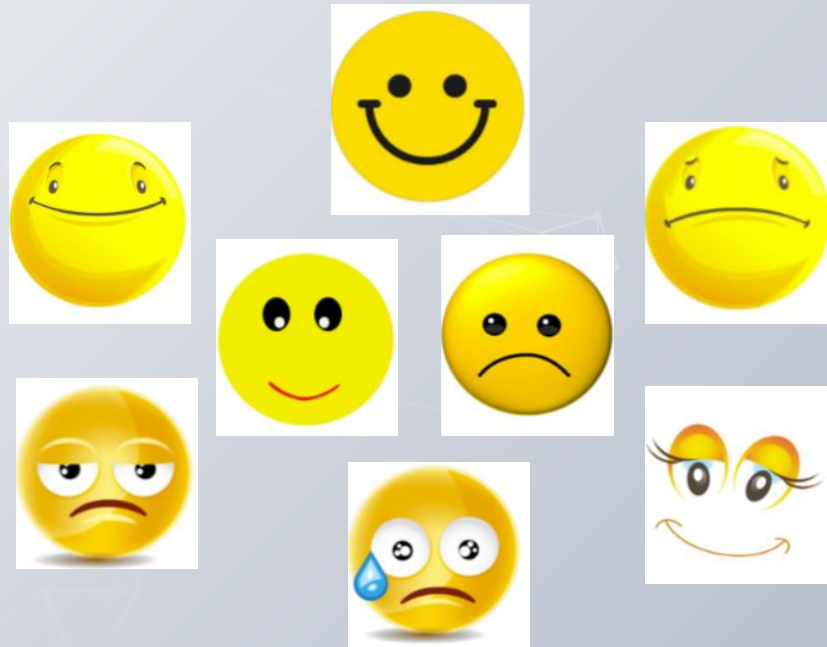
Objective

Build an AI model that can distinguish happy and sad emojis.



1) Collect Learning Data

We collect images of happy and sad emojis



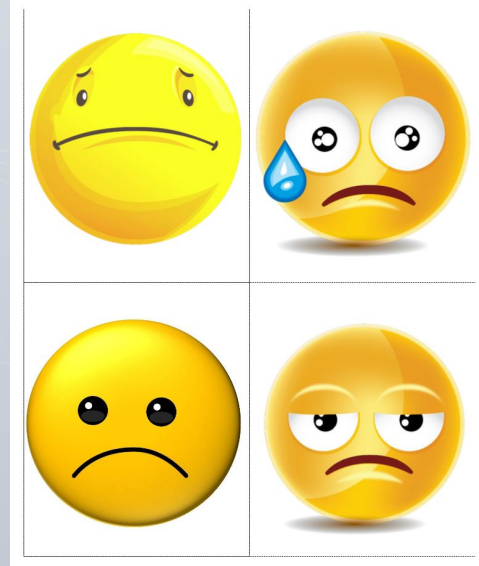
2) Group the learning Data

We then group and label the learning data, for example by placing the images in labeled buckets

happy



sad

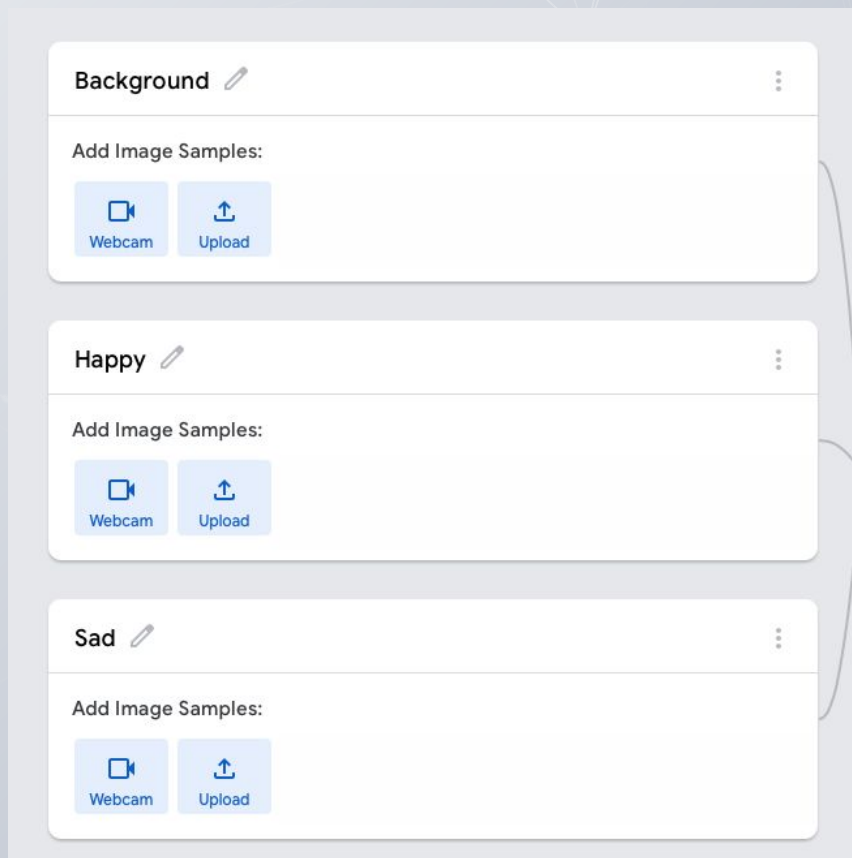


3. Configuring the AI

Set up the groups in the AI.

These are **Happy** and **Sad**.

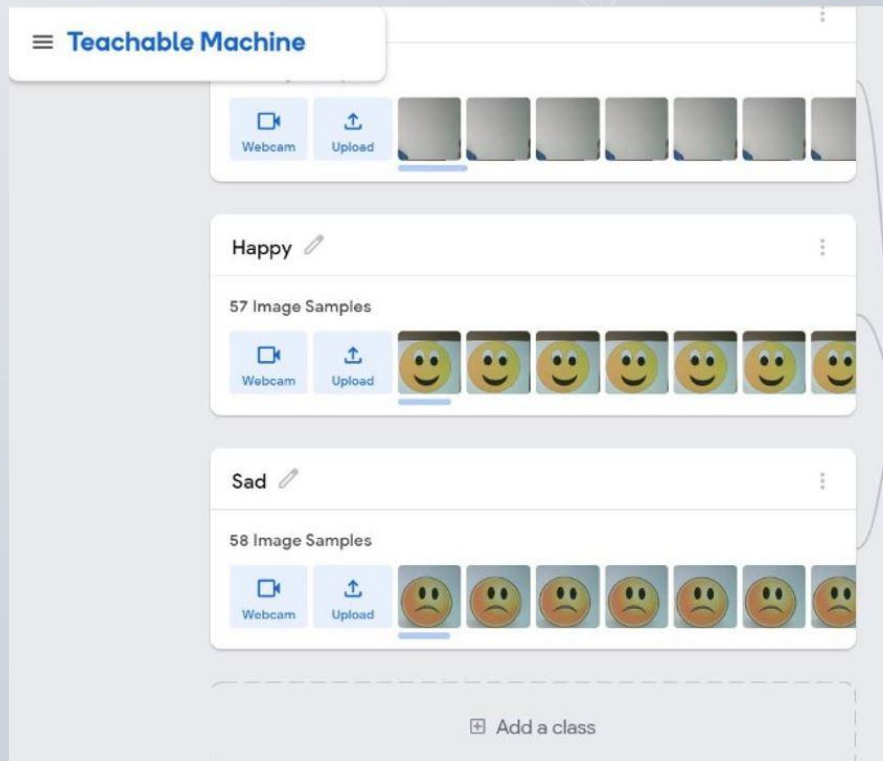
Also, add a **Background** Group



4. Add the images to the AI

Record the images under their respective groups, **Happy** or **Sad**.

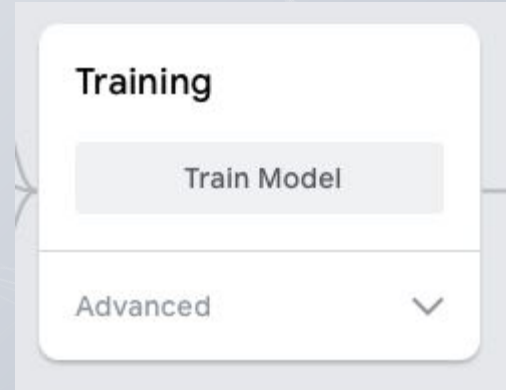
In the **Background** group, record images of the background. This is our control group.



5. Start the learning process

Tell the AI to begin learning.

Internally, the AI will begin a process of adjusting its internal values. Teachable Machine doesn't show this, but you can view this in the [My Computer Brain AI](#).



6) Learning process complete

We have given all these different images to the AI and told it to learn. Because we have labeled the images, the AI knows what they mean. (supervised learning)

The screenshot displays the Teachable Machine web application interface. On the left, the 'Training' section shows two classes: 'Happy' and 'Sad'. The 'Happy' class has 57 image samples and is circled in red. The 'Sad' class has 58 image samples and is also circled in red. Each class has 'Webcam' and 'Upload' buttons. A 'Model Trained' button is visible in the center. On the right, the 'Preview' section shows a large yellow smiley face emoji on the input screen. The 'Output' section shows three colored bars: 'Backg...' (orange), 'Happy' (pink, 100%), and 'Sad' (purple). The 'Happy' bar is highlighted, indicating the model's prediction.

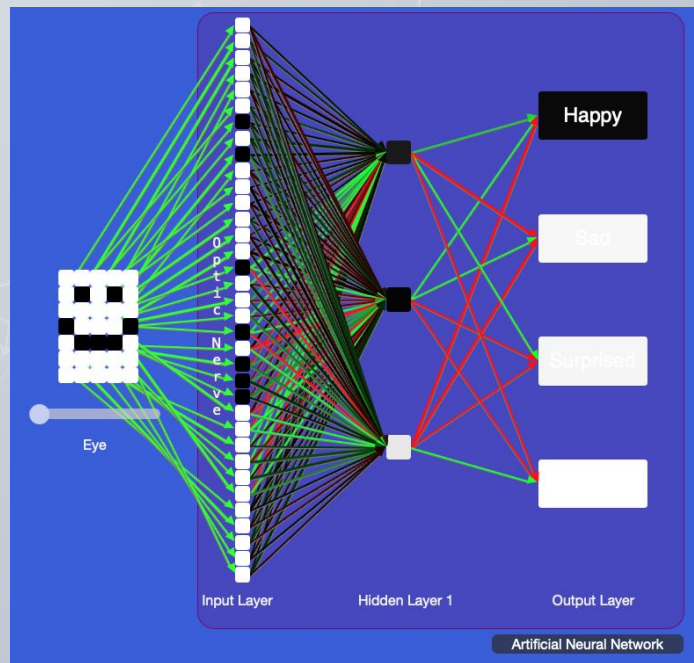
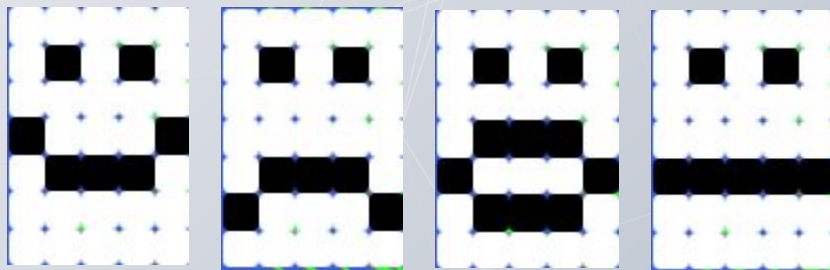
Science Alert



What actually happens in an AI when it learns?

What is learning anyway?

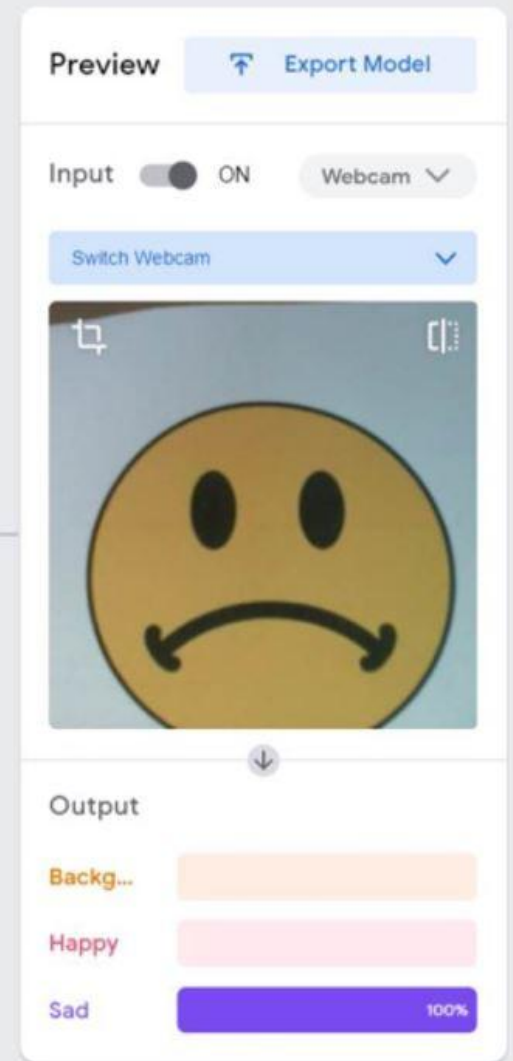
Demo



Testing the model

We then test our model

For the test, we use different data



LESSON: Can AI guess your emotion?

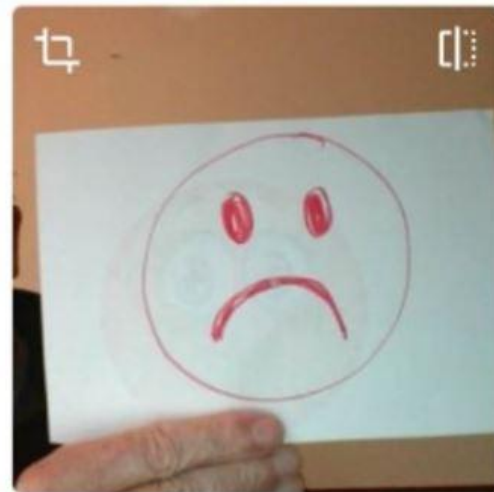
(Years F-4)

Confidence

The AI tells us how likely it is that the input belongs to a certain class.

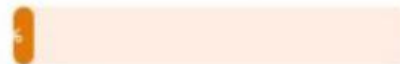
Here, the AI is 17% certain that this is a happy face, but 78% that it is a sad face.

Preview this model live



Output

Backgr...



Happy



Sad



Confidence issues


The AI can be confident, but still be wrong.

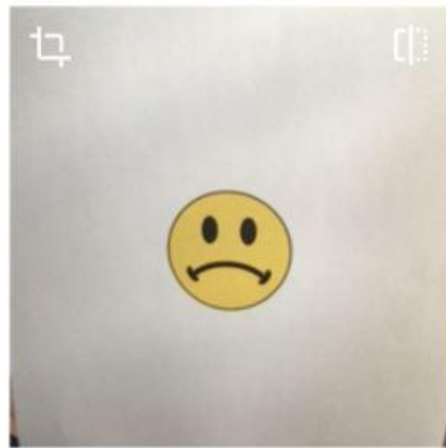
Here, the AI is 93% confident that it sees a happy face, yet it is wrong.

Our AI model has a problem.

Preview this model live

Input: ☒ ON

Webcam 

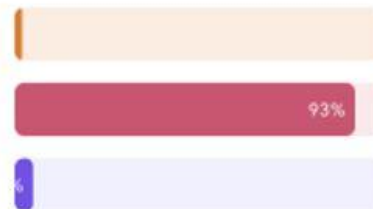


Output

Backgr...

Happy

Sad



Creating & training an AI model



Use this pre-made model to test the AI to see how well it recognises happy and sad. (You will need a device with a webcam).

Bias

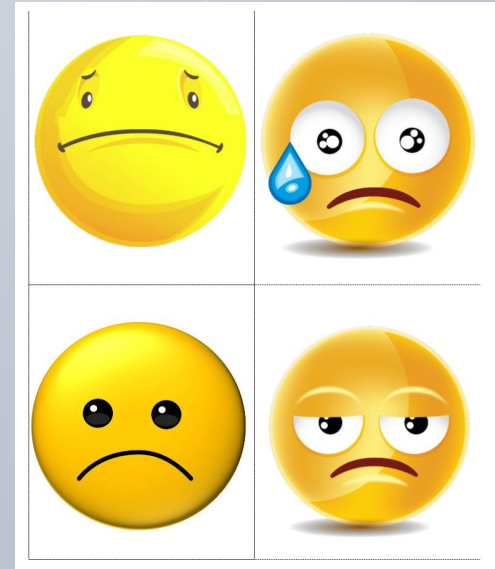
For an AI system to be unbiased requires the training data to be balanced.

Bias can be intentional, but often creeps in unintended.

Bias

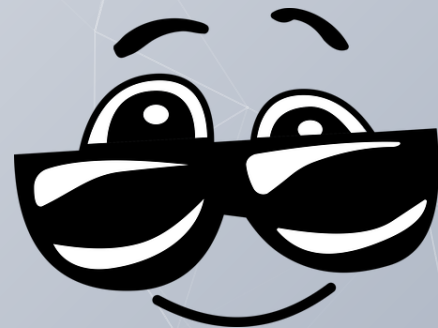
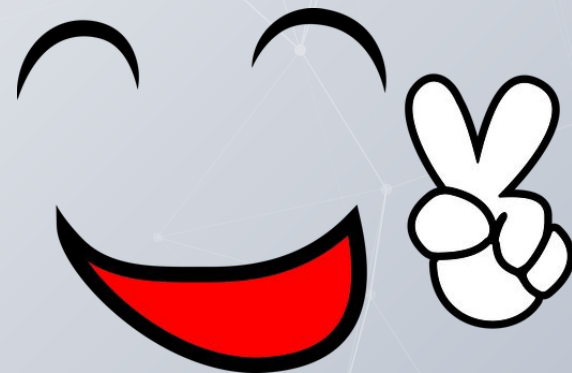
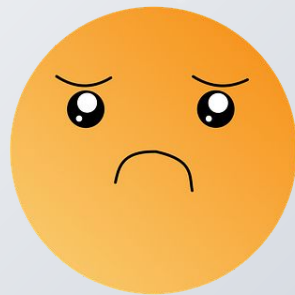
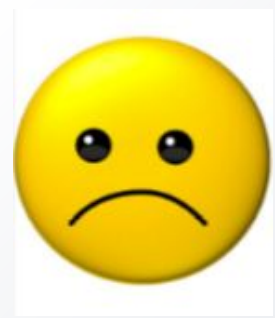
Our model has a a number of biases:

- Shape
- Colour
- Size
- Backgrounds
- Rotation
- ~~Eye, mouth shape ...~~



Activity

How could we address data bias in our model?

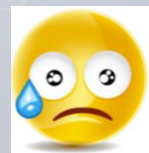
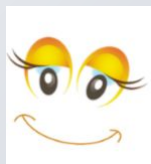
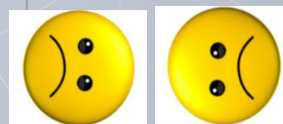
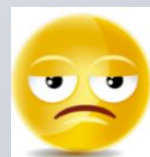
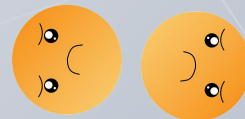
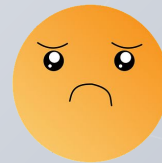
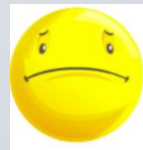


Retraining our model

By following the previous steps, we can re-train our model with the additional data

Happy

Sad



Bias.

It is quite normal for initial AI models to have a bias. It takes some experience to anticipate it.

It even happens to the professionals.

Note that we need to combine bias dimensions in our learning data for a better balanced model:

E.g.

- round-yellow-large-white background - **upright**
- round-yellow-large-white background - **rotated 90 degrees**



Bias - Summary

Is hard to avoid

Often creeps in unintended

Is a great hook to discuss limitations of AI, fairness and ethics with students.

We'll come back to this topic in Deep dive 5, AI and Ethics.

Creating an AI model

What contexts would engage your students when creating an AI model?

Put your idea in the chat!

Creating an AI model: engaging contexts

Possible contexts

- Paper, scissors, rock game
- AI fruit and veg identifier
- Cat sensor cat door (no dogs allowed!)
- Landmark identifier
- Flag recogniser
- Litter sorter (waste, recyclable, organic)

Creating an AI model: engaging contexts

How can we avoid bias when creating these models?

- Paper, scissors, rock game
- AI fruit and veg identifier
- Cat sensor cat door (no dogs allowed!)
- Landmark identifier
- Flag recogniser
- Litter sorter (waste, recyclable, organic)

Data: sourcing images

Recognise intellectual property

Part of **Applying social and ethical protocols and practices when using ICT**

Provide links to image free library or provide a folder of images (teacher curated)

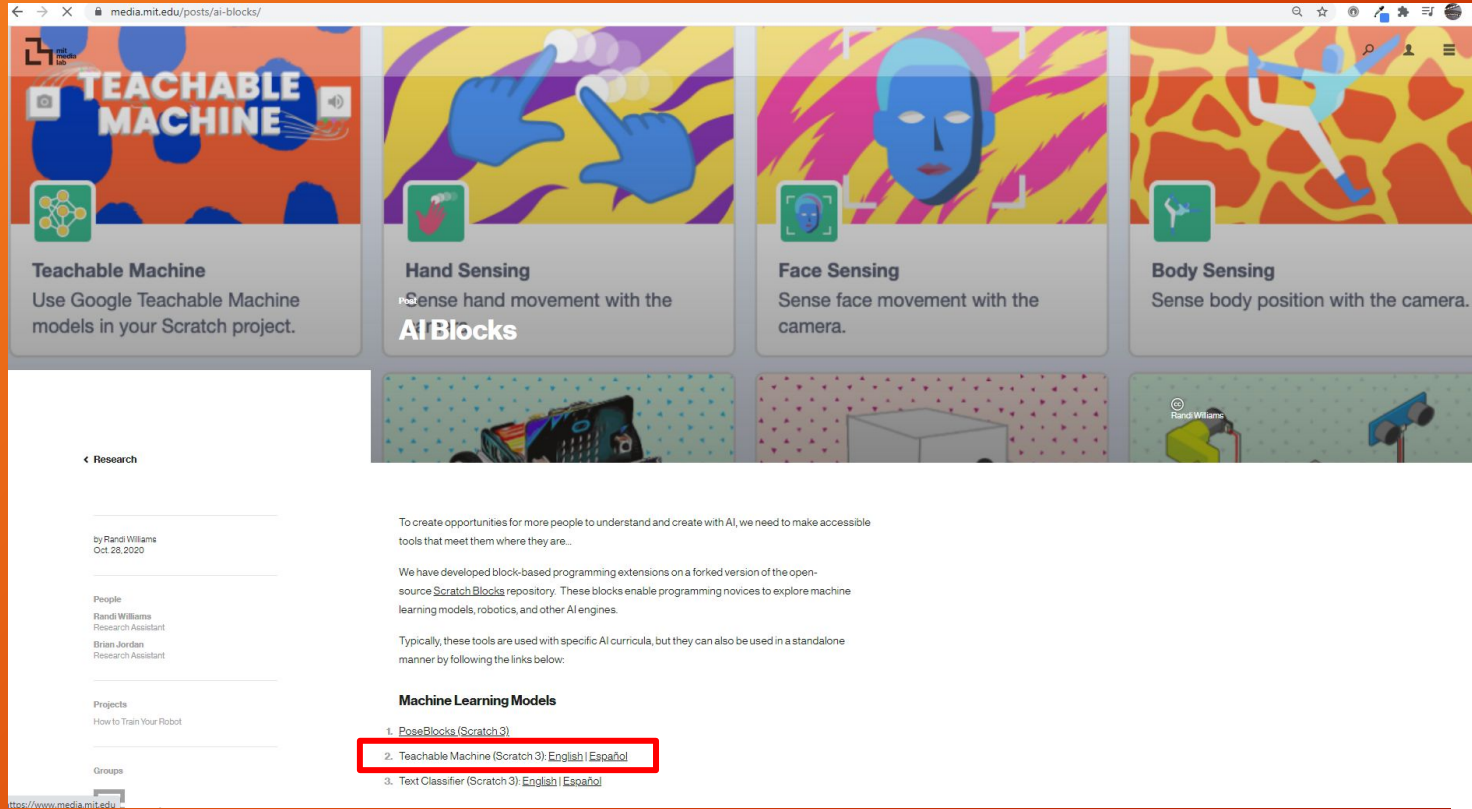


Image recognition

The ability of machines to recognise objects in images or videos (part of **computer vision**).

A new version of Scratch that is AI compatible!

Created by
MIT Media Lab



The screenshot shows a web browser displaying the MIT Media Lab website. The main content area features a grid of project cards. The first card is titled 'TEACHABLE MACHINE' and describes using Google Teachable Machine models in a Scratch project. The second card is titled 'Hand Sensing' and describes sensing hand movement with AI Blocks. The third card is titled 'Face Sensing' and describes sensing face movement with the camera. The fourth card is titled 'Body Sensing' and describes sensing body position with the camera. Below the grid, there is a section titled 'Machine Learning Models' with a list of projects. The second item in the list, 'Teachable Machine (Scratch 3): English | Español', is highlighted with a red box. The left sidebar contains a 'Research' section with a list of people, including Randi Williams, Brian Jordan, and Randi Williams.

media.mit.edu/posts/ai-blocks/

TEACHABLE MACHINE

Use Google Teachable Machine models in your Scratch project.

Hand Sensing

Sense hand movement with the AI Blocks

Face Sensing

Sense face movement with the camera.

Body Sensing

Sense body position with the camera.

Research

by Randi Williams
Oct 28, 2020

People

Randi Williams
Research Assistant

Brian Jordan
Research Assistant

Projects

How to Train Your Robot

Groups

Machine Learning Models

1. PoseBlocks (Scratch 3)
2. Teachable Machine (Scratch 3): English | Español
3. Text Classifier (Scratch 3): English | Español

<https://www.media.mit.edu/posts/ai-blocks/>



A new version of Scratch that is AI compatible!

The screenshot displays the Scratch AI extension interface. On the left, the 'Scripts' area contains three event-driven code blocks: 'when clicked' (turn video on, use model URL), 'when model detects Soft drink bottle' (say 'That looks a soft drink bottle' for 2 seconds, say 'That goes in recycling bin' for 2 seconds), and 'when model detects Chip/lollies packet' (say 'That looks like a chip packet' for 2 seconds, say 'That goes in rubbish bin' for 2 seconds). Below these are two more event-driven blocks: 'when model detects Banana skin' (say 'That looks like a banana skin' for 2 seconds, say 'That goes in green waste or compost' for 2 seconds). The right side shows a video prediction window titled 'Teachable Machine: model prediction' with a 'Chip/lollies packet' label. A Scratch character is overlaid on the video, saying 'That looks like a chip packet'. The bottom of the interface shows the 'Sprite' area with 'Sprite1' selected, and the 'Stage' area with 'Backdrops' set to 1.

Paste the model's unique URL.

Add some code blocks and you are away.

It is that simple!

<https://mitmedialab.github.io/prg-extension-boilerplate/create/>

Training an AI model

The screenshot displays the Teachable Machine web interface. On the left, there are four class categories: 'Soft drink bottles' (26 Image Samples), 'Banana skins' (7 Image Samples), and 'Background' (12 Image Samples). Each category has 'Webcam' and 'Upload' buttons. A red rectangle highlights the 'Training' panel in the center, which contains a 'Train Model' button and an 'Advanced' dropdown menu. To the right of the 'Training' panel is a 'Preview' panel with an 'Export Model' button and a message: 'You must train a model on the left before you can preview it here.' The bottom of the interface shows a language selector set to 'English' and a version number 'release-2.4.4 - 2.4.4#95c54c'.

TIPS...

Have all your
data samples
ready

Have your
classes
(buckets)
worked out

<https://teachablemachine.withgoogle.com/>

A trained AI model

The screenshot displays the Teachable Machine web interface. On the left, four classes are listed: 'Banana skin' (93 Image Samples), 'Soft drink bottle' (167 Image Samples), 'Chip/lollies packet' (183 Image Samples), and 'Background' (24 Image Samples). Each class has a 'Webcam' and 'Upload' button, and a row of sample images. A 'Training' panel in the center shows 'Model Trained' and 'Advanced' options. On the right, a 'Preview' panel shows a live webcam feed of a bottle and the model's output. The output shows 'Soft drink bottle' with a 99% confidence score. At the bottom, there is a language dropdown set to 'English' and a version number 'release-2-4-4 - 2.4.4#95c54c'.

Teachable Machine

Banana skin ✎

93 Image Samples

Webcam Upload

Soft drink bottle ✎

167 Image Samples

Webcam Upload

Chip/lollies packet ✎

183 Image Samples

Webcam Upload

Background ✎

24 Image Samples

Webcam Upload

Add a class

Add a class

Training

Model Trained

Advanced

Preview Export Model

Input ON Webcam

Output

Banana skin

Soft drink bottle 99%

Chip/l... packet

Backg...

English

release-2-4-4 - 2.4.4#95c54c

Have something
in place for the
time taken to
train the model.
... wait

<https://teachablemachine.withgoogle.com/models/uWryl1Rue/>

Exporting a trained AI model

Teachable Machine

What is this?

This link hosts a machine learning model created using Teachable Machine, a tool that makes it easier for anyone – teachers, students, artists, makers of all kinds – to train machine learning models.

How does it work?

Machine learning models are trained on examples (e.g., images, sounds, poses) gathered by the creator. Their results depend on the data they've been trained on.

Want to use this model in your project?

See [this link](#) to learn how to use Teachable Machine models in your projects.

Report this model:

If you have concerns about this model, report it using [this form](#).

This model:

[teachablemachine.withgoogle.com/models/uWryl1Rue/](#)

[/model.json](#)

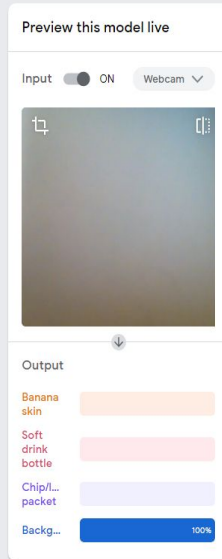
The model architecture, used by TensorFlow.js library

[/metadata.json](#)

Contains the model metadata, for example class labels and version of library

[/model.weights.bin](#)

TensorFlow.js binary file containing the model weights

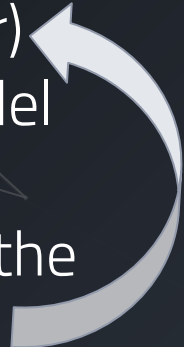


Copy and save the URL.

Test in new browser

<https://teachablemachine.withgoogle.com/models/uWryl1Rue/>

5 step process: Using an AI model in Scratch

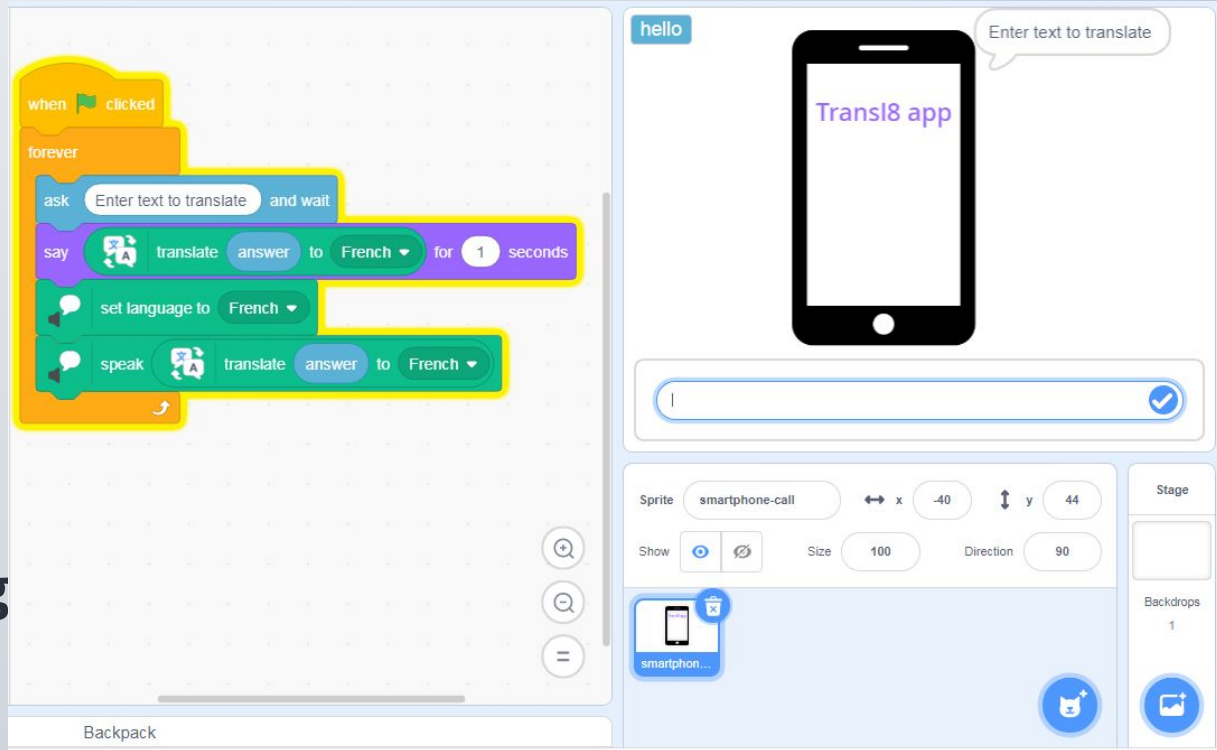
1. Define the problem and data requirements
 2. Collect the data (folder)
 3. Train and test the model (copy URL of model)
 4. Design and represent the algorithm
 5. Build the program (user input) then modify to include AI model (URL)
- 

Design thinking

Build on this basic translating app to include other languages.

We draw on:

- **design thinking** skills for layout (how it will look)
- **computational thinking** to program the solution (how it will work).



Design our algorithm

START...

Repeat

User selects from three options (Flags)

Check selection: yes or no

If answer = yes

Ask for text to be translated

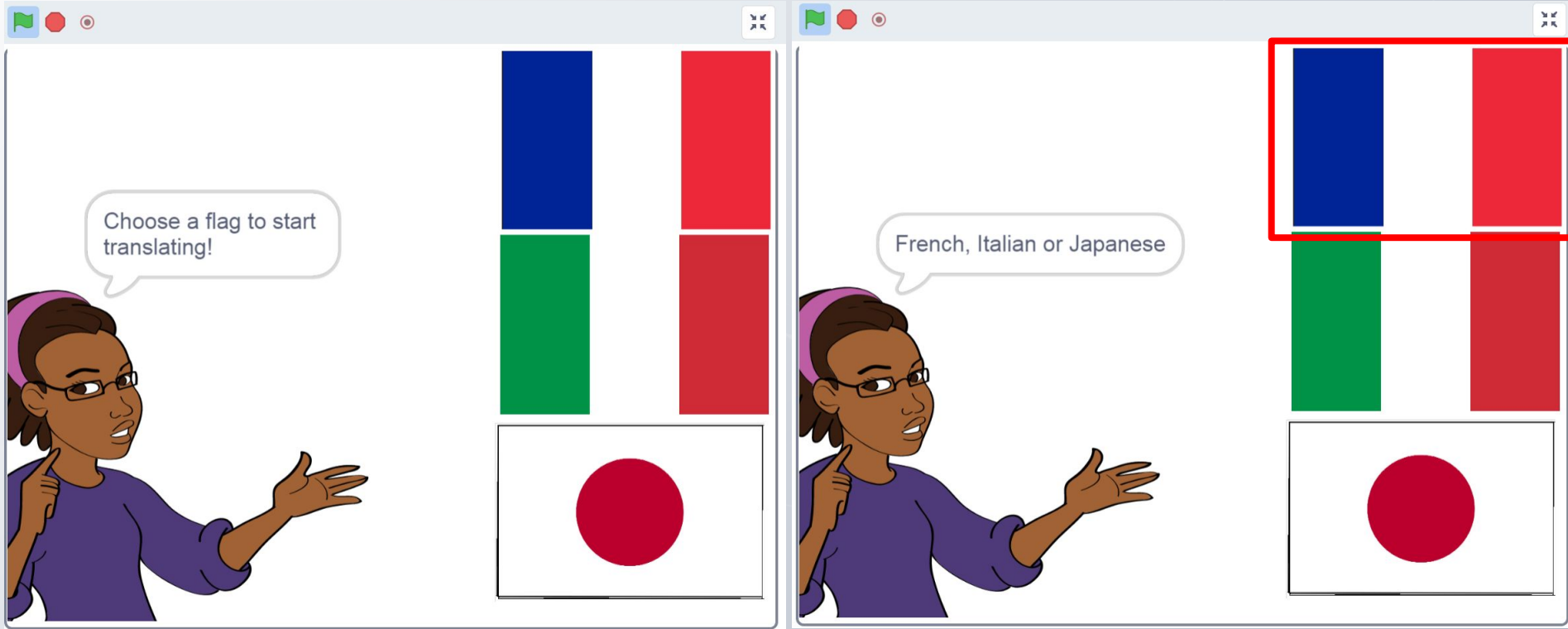
Say (Print) translated text

Speak translated text

END If (answer = no)

END Repeat

Design thinking: user input

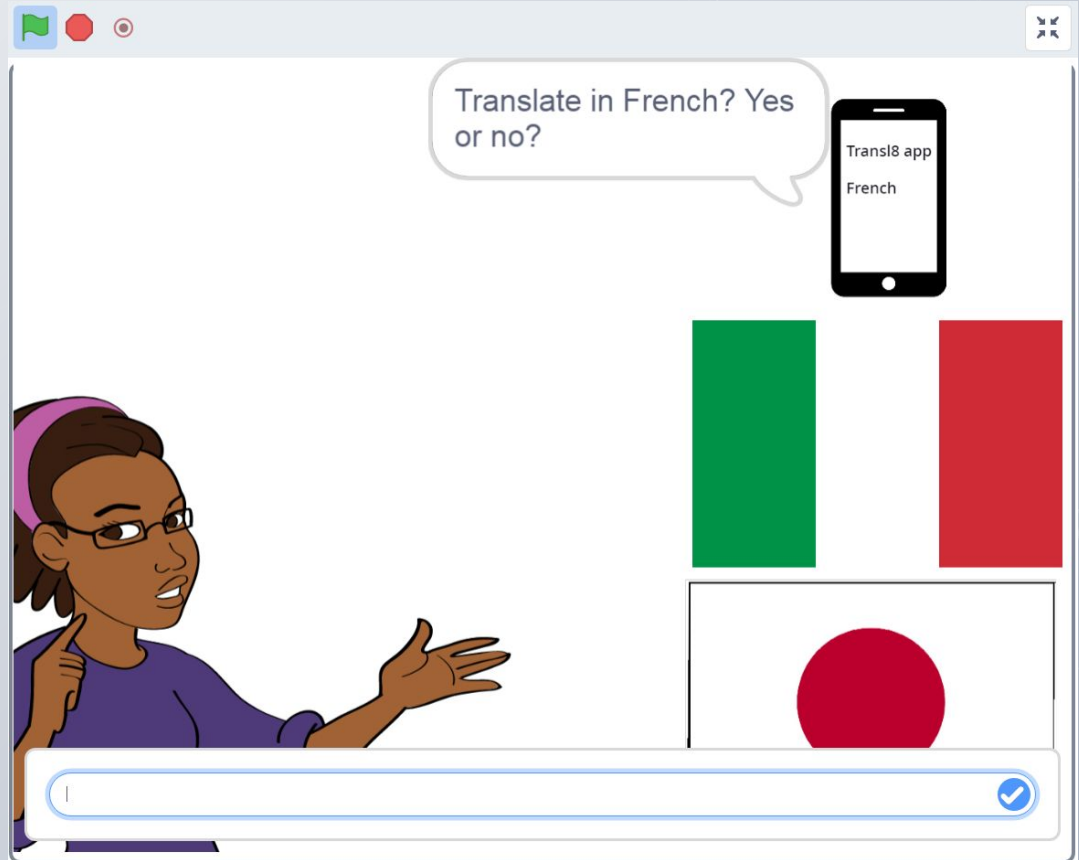


Design thinking: user input

When sprite is clicked...
check whether user
meant to choose the
language.

If **no...** go back. (also
use as the way to exit
the language)

If **yes...** start to
translate.



Design thinking

The image displays a Scratch project for a Japanese translation application. The script on the left is triggered by a 'when this sprite clicked' event. It switches the costume to 'smartphone-Japanese', shows it, and enters a 'repeat until' loop where the answer is 'no'. Inside the loop, it asks 'Translate in Japanese? Yes or no? and wait'. If the answer is 'yes', it asks for text to translate, says the translated text for 0.25 seconds, sets the language to Japanese, and speaks the translated text. If the answer is 'no', it switches the costume to 'Japanese flag'. Two yellow callout boxes provide instructions: 'Check if the user selected this language. Option to go back. (select no). If yes start to translate.' and 'Answering no gives the user the option to stop and select a different language back to start. The end of translating in this language is a change back to flag.'

The stage on the right features a character pointing at a screen displaying the Japanese flag, which is highlighted with a red rectangle. Above the screen are three vertical bars in blue, green, and red. The bottom interface shows the 'smartphone-call3' sprite with a size of 35 and a direction of 90. The 'Backpack' at the bottom contains three smartphone sprites with different flags (French, Italian, Japanese) and a character named Avery.

Source: Flagpedia

Design thinking

The image displays a Scratch project for a Japanese translation application. The script on the left is triggered by a 'when this sprite clicked' event. It begins with a red box highlighting the 'switch costume to smartphone-Japanese' and 'show' blocks. This is followed by a 'repeat until' loop where the condition is 'answer = no'. Inside the loop, the user is asked 'Translate in Japanese? Yes or no? and wait'. If the answer is 'yes', the user is asked to 'Enter text to translate and wait', then the text is translated to Japanese (displayed for 0.25 seconds), the language is set to Japanese, and the translation is spoken. If the answer is 'no', the costume is switched to 'Japanese flag'. Two yellow callout boxes provide context: one explains the language selection logic, and the other explains the 'no' response logic. The stage on the right features a character pointing at a background of Japanese flags (blue, white, red, green, red). A smartphone sprite displays 'Transl8 app Japanese'. The bottom interface shows the 'smartphone-call3' sprite selected, with its position at (149, -112) and size 35. The 'Backpack' at the bottom left contains three smartphone sprites and a character named Avery.

when this sprite clicked

switch costume to smartphone-Japanese

show

repeat until answer = no

ask Translate in Japanese? Yes or no? and wait

if answer = yes then

ask Enter text to translate and wait

say translate answer to Japanese for 0.25 seconds

set language to Japanese

speak translate answer to Japanese

if answer = no then

switch costume to Japanese flag

Check if the user selected this language

Option to go back. (select no)

If yes start to translate.

Answering no gives the user the option to stop and select a different language back to start.

The end of translating in this language is a change back to flag.

Sprite smartphone-call3 x 149 y -112

Show Size 35 Direction 90

Backpack

smartpho... smartpho... smartpho... Avery

Stage

Backdrops 1

Design thinking

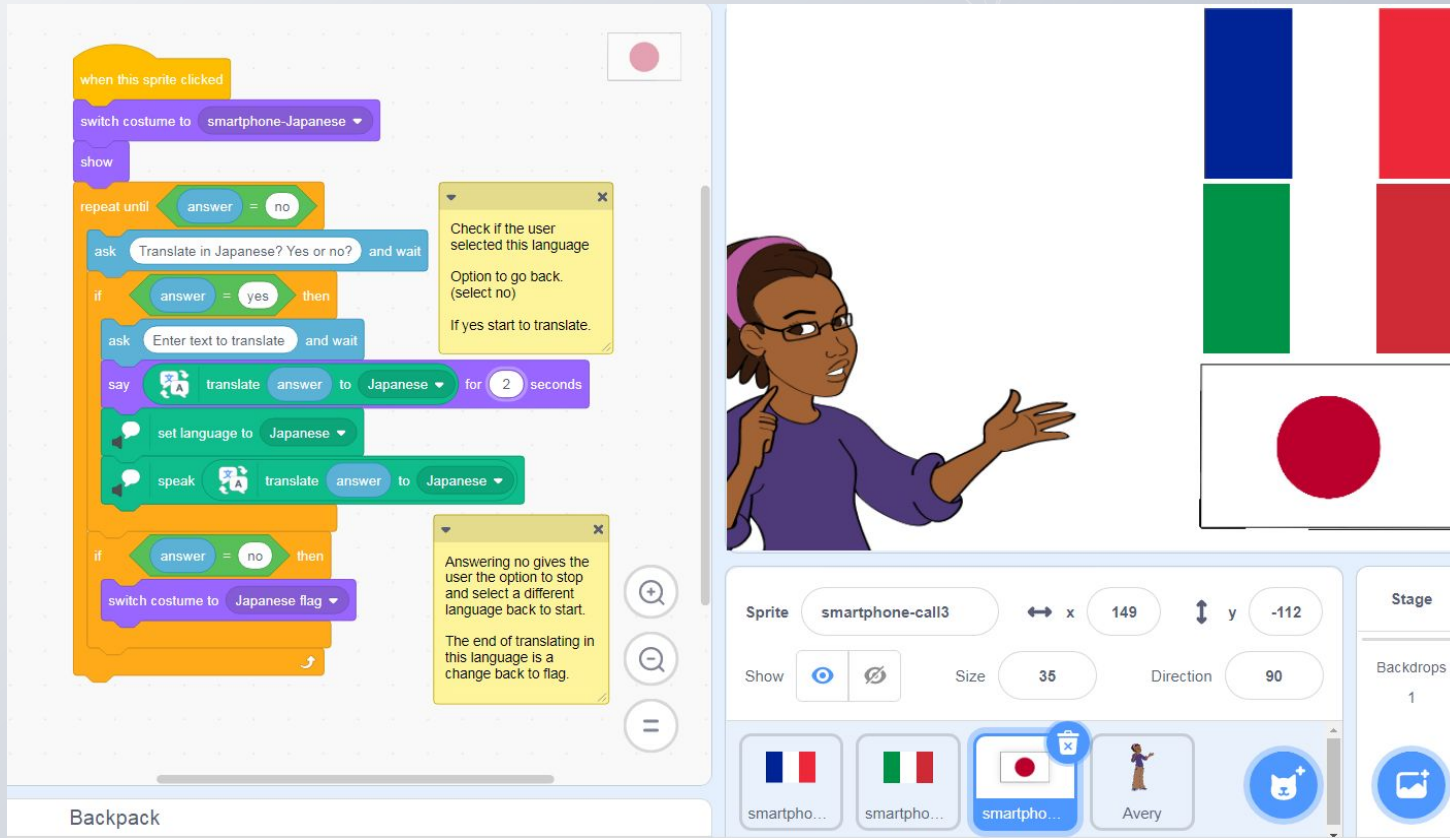
Use of repeat until block...

- Repeats the program until a condition is met

Combined with an operator block




Source: Flagpedia







An AI that recognises flags

Teachable Machine




French 


80 Image Samples

 Webcam  Upload 




Italian 


78 Image Samples

 Webcam  Upload 



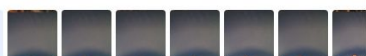
Japanese 

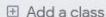
4 Image Samples

 Webcam  Upload 


Background 

33 Image Samples


 Webcam  Upload 




Training



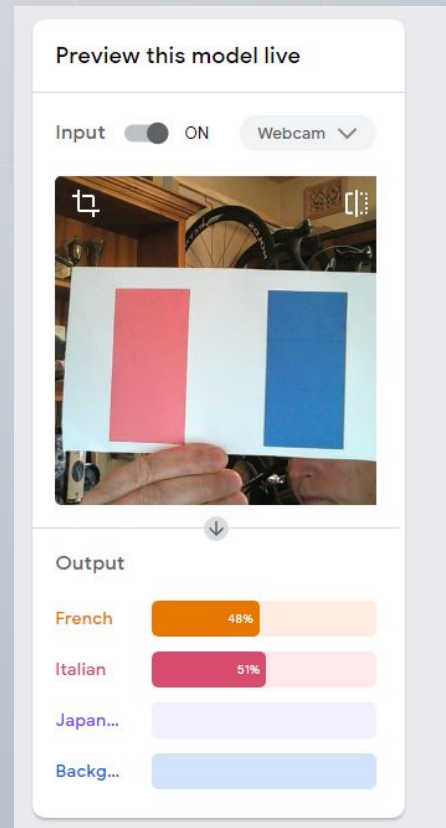
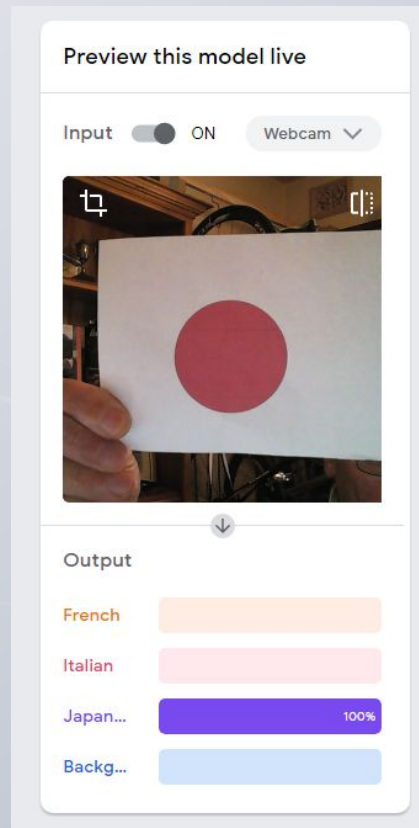
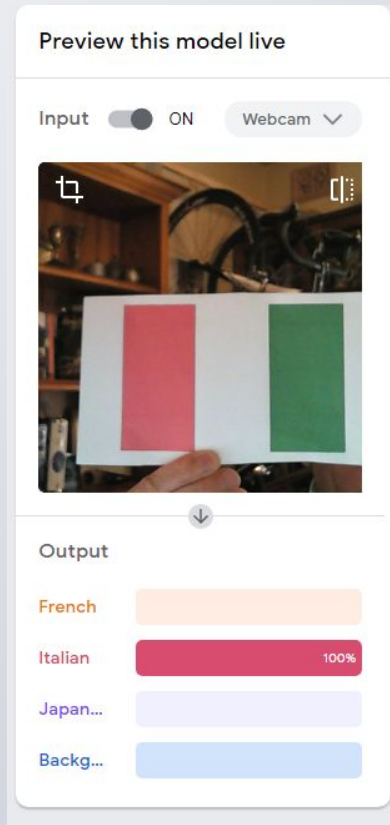
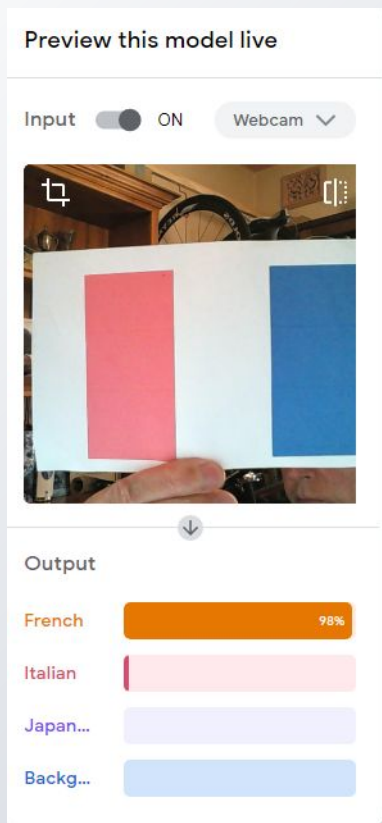
Preparing training data...

Advanced 

Preview 

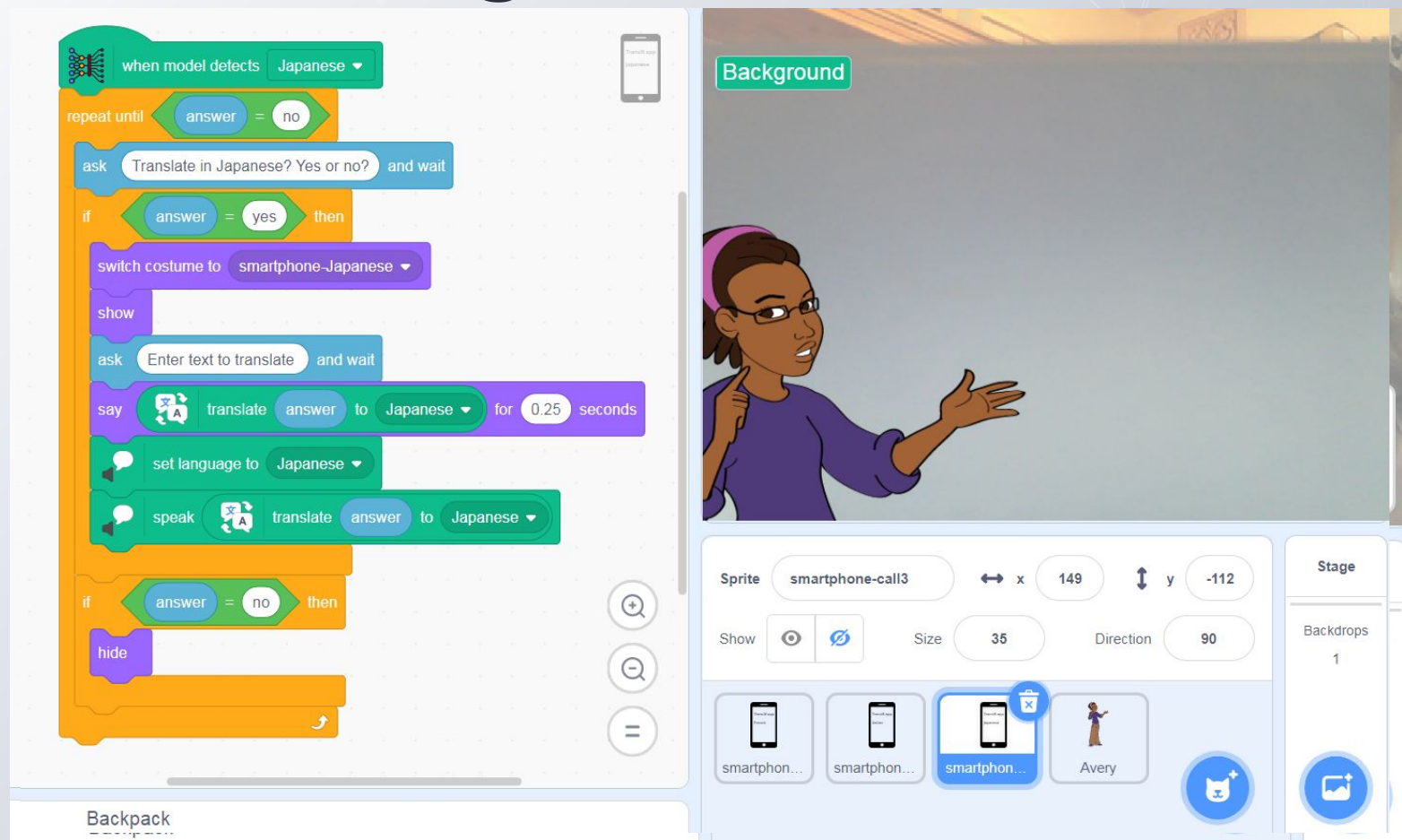
You must train a model on the left before you can preview it here.

An AI that recognises flags



Source: Flagpedia

Using the AI model



The image displays a Scratch project interface for an AI translation application. The script on the left is triggered by a 'when model detects' event set to 'Japanese'. It enters a 'repeat until' loop that continues as long as the 'answer' variable is not 'no'. Inside the loop, it asks the user 'Translate in Japanese? Yes or no?'. If the answer is 'yes', it switches the costume to 'smartphone-Japanese', shows the sprite, asks for text to translate, and uses the 'AI translate' block to convert the input to Japanese. It then sets the language to 'Japanese' and speaks the translated text. If the answer is 'no', it hides the sprite. The stage view on the right shows a character named Avery pointing at a 'Background' backdrop. The sprite panel at the bottom lists three 'smartphone-call3' costumes, with the third one selected. The 'Stage' panel shows one backdrop named 'Background'.

Script:

- when model detects Japanese
- repeat until answer = no
 - ask Translate in Japanese? Yes or no? and wait
 - if answer = yes then
 - switch costume to smartphone-Japanese
 - show
 - ask Enter text to translate and wait
 - say AI translate answer to Japanese for 0.25 seconds
 - set language to Japanese
 - speak AI translate answer to Japanese
 - if answer = no then
 - hide

Stage:

- Background

Sprite: smartphone-call3 (x: 149, y: -112, size: 35, direction: 90)

Backpack: smartphone-call3, smartphone-call3, smartphone-call3, Avery

Design our algorithm

START...

Repeat

User selects from three options (Scissors, paper, rock)

Check selection: yes or no

If answer = yes

Computer randomly selects one option

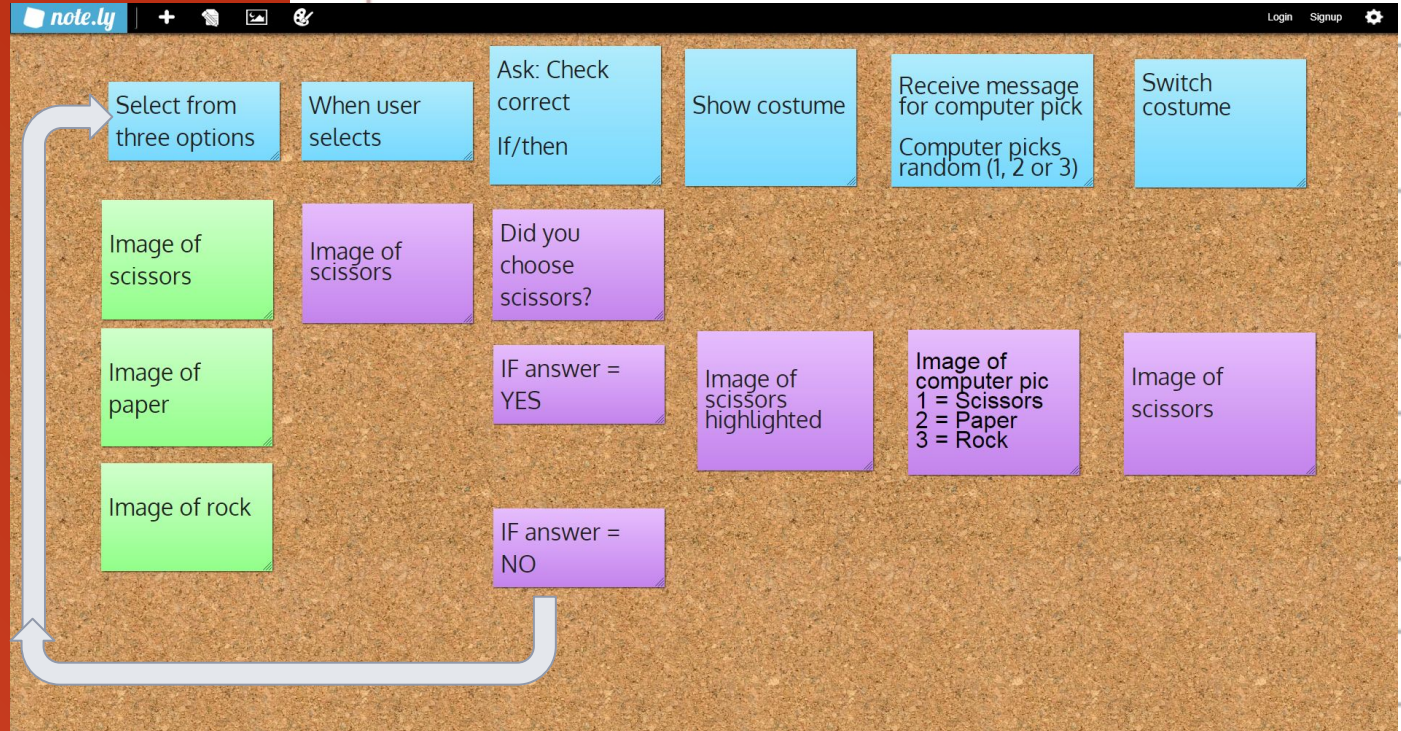
Show computer selection

Compare user choice against computer choice

END If (answer = no)

END Repeat

Design our algorithm



Setting up program for ML model

START... Ask a question to enable input

Paper, scissors, rock?

Users selects from three options

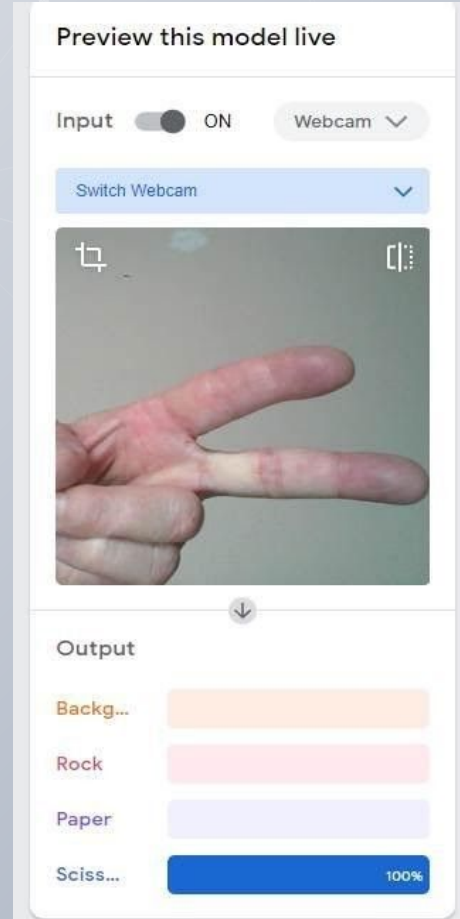
Check selection: Did you choose 'Rock'? Yes or no (if no choose again)

If yes compare to computer choice: (Create a variable
Computer pick: Random 1, 2 or 3)

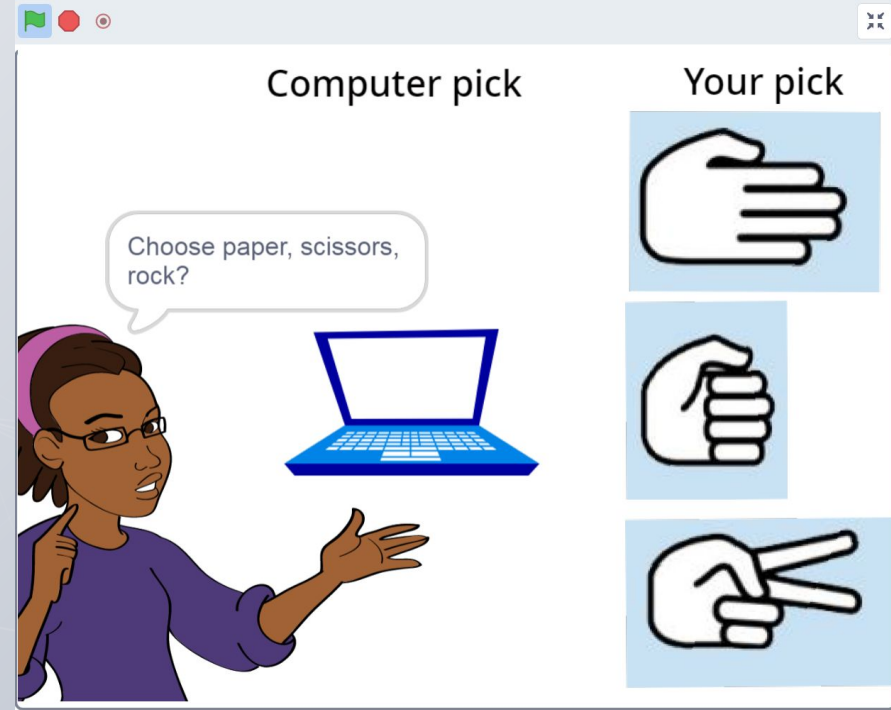
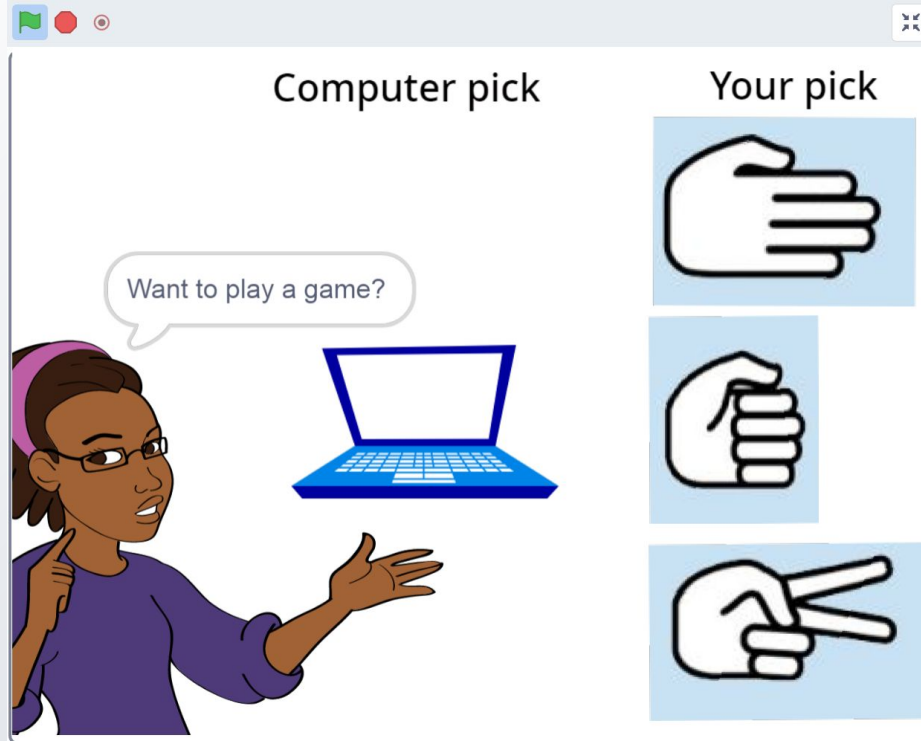
If/then (if computer pick = 1, show paper etc)

Compare computer pick to user pick

END



Design thinking: user input



Design thinking: user input select image

The image displays a Scratch project for a Rock Paper Scissors game, showing both the code and the stage view.

Code (Left Panel):

- when I receive** Computer pick
- set** Computer pick to **pick random** 1 to 3
- if** Computer pick = 1 **then**
 - switch costume to** Paper
- if** Computer pick = 2 **then**
 - switch costume to** Scissors
- if** Computer pick = 3 **then**
 - switch costume to** Rock
- wait** 5 seconds
- switch costume to** laptop
- say** Click green flag for new game for 2 seconds

Stage View (Right Panel):

- Computer pick:** A blue laptop icon.
- Your pick:** Three hand icons representing Paper, Rock, and Scissors.
- Speech bubble:** Click green flag for new game
- Sprite:** Laptop (x: -23, y: -15)
- Costumes:** Paper, Rock, Scissors, Avery, and a custom laptop costume.
- Backpack:** A blue backpack icon.

Using AI model as input

The image displays a Scratch project titled "Using AI model as input". The project is a Rock-Paper-Scissors game. The main stage features a background image of a person's hand making a rock gesture. Overlaid on this are two hand icons: a rock hand on the left and a scissors hand on the right. The text "AI predicts" is above the rock hand, "Computer pick" is above the scissors hand, and "Your pick" is to the right of the scissors hand. The "AI predicts" text is highlighted in green.

The Scratch script area on the left contains the following code:

```
when model detects Scissors
ask "Choose scissors? Yes or no?" and wait
if answer = yes then
  switch costume to Scissors2
  show
  broadcast "Computer pick"
if answer = no then
  switch costume to Scissors
wait 5 seconds
hide
```

The Scratch interface also shows the "Sprite" panel with the "Scissors" sprite selected. The "Stage" panel shows the "Backdrops" list with "1" backdrop.

Sound recognition

The ability of machines to recognise sounds including speech (part of **NLP Natural Language Processing**).

An AI that recognises sounds

94% certain
that the sound
is a dog

A challenge to
use this app in
a noisy classroom!

The screenshot displays the Teachable Machine web interface. On the left, three classes are listed: an unnamed class, 'Frog', and 'Dog'. Each class has a 'Mic' button, an 'Upload' button, and a row of eight spectrogram audio samples. The 'Dog' class samples show a distinct red vertical band, indicating a specific frequency range. In the center, a 'Training' panel shows a 'Model Trained' button and an 'Advanced' dropdown. On the right, the 'Preview' panel shows the 'Input' toggle set to 'ON', a spectrogram of the selected audio sample, an 'Overlap Factor' slider set to 0.5, and an 'Output' section. The 'Output' section shows three bars: 'Backg... Noise' (orange), 'Frog' (pink), and 'Dog' (purple) with a '94%' confidence level. At the bottom, there is an 'Add a class' button, a language selector set to 'English', and a version number 'release-2-4-1 - 2.4.1#eea8d2 - -'.

Teachable Machine

Mic Upload

Frog

8 Audio Samples / 8 minimum

Mic Upload

Dog

8 Audio Samples / 8 minimum

Mic Upload

Training

Model Trained

Advanced

Preview

Export Model

Input ON

Overlap Factor: 0.5

Output

Backg... Noise

Frog


Dog 94%













English

release-2-4-1 - 2.4.1#eea8d2 - -

E-safety: risk assessment

 Risk identified: take appropriate action to mitigate risks before using

 Proceed with caution: continue to monitor for risks

Consider	Yes	No	Suggestions to mitigate risks
Will students' personal information be publicly displayed (e.g. photograph, date of birth, gender or name of school)?			<ul style="list-style-type: none">• Obtain consent from students and their parents/carers before displaying personal information online.• Where possible, de-identify student information.
Can external, unauthorised users communicate with students?			<ul style="list-style-type: none">• Install appropriate technologies to monitor and filter activities on school ICT systems.• Teach students strategies to report external, unauthorised communication and block inappropriate content or contact.
Does the platform encourage students to use their existing email or social networking accounts for sign in or use?			<ul style="list-style-type: none">• Often platforms also have the option to sign up or log in using unique usernames and passwords. While using existing social networking accounts might be quicker, unique logins are a safer option.• Teach students the importance of strong passwords and not sharing passwords.
Are student profiles linked to apps that can display their location?			<ul style="list-style-type: none">• Teach students strategies to turn off location services functions, or to block apps that have these turned on.
Does the education department prohibit the use of this technology or platform?			<ul style="list-style-type: none">• If the education department's policies prohibit the use of this technology or platform it is recommended not to use it.
Can students access inappropriate content using this technology or platform?			<ul style="list-style-type: none">• Install appropriate technologies to monitor and filter activities on school ICT systems.• Encourage help-seeking behaviours so students know the steps to take if they come across inappropriate content.

[Basics](#)

[Getting started](#)

[Tweaking your model](#)

[Saving & Exporting](#)

[Diving Deeper](#)

[Tools and Resources](#)

Saving & Exporting

How can I save my project? 

Can I use my model outside Teachable Machine? 

Who can access my samples? 

Your samples aren't sent to any servers, unless you save your project to Google Drive — and even then, it's in *your* Google Drive, so that sample data is still yours. When you train the model, it trains in your browser tab without sending anything to any servers.

Where is my model published when I click “upload my model”? 

Who can use my model once it's published? 

Can I use Teachable Machine models with other libraries and platforms? 

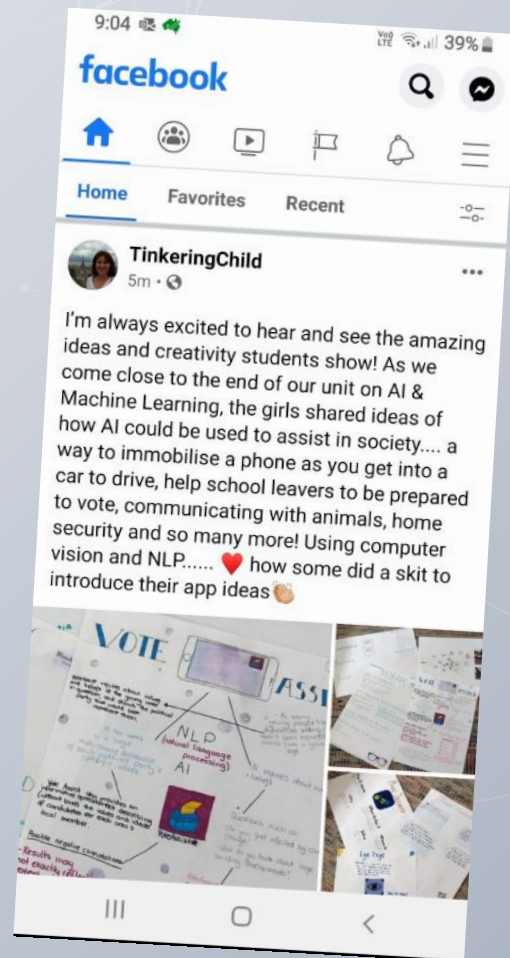
Risk
assessment

Check FAQ,
for privacy
issues:

Student project

Coming up with their own AI app idea

Our team is...	
We're creating an AI model that...	(the target user)
to help people to...	(the problem/challenge)
by providing them with...	(the possible solution)



Assessment

Artificial Intelligence is a rich field for assessment opportunities.

Here are a few examples in the core concept areas of data, algorithms and implementation

Data representation and algorithms



AI
model

Prediction

Chip packet

When model
detects, Show



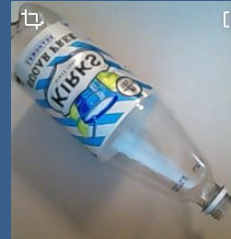
Say for 2 seconds

If that is a chip
packet its made of
non-recycled plastic.
Put in rubbish bin

Change
costume, wait,
then hide



Soft drink bottle



If that is a soft drink
bottle its made of
PET plastic.
Put in recycling bin



Banana skin

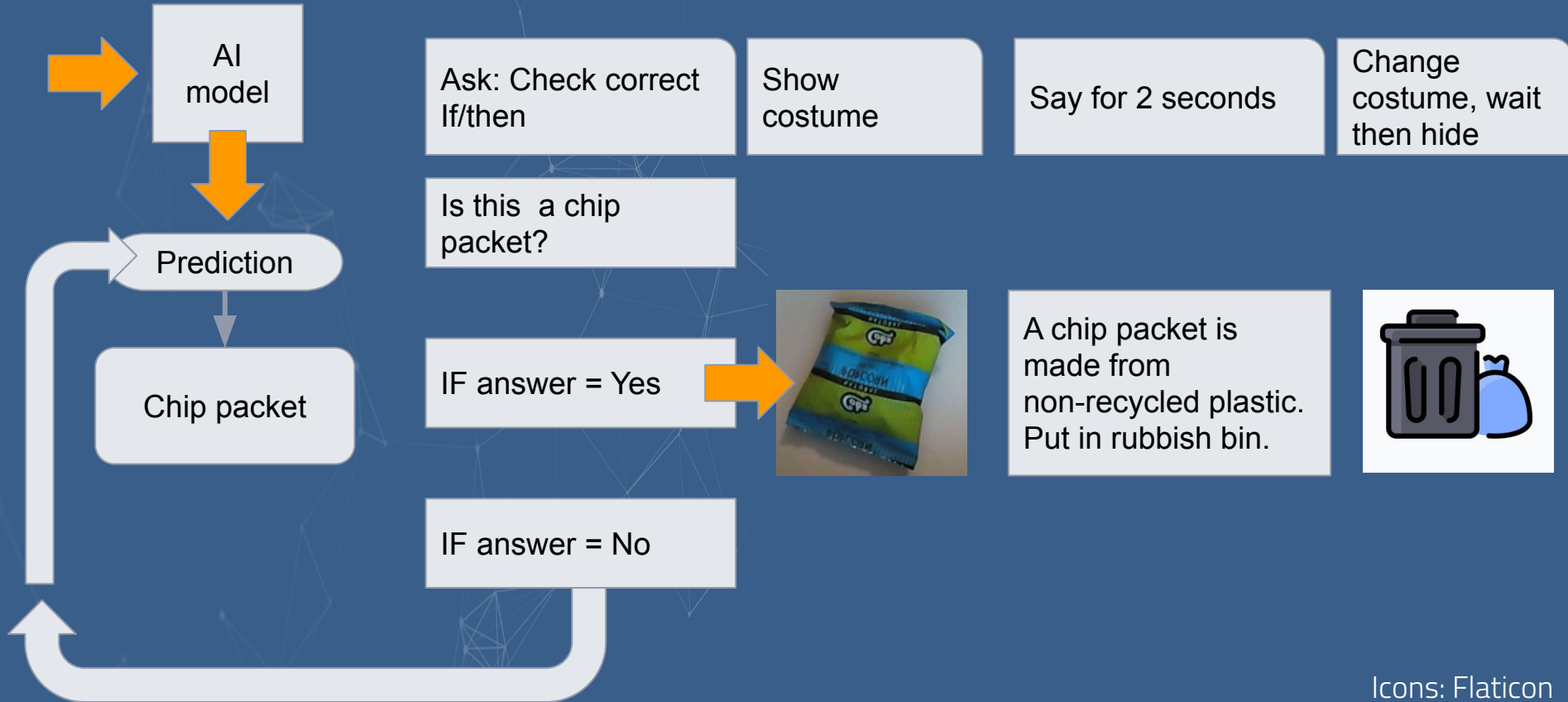


If that is a banana
skin its organic.

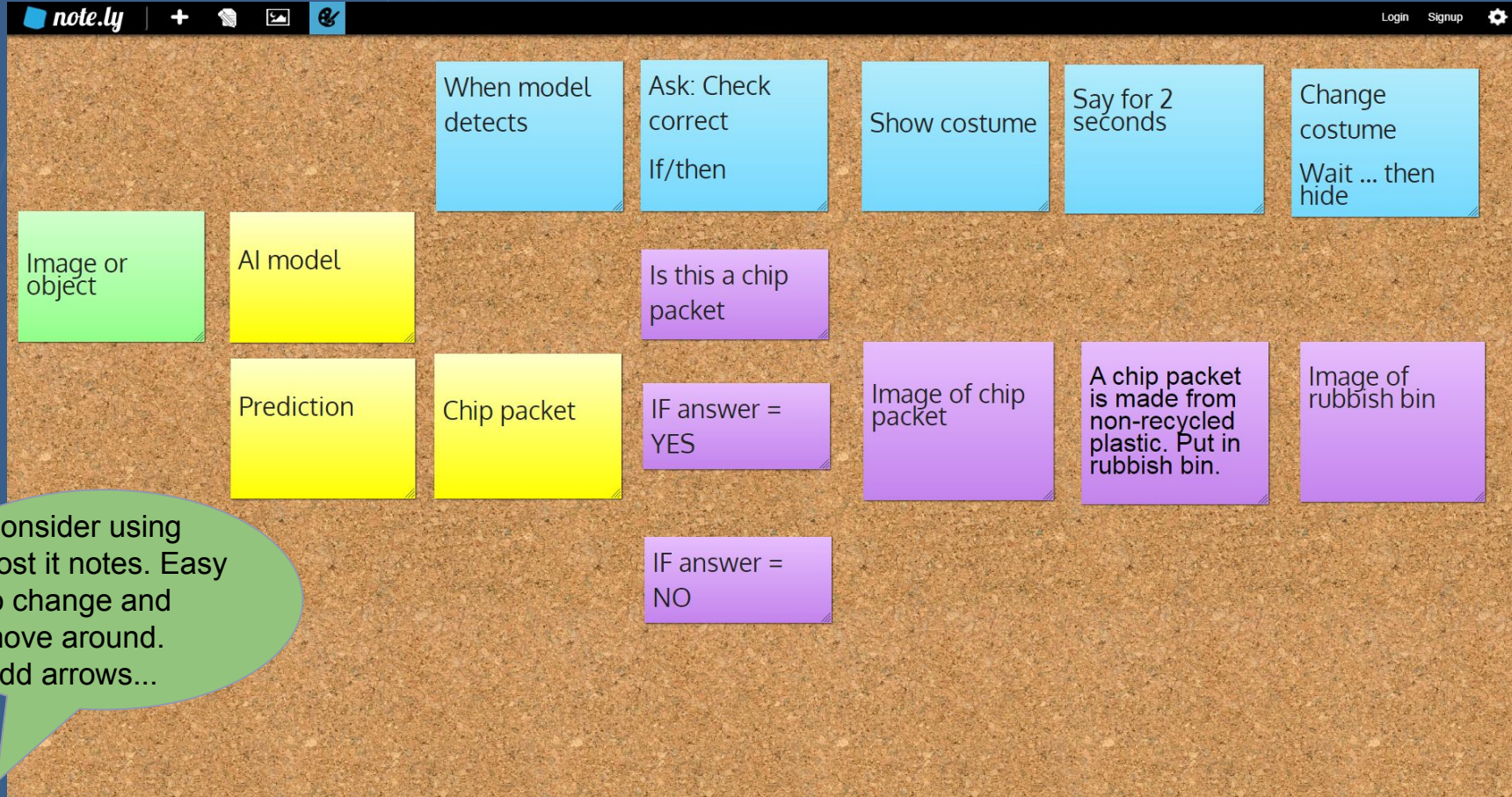
Put in green waste of
compost



Data representation and algorithms



Formative assessment



Data representation and algorithms



AI
model

Prediction

Class 1

Class 2

Class 3

Consider providing
a template.
Take away or add
columns

Data representation Algorithms, Implementation

Evidence of learning:

Trained and tested AI model

Plan of the training data

How well did they avoid bias?

How well did their model work?

(Star rating)

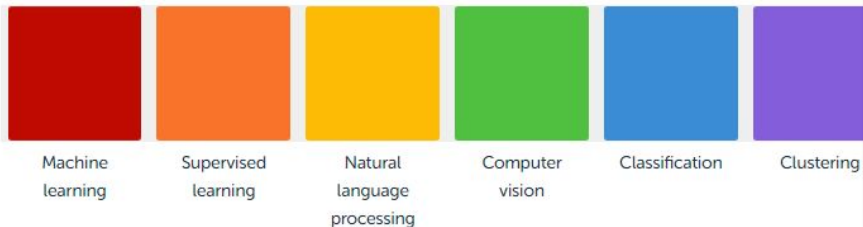
Think aloud (student records a video of the AI model in action with commentary or as an interview with the teacher.

Artificial Intelligence lesson plans

Humans display natural intelligence in contrast to machines that demonstrate artificial intelligence (AI).

AI has various definitions however for our purposes we are using the definition 'any device that perceives its environment and takes actions that maximize its chance of successfully achieving its goals' [1]. [Read more...](#)

The following lesson ideas cover a range of specialisations and subsets as indicated by the colour coding. Click on the coloured squares to learn more about each definition.



Lesson plans

Artificial Intelligence

[Access DT Hub AI lesson plans](#)



Cliffhanger ...

Join us next week, when Martin and Karsten explore some captivating ethical dilemmas of AI.

This will be the final webinar in this series.

POLL: pls let us know how we did in meeting your needs



**DIGITAL
TECHNOLOGIES
HUB**