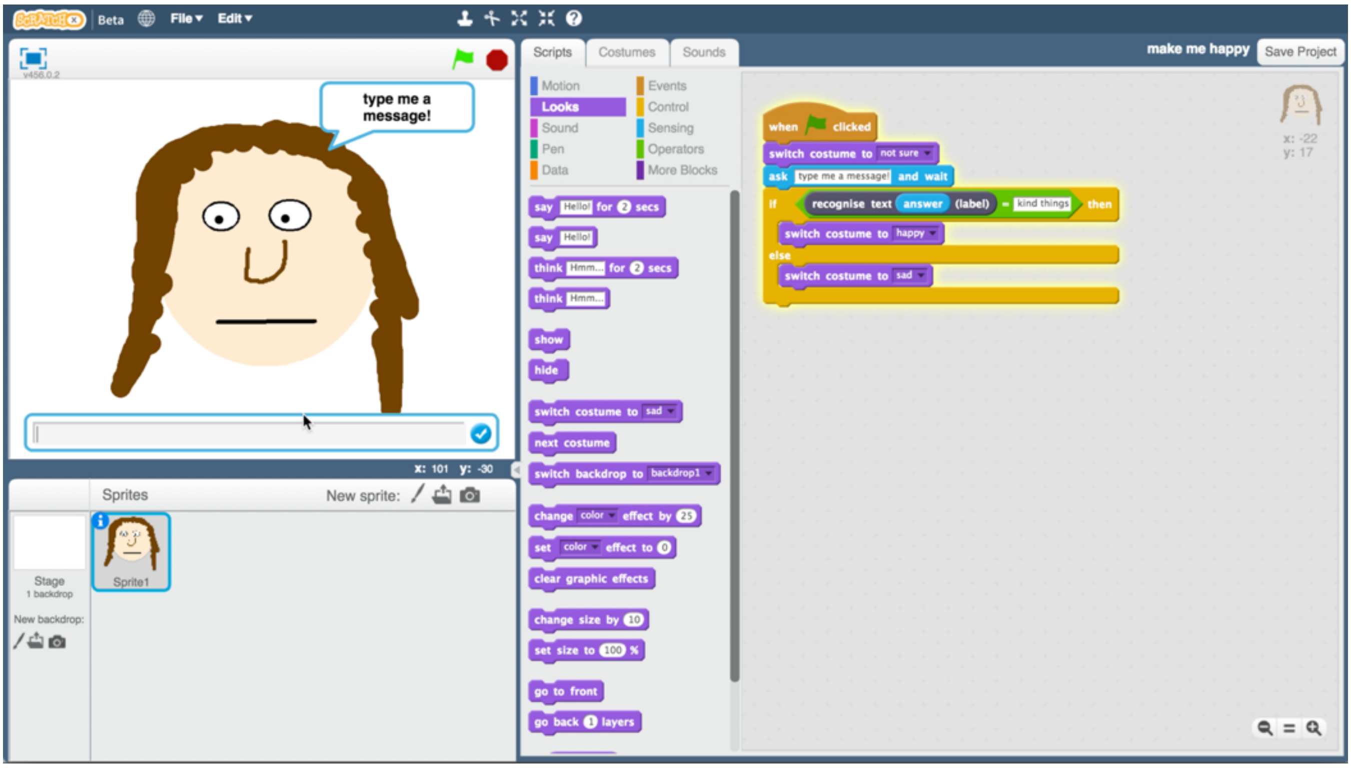
Make me happy!

In this project you will make a character that can react to what you say to it.

If you say kind things to it, it will look happy. If you say mean things to it, it will look sad.

We won't try to program a list of rules for what is kind and what is mean.

Instead, you will teach the computer to recognise kind messages and mean messages by giving it examples of each.



```
when ⚑ clicked
switch costume to not sure
ask type me a message! and wait
if recognise text answer (label) = kind things then
    switch costume to happy
else
    switch costume to sad
```

1. Go to http://machinelearningforkids.co.uk in a web browser

2. Click on “**Get started**”

3. Click on “**Log In**” and type in your username and password

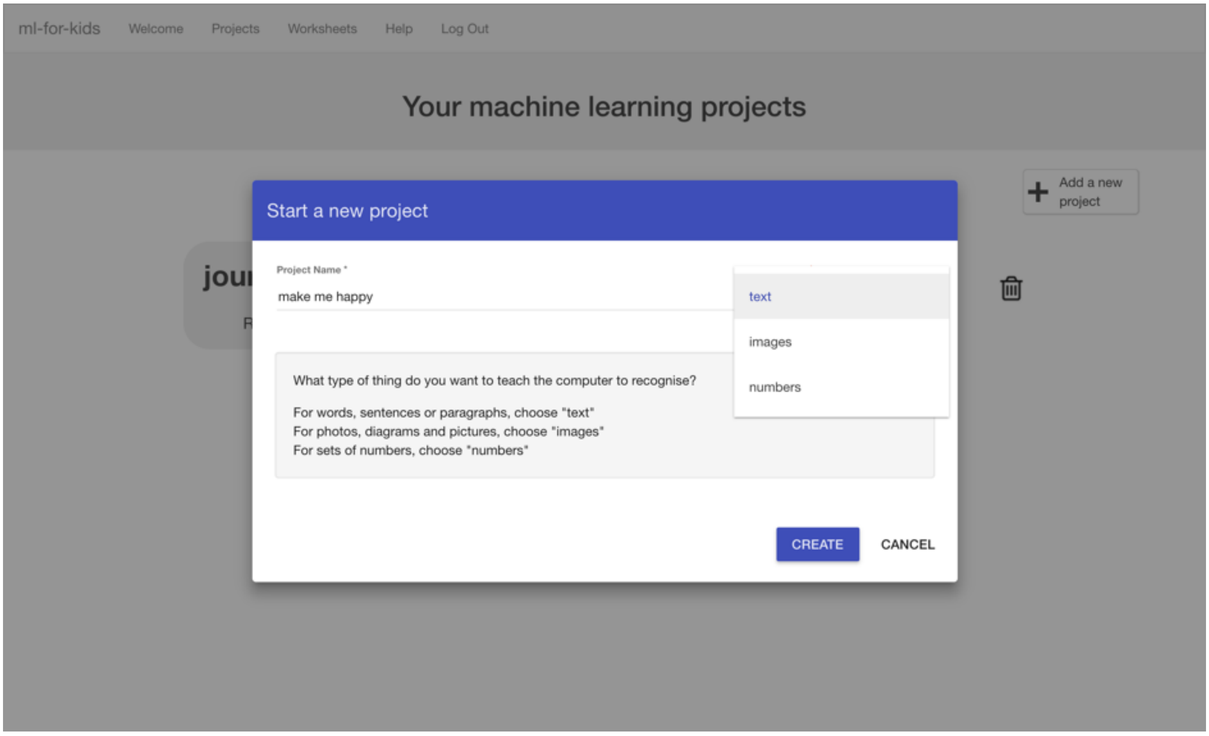
If you don’t have a username, ask your teacher or group leader to create one for you.

If you can’t remember your username or password, ask your teacher or group leader to reset it for you.

4. Click on “**Projects**” on the top menu bar

5. Click on the **“+ Add a new project**” button.

6. Name your project “make me happy” and set it to learn how to recognise “**text**”

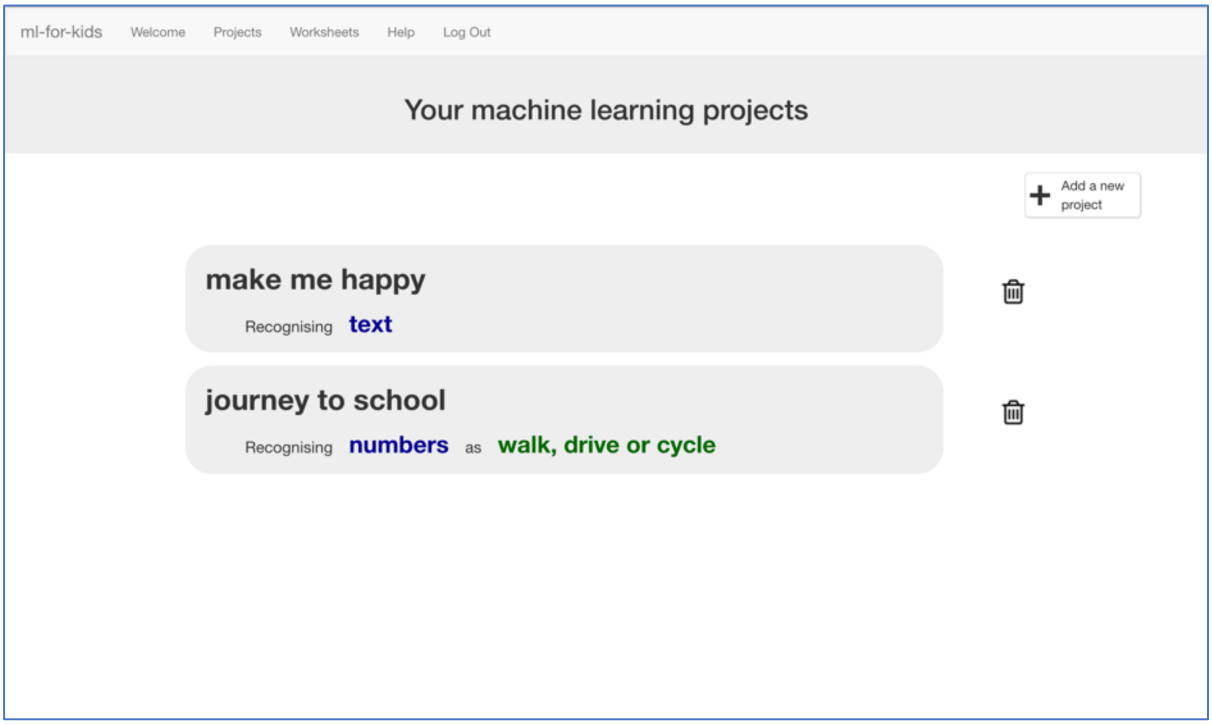


Project Name *: make me happy

What type of thing do you want to teach the computer to recognise?

For words, sentences or paragraphs, choose "text"
For photos, diagrams and pictures, choose "images"
For sets of numbers, choose "numbers"

7. You should now see "make me happy" show up in the list of your projects. Click on it.



Your machine learning projects

make me happy

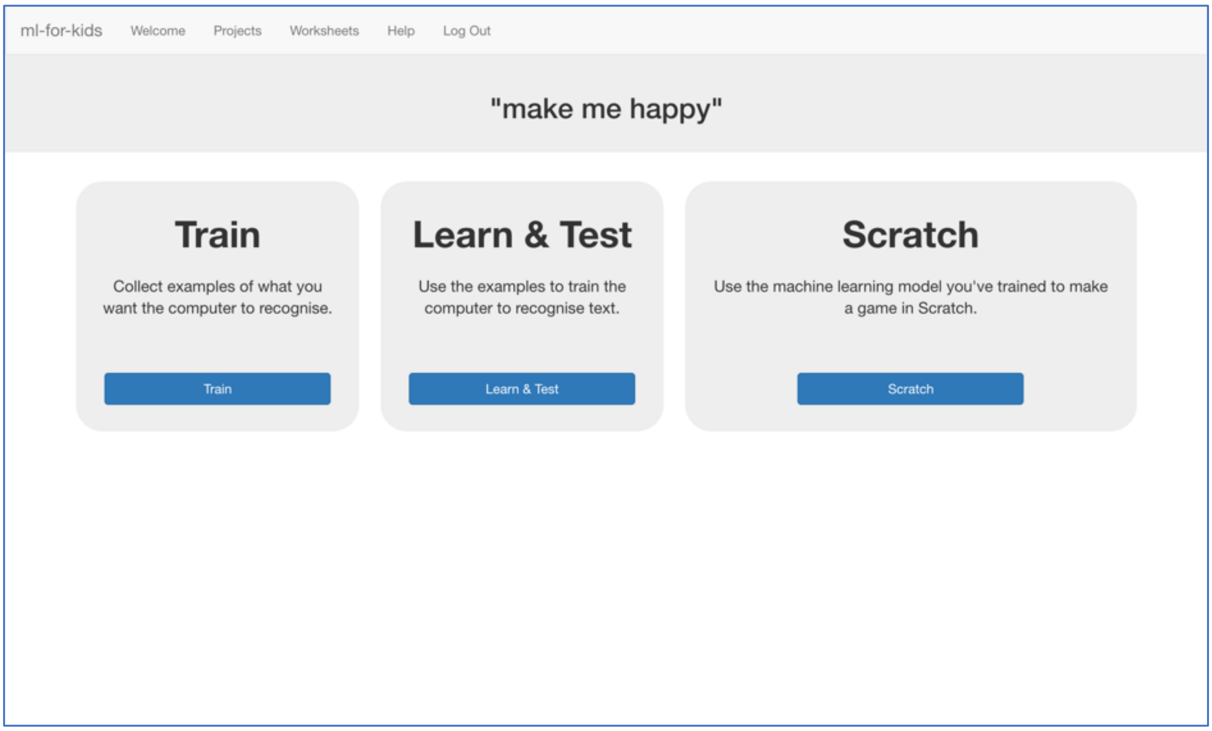
Recognising text

journey to school

Recognising numbers as walk, drive or cycle

8. We'll start by getting a project ready in Scratch. Click on the **Scratch** button.

The next page will warn you that you haven't done any machine learning yet, but clicking on ***Scratch by itself*** *will launch Scratch.*



"make me happy"

Train

Collect examples of what you want the computer to recognise.

Learn & Test

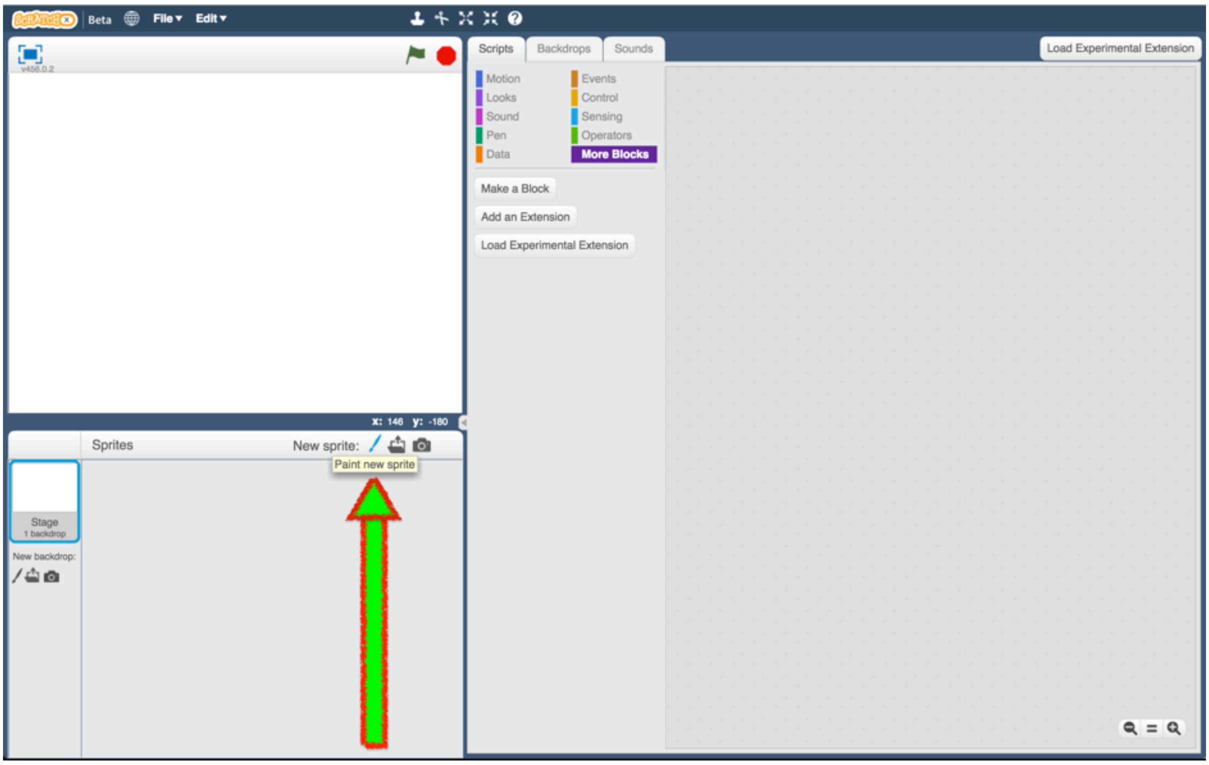
Use the examples to train the computer to recognise text.

Scratch

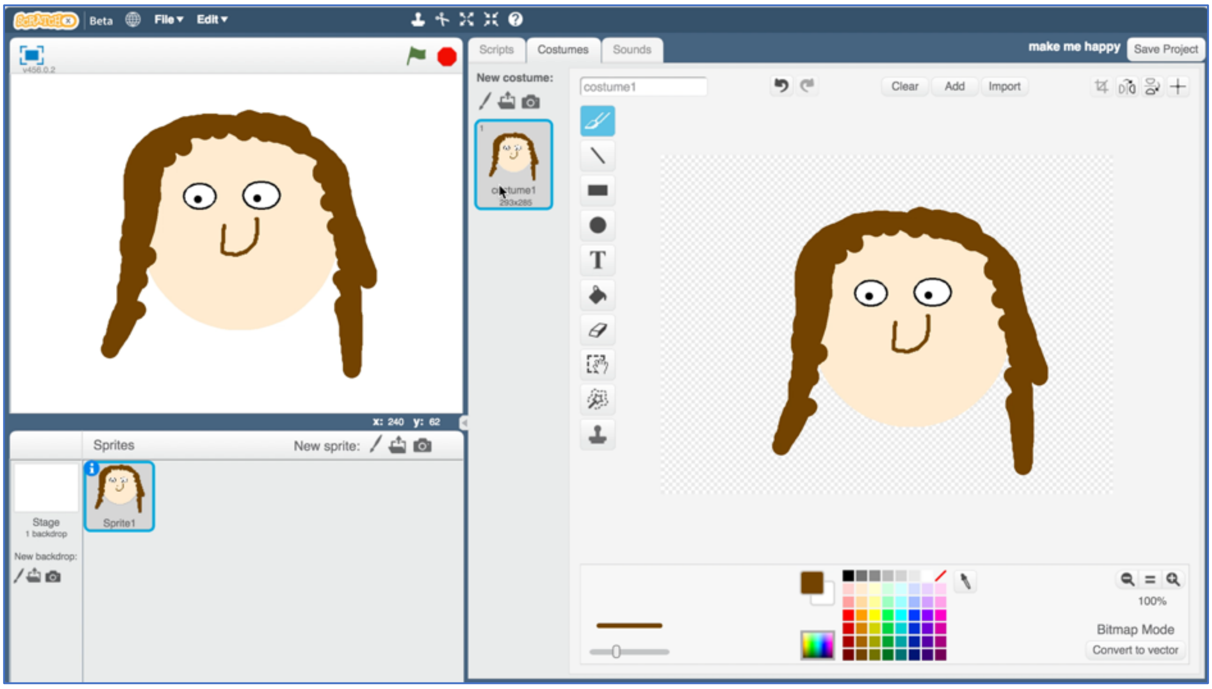
Use the machine learning model you've trained to make a game in Scratch.

9. Create a new sprite by clicking on the paintbrush icon in the Sprites window.

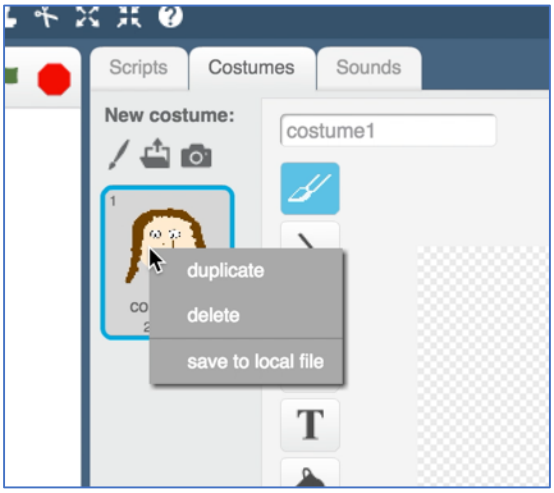
There are a few similar looking paintbrush buttons – make sure you click the one marked below.



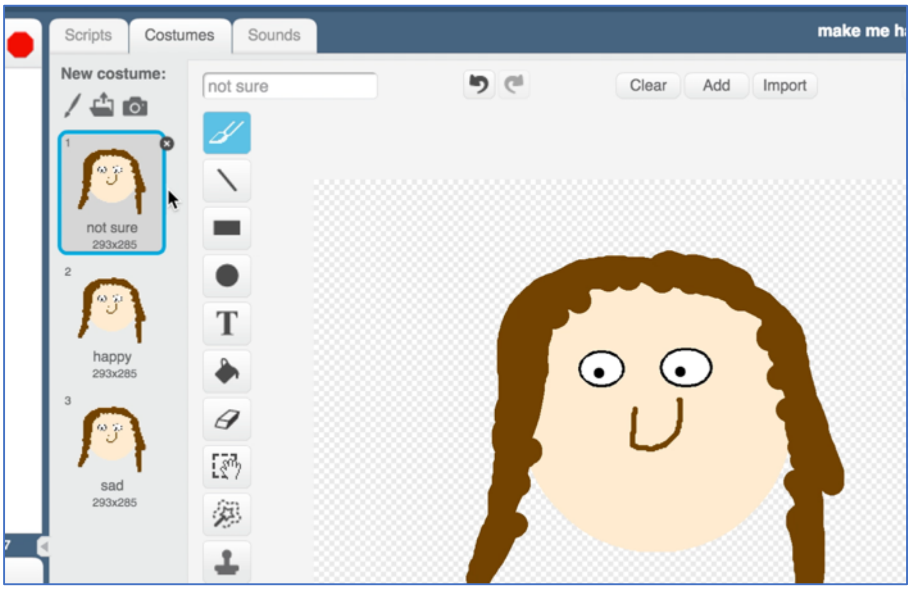
10. Draw a face, without a mouth, in the sprites editor view on the right.



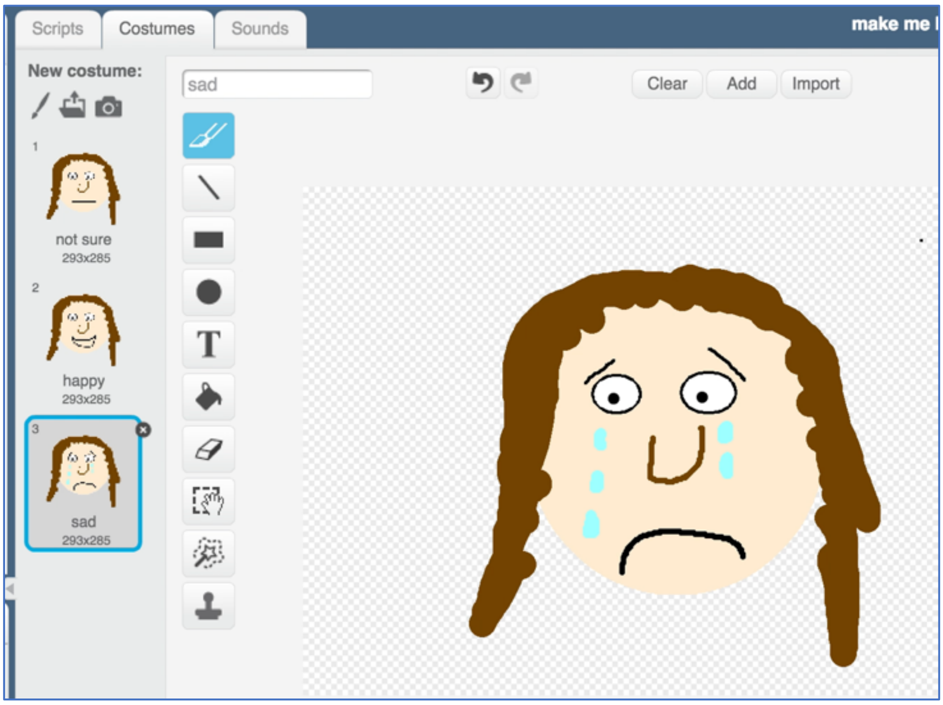
11. Right-click on the costume, and click "Duplicate". Do that again so you have three copies of the costume.



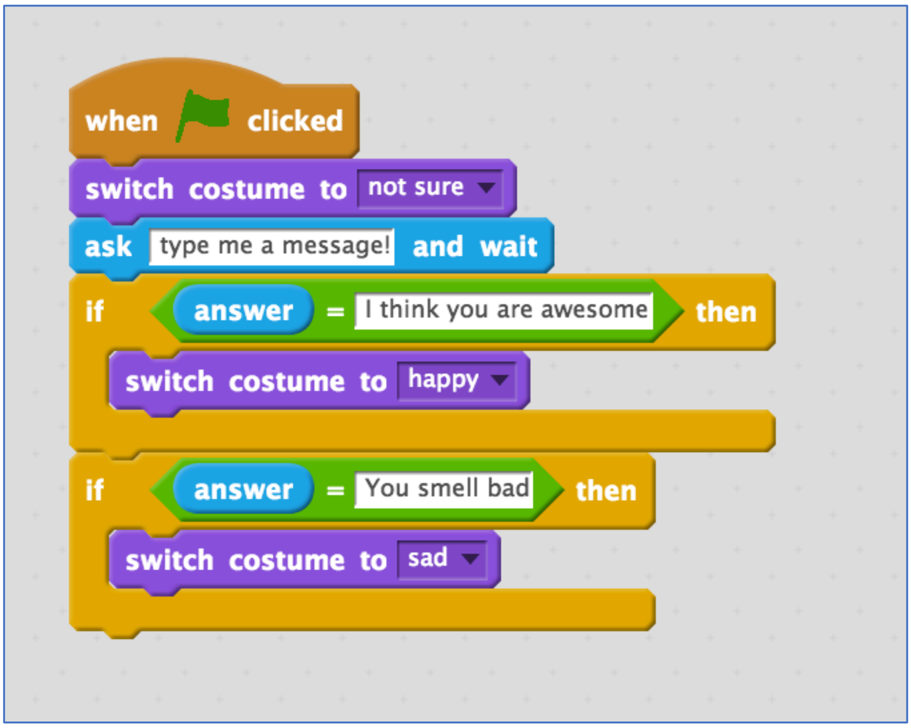
12. Name the three costumes "not sure", "happy" and "sad"



13. Draw a mouth on each of the costumes.
The "not sure" face should be a straight line.
The "happy" face should have a smile.
The "sad" face should look sad.



14. Click on the "**Scripts**" tab, and enter the following script.

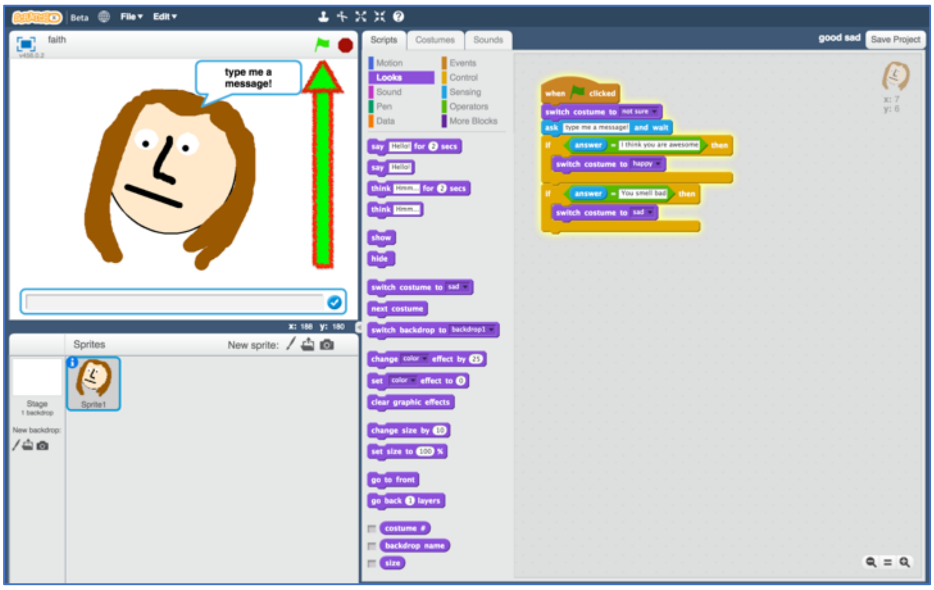


```
when [flag] clicked
switch costume to [not sure ▼]
ask [type me a message!] and wait
if <(answer) = [I think you are awesome]> then
    switch costume to [happy ▼]
if <(answer) = [You smell bad]> then
    switch costume to [sad ▼]
```

15. Save your project.

*Click on **File** -> **Save** to save the project to a file.*

16. Click on the **green flag** to test.



```
when [flag] clicked
switch costume to [not sure]
ask [type me a message!] and wait
if <(answer) = [I think you are awesome]> then
    switch costume to [happy]
if <(answer) = [You smell bad]> then
    switch costume to [sad]
```

17. Type in a message and watch it react!

Type "I think you are awesome" and press enter. The character smiles.
Click on the green flag again and type "You smell bad". The character should cry.
Type anything else, and the character's face won't change.

What have we done so far?

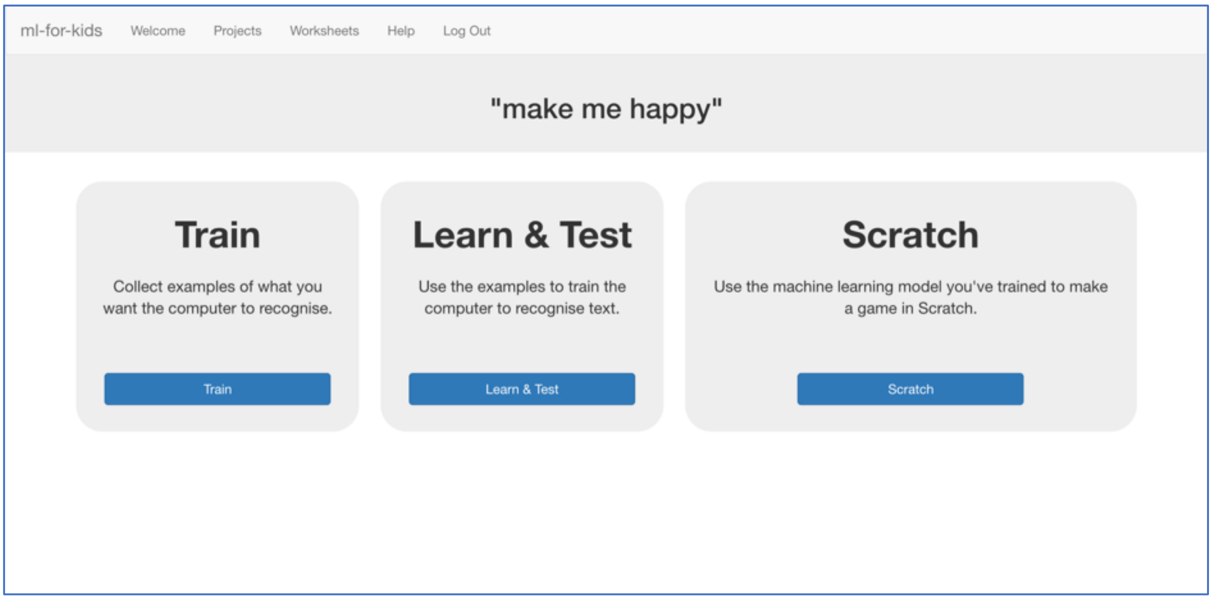
You've created a character that should react to what people type, and programmed it using a simple rules-based approach.

If you want it to react to other messages, you need to add extra **if** blocks.

The problem with this approach is that you need to predict exactly what messages the character will receive, and making a list of every possible message would take forever.

Next, we'll try a better approach – teaching the computer to recognise the messages for itself.

18. The first thing we need to do is collect some examples we can use to train the computer. Click on the **Train** button.



"make me happy"

Train

Collect examples of what you want the computer to recognise.

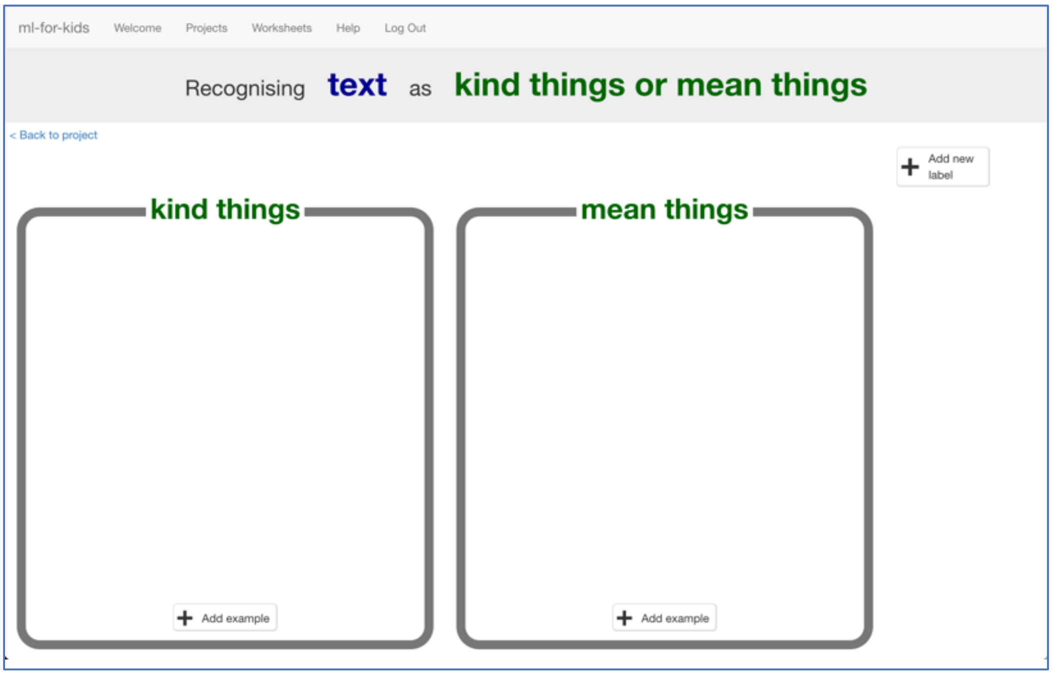
Learn & Test

Use the examples to train the computer to recognise text.

Scratch

Use the machine learning model you've trained to make a game in Scratch.

19. Click on **"+ Add new label**" and call it "kind things".
Do that again, and create a second bucket called "mean things".

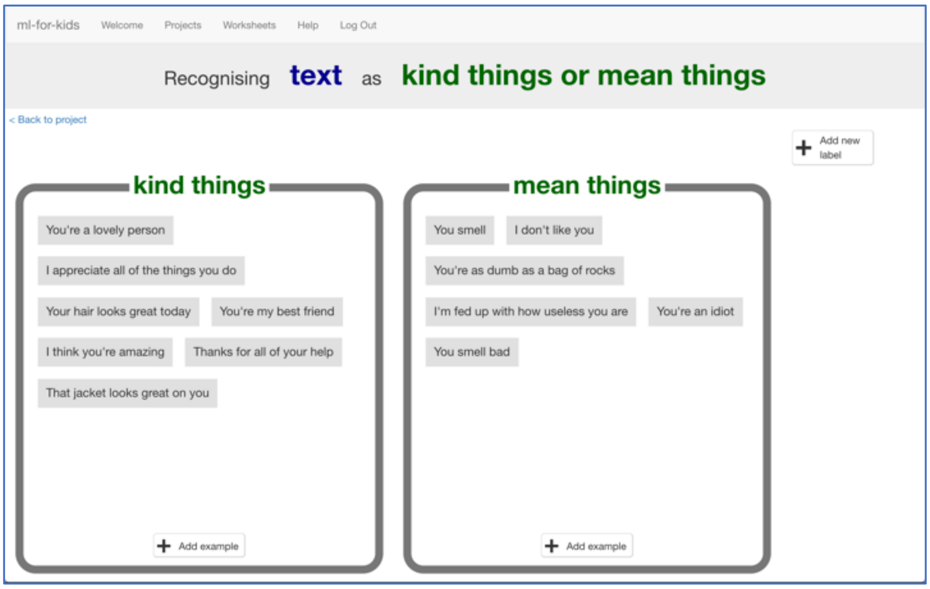


Recognising **text** as **kind things or mean things**

20. Click on the "**Add example**" button in the "kind things" bucket, and type in a kind message.

21. Click on the "**Add example**" button in the "mean things" bucket, and type in a mean message.

22. Repeat steps 20 and 21 until you’ve collected at least ten examples of each.



Recognising **text** as **kind things or mean things**

kind things

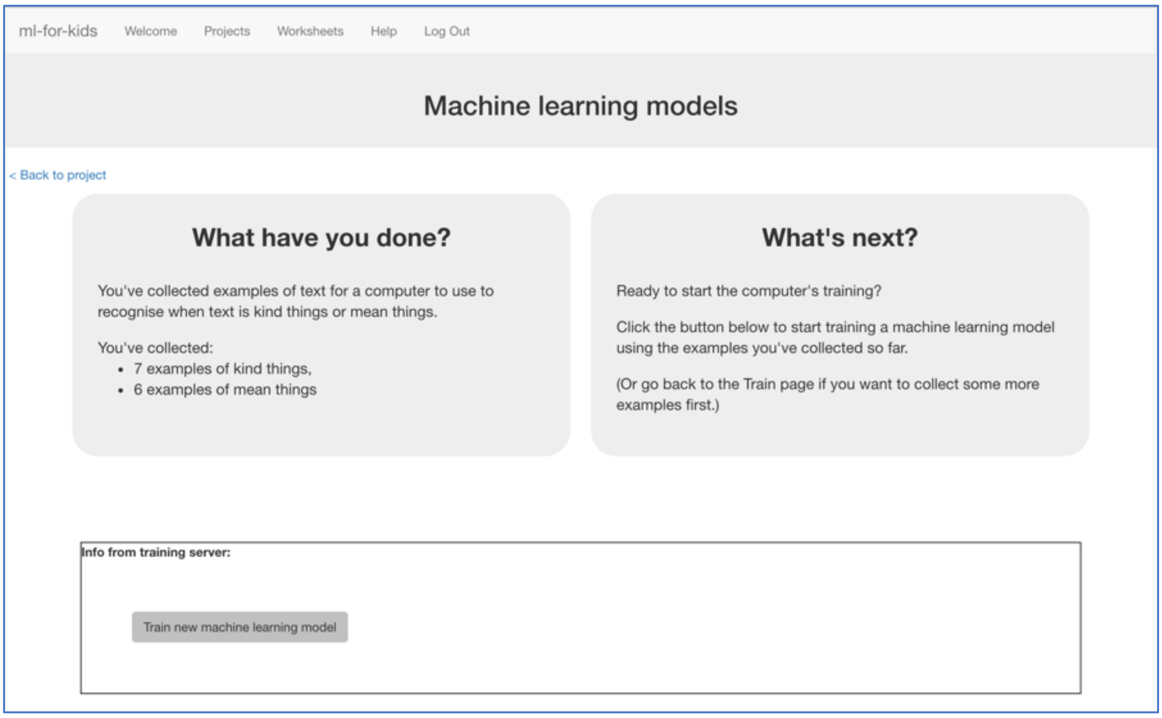
- You're a lovely person
- I appreciate all of the things you do
- Your hair looks great today
- You're my best friend
- I think you're amazing
- Thanks for all of your help
- That jacket looks great on you

mean things

- You smell
- I don't like you
- You're as dumb as a bag of rocks
- I'm fed up with how useless you are
- You're an idiot
- You smell bad

23. Click on the **“< Back to project**” link, then click on the “**Learn & Test**” button.

24. Click on the “**Train new machine learning model**” button. *As long as you’ve collected enough examples, the computer should start to learn how to recognise messages from the examples you’ve given to it.*



Machine learning models

What have you done?

You've collected examples of text for a computer to use to recognise when text is kind things or mean things.

You've collected:

- 7 examples of kind things,
- 6 examples of mean things

What's next?

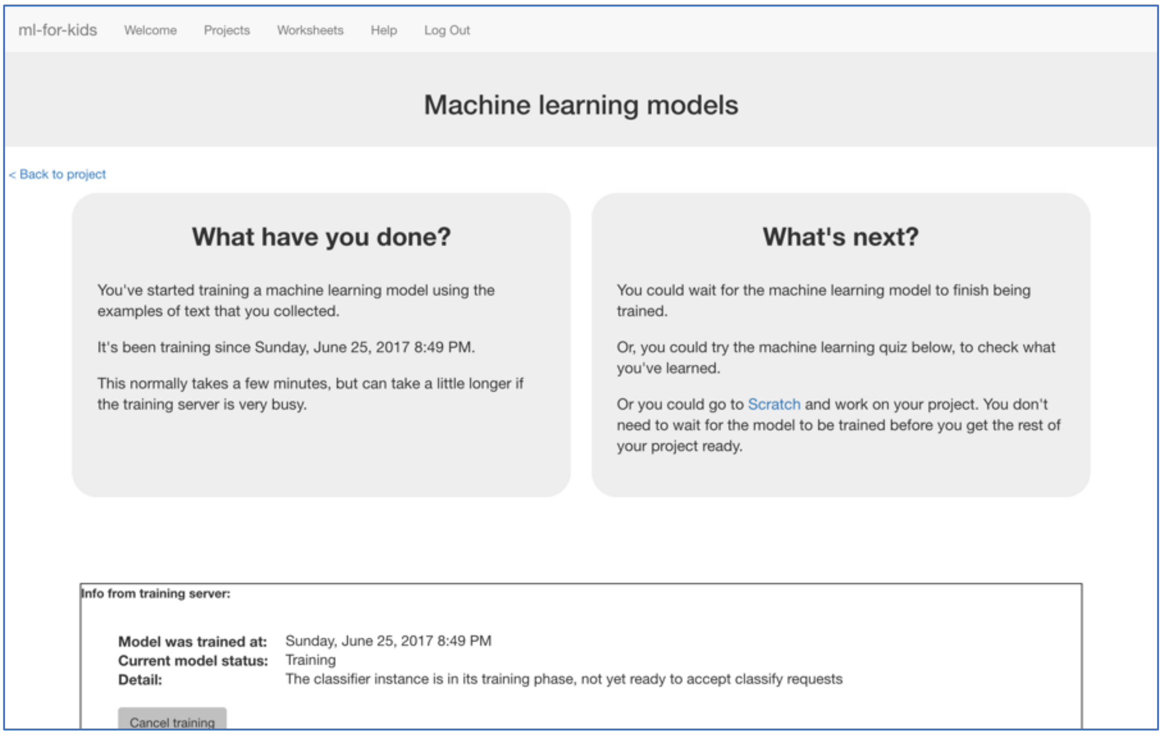
Ready to start the computer's training?

Click the button below to start training a machine learning model using the examples you've collected so far.

(Or go back to the Train page if you want to collect some more examples first.)

Info from training server:

25. Wait for the training to complete. This might take a few minutes.
While waiting, try to complete the machine-learning multi-choice quiz at the bottom of the page.



Machine learning models

< Back to project

What have you done?

You've started training a machine learning model using the examples of text that you collected.

It's been training since Sunday, June 25, 2017 8:49 PM.

This normally takes a few minutes, but can take a little longer if the training server is very busy.

What's next?

You could wait for the machine learning model to finish being trained.

Or, you could try the machine learning quiz below, to check what you've learned.

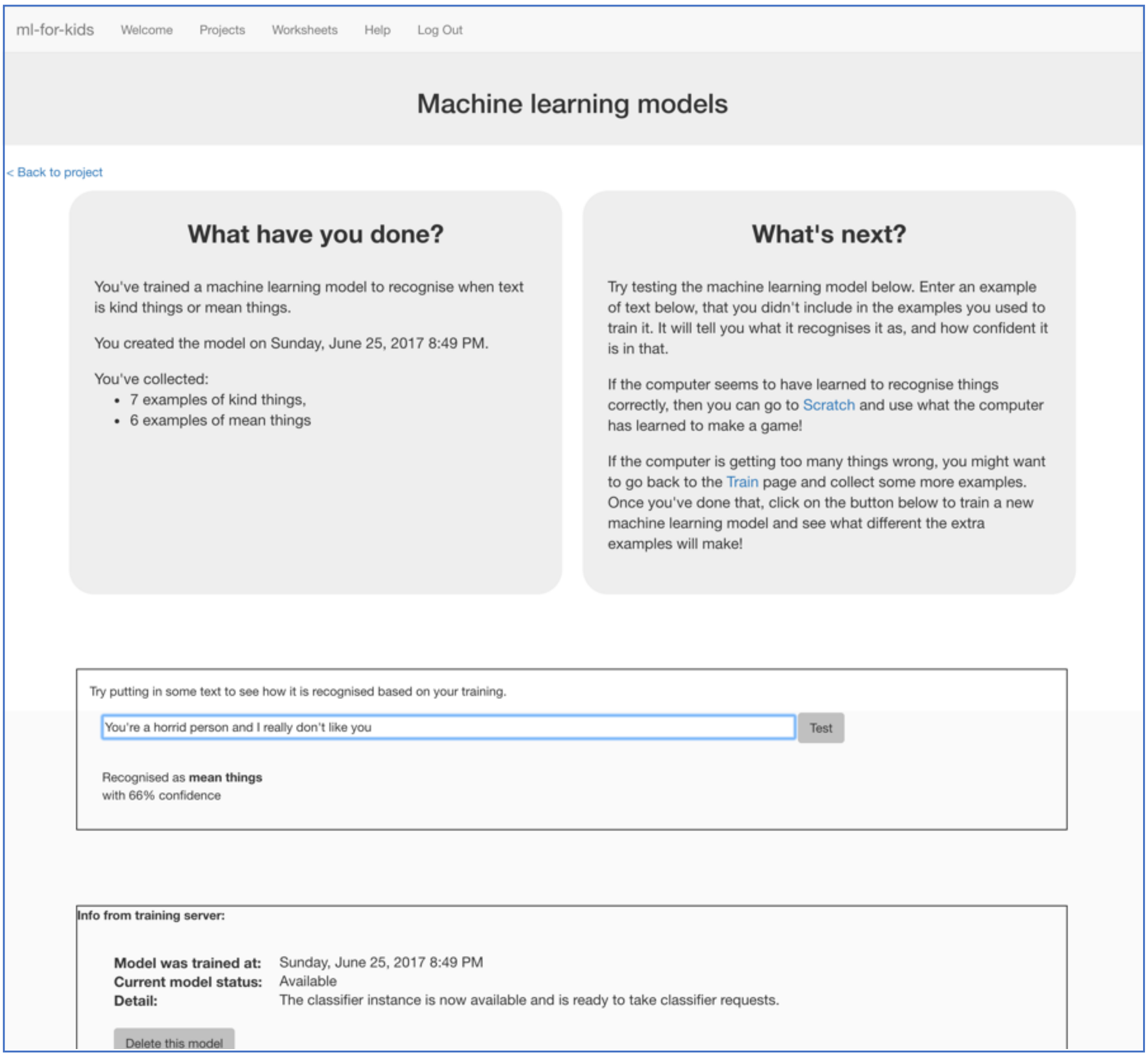
Or you could go to Scratch and work on your project. You don't need to wait for the model to be trained before you get the rest of your project ready.

Info from training server:

Model was trained at: Sunday, June 25, 2017 8:49 PM
Current model status: Training
Detail: The classifier instance is in its training phase, not yet ready to accept classify requests

26. Once the training has completed, a Test box will be displayed.
Try testing your machine learning model to see what the computer has learned.
Type something kind, and press enter. It should be recognised as kind.
Type something mean, and press enter. It should be recognised as mean.

Test it with examples that you haven't shown the computer before.
If you're not happy with how the computer recognises the messages, go back to step 20, and add some more examples.
Make sure you repeat step 24 to train with the new examples though!



Machine learning models

< Back to project

What have you done?

You've trained a machine learning model to recognise when text is kind things or mean things.

You created the model on Sunday, June 25, 2017 8:49 PM.

You've collected:
- 7 examples of kind things,
- 6 examples of mean things

What's next?

Try testing the machine learning model below. Enter an example of text below, that you didn't include in the examples you used to train it. It will tell you what it recognises it as, and how confident it is in that.

If the computer seems to have learned to recognise things correctly, then you can go to Scratch and use what the computer has learned to make a game!

If the computer is getting too many things wrong, you might want to go back to the Train page and collect some more examples. Once you've done that, click on the button below to train a new machine learning model and see what different the extra examples will make!

Try putting in some text to see how it is recognised based on your training.

You're a horrid person and I really don't like you

Recognised as **mean things**
with 66% confidence

Info from training server:

Model was trained at: Sunday, June 25, 2017 8:49 PM
Current model status: Available
Detail: The classifier instance is now available and is ready to take classifier requests.

What have we done so far?

You've started to train a computer to recognise text as being kind or mean. Instead of trying to write rules to be able to do this, you are doing it by collecting examples. These examples are being used to train a machine learning "model".

This is called "supervised learning" because of the way you are supervising the computer's training.

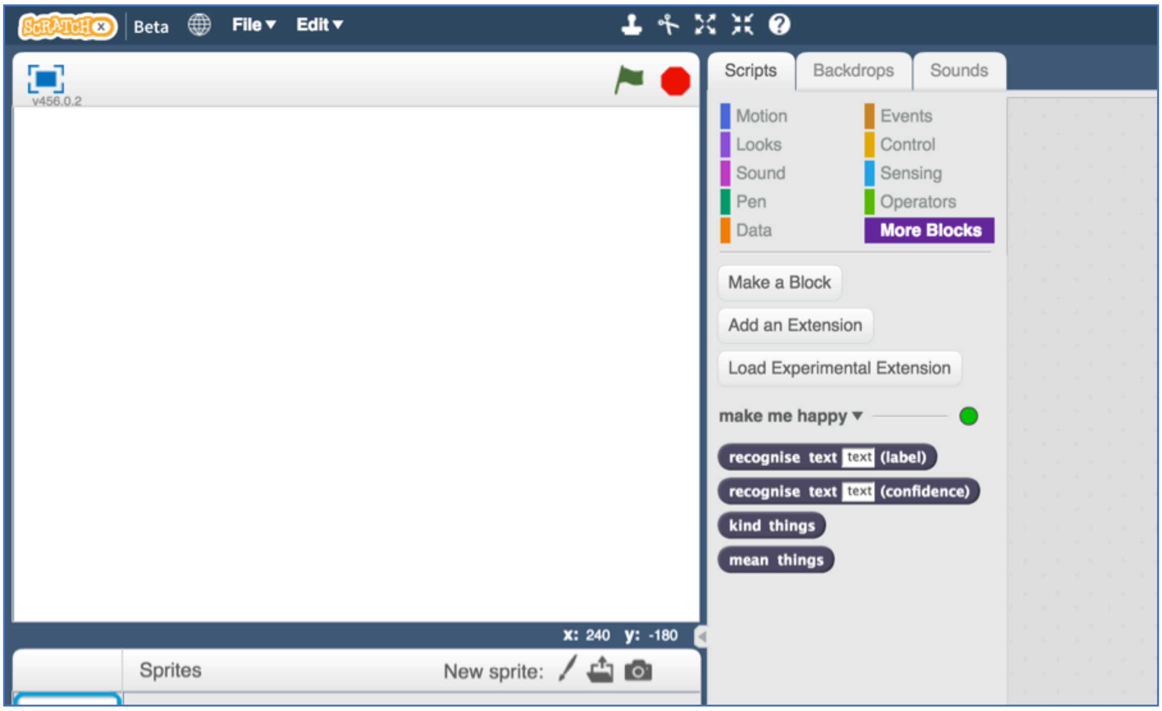
The computer will learn from patterns in the examples you've given it, such as the choice of words, and the way sentences are structured. These will be used to be able to recognise new messages.

27. Click on the **"< Back to project**" link, then back to the "**Scratch**" button.

This page will be updated with instructions on how to use the new blocks in Scratch from your project. Keep the page open if you need to check back on how to use them.

28. Click on the "**Open in Scratch**" button at the bottom to launch the Scratch editor.

You should see four new blocks in the "More blocks" section from your "make me happy" project.



29. Load the Scratch project you saved before.

Click on ***File*** *->* ***Load***

Tips

You don't have to wait for the training to finish to work on your Scratch project

If you use Scratch before your model has finished training, it will just make random guesses for kind/mean.

That means it will get it wrong a lot, but that shouldn't stop you working on your character or it's script blocks.

More examples!

The more examples you give it, the better the computer should get at recognising whether a message is kind or mean.

Try and be even

Try and come up with roughly the same number of examples for kind and mean.

If you have a lot of examples for one type, and not the other, the computer might learn that type is more likely, so you'll affect the way that it learns to recognise messages.

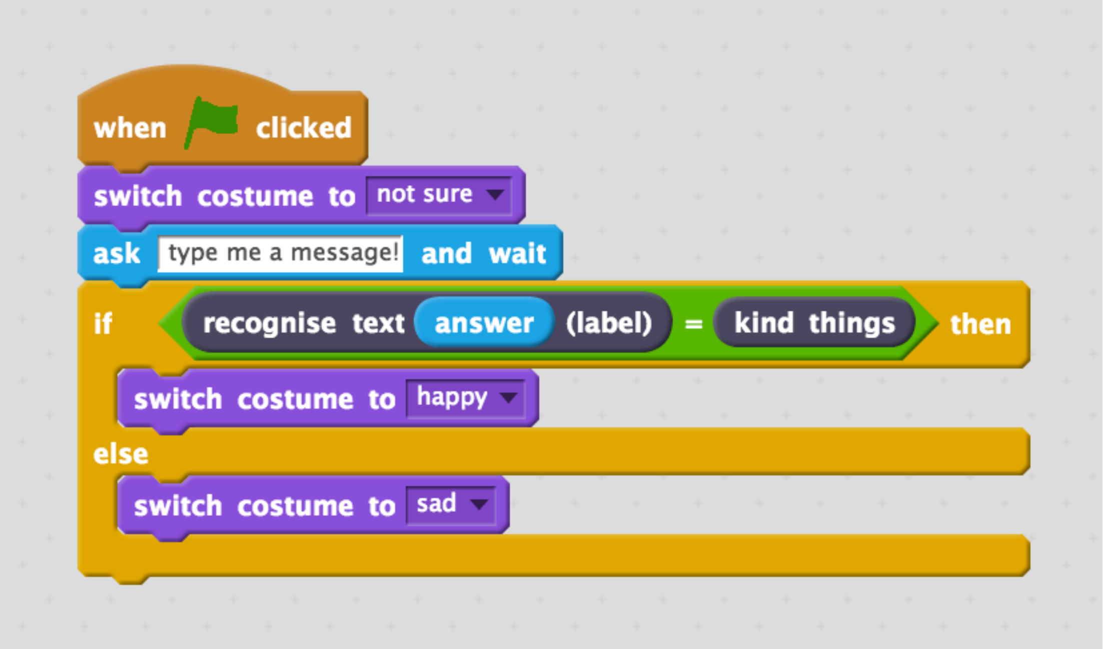
Mix things up with your examples

Try to come up with lots of different types of examples.
For example, make sure that you include some long examples and some very short ones.

30. Click on the "**Scripts**" tab, and update the script to use your machine learning model instead of the rules you made before.

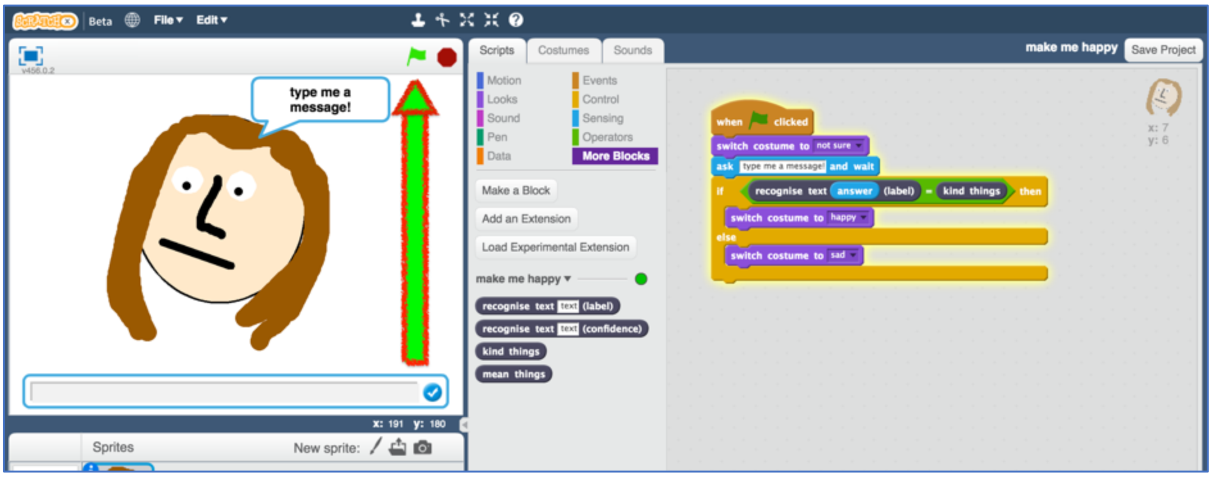
The "recognise text … (label)" block is a new block added by your project. If you give it a text message, it will return either "kind things" or "mean things" based on the training you've given to the computer.

You can use this to choose the costume to switch to.



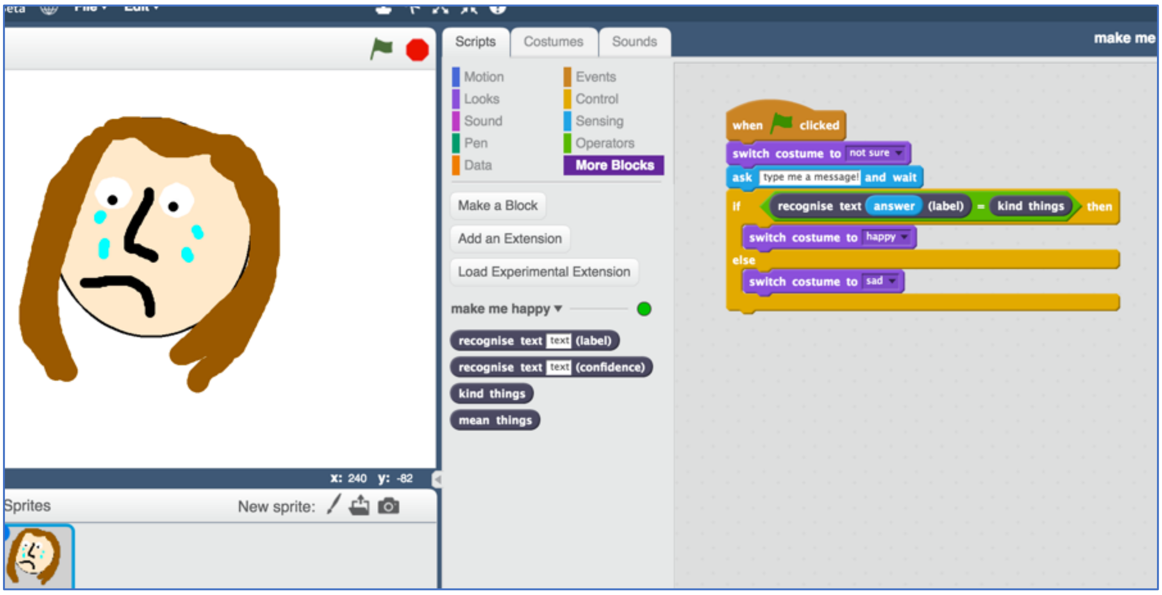
```
when ⚑ clicked
switch costume to not sure
ask type me a message! and wait
if recognise text answer (label) = kind things then
    switch costume to happy
else
    switch costume to sad
```

31. Click on the **green flag** to test again.



32. Test your project

Type a kind message and press enter. The character should smile.
Click the green flag again. Type a mean and unkind message and press enter. The character should look sad.
This should work for messages that you didn't include in your training.



```
when [flag] clicked
switch costume to not sure
ask type me a message! and wait
if recognise text answer (label) = kind things then
    switch costume to happy
else
    switch costume to sad
```

33. Save your project.

What have we done so far?

You've modified your Scratch character to use machine learning instead of your earlier rules-based approach.

Training the computer to be able to recognise messages for itself should be much quicker than trying to make a list of every possible message.

The more examples you give it, the better it should get at recognising messages correctly.

Ideas and Extensions

Now that you've finished, why not give one of these ideas a try?

Or come up with one of your own?

Write a reply

Instead of just changing the way they look, make your character reply, based on what it recognises in the message!

Try a different character

Instead of a person's face, why not try something different, like an animal?

It could react in different ways, instead of smiling.

For example, you could make a dog that wags their tail if you say something kind to it!

Recognising random messages

Try adding a third bucket to recognise messages that aren't particularly kind or mean – like "What is the time?".

Different emotions

Instead of kind and mean, could you train the character to recognise other types of message?