



Alien Language

In this project you will train the computer to understand an alien language.

You'll use that to control an alien character so that it can understand what you tell it to do.

The image shows a Scratch project titled "alien-language". The stage features a green alien with one eye and a smiling mouth, standing on a purple planet with a city skyline in the background. The backdrop is a space scene with stars and a planet. In the script editor, there are several scripts for the "alien" sprite:

- A "when green flag clicked" script:
 - Trains a machine learning model: `train new machine learning model`.
 - Waits until the model is ready: `wait until [is the machine learning model ready to use?]`.
 - Starts listening: `start listening`.
- A "when I hear left" sound event script:
 - Trains a machine learning model: `train new machine learning model`.
 - Is the machine learning model ready? `[is the machine learning model ready?]`
 - Walk left: `walk left`.
- A "when I hear right" sound event script:
 - Trains a machine learning model: `train new machine learning model`.
 - Is the machine learning model ready? `[is the machine learning model ready?]`
 - Walk right: `walk right`.
- Two movement scripts triggered by `when green flag clicked`:
 - Move left: `go to x: [x position - 5] y: [y position]`, `turn [5] degrees`, `wait [0.1] seconds`.
 - Move right: `go to x: [x position + 5] y: [y position]`, `turn [5] degrees`, `wait [0.1] seconds`.

The script editor also shows the "Code" tab selected, along with tabs for "Costumes" and "Sounds". The left sidebar lists categories like Motion, Looks, Sound, Events, Control, Sensing, Operators, Variables, My Blocks, Images, and Alien Language, with the Alien Language category currently expanded.



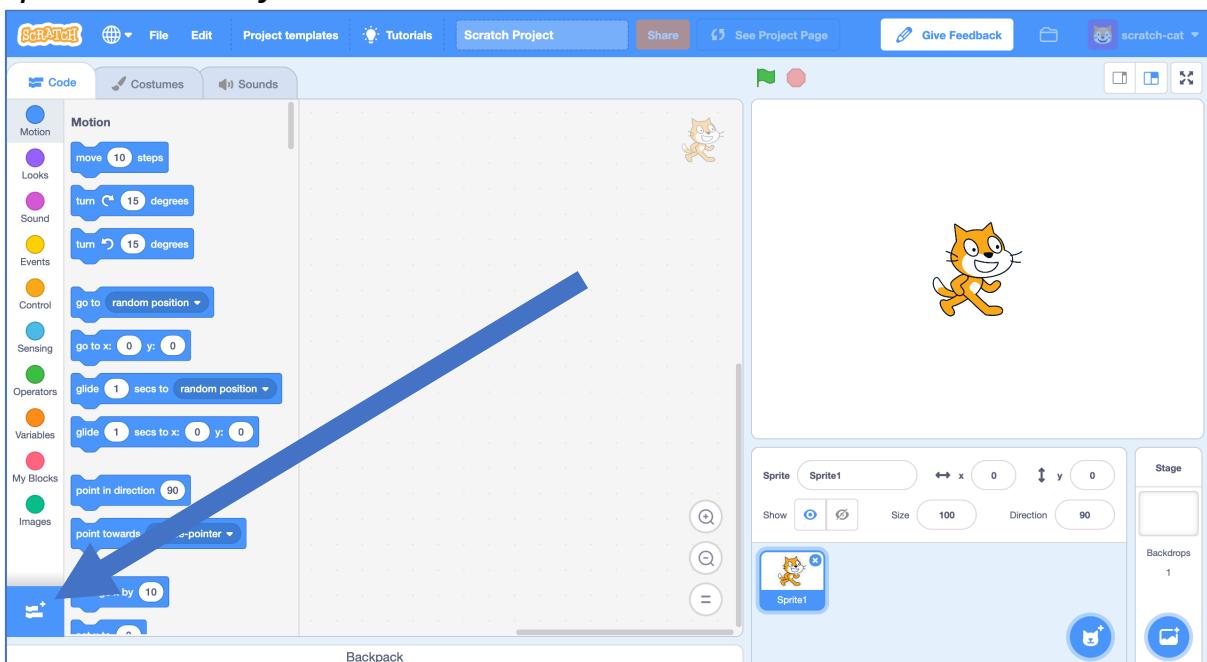
This project worksheet is licensed under a Creative Commons Attribution Non-Commercial Share-Alike License
<http://creativecommons.org/licenses/by-nc-sa/4.0/>

This project requires a **microphone**. If you don't have a computer with a microphone, you might prefer to try a different worksheet.

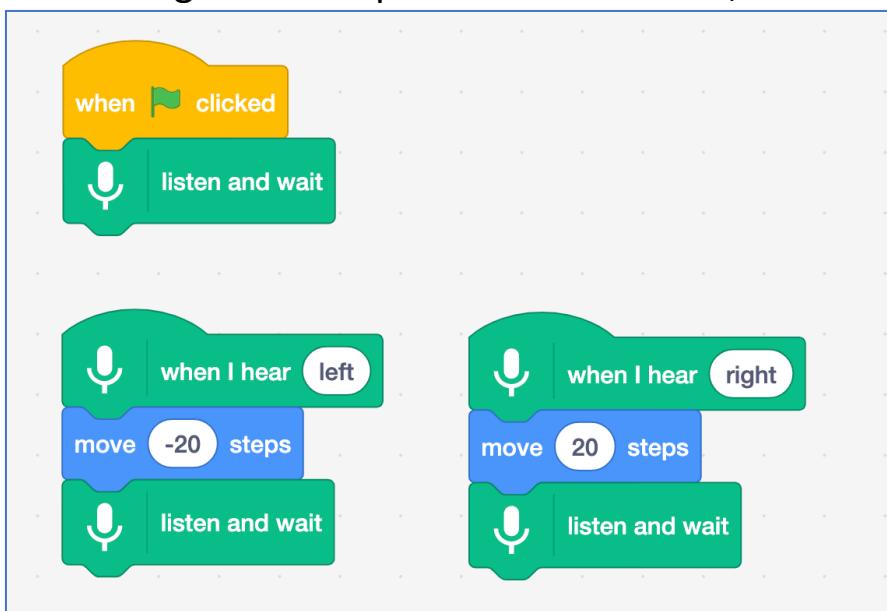
1. Go to <https://machinelearningforkids.co.uk/scratch3/>

2. Load the **Speech to Text** extension

Click on the Extensions (plus) button in the bottom left, and then choose Speech to Text from the list.

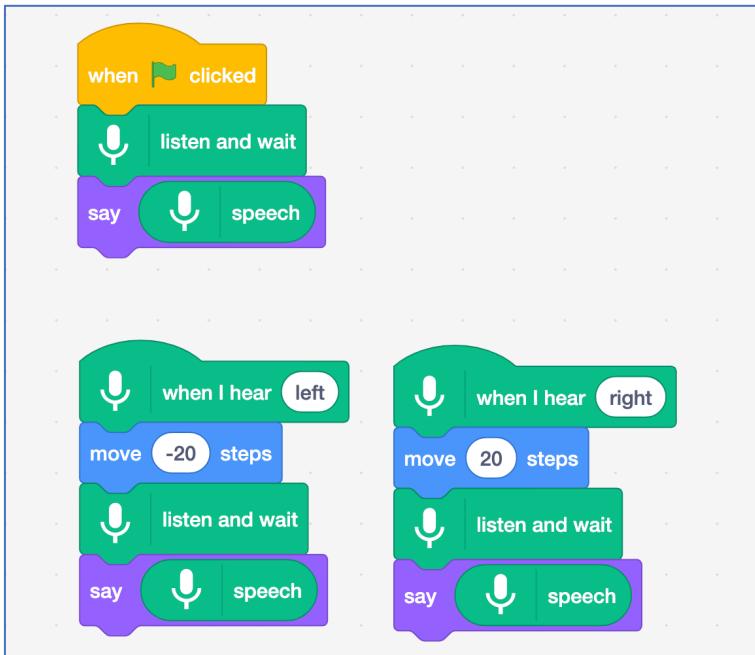


3. Using the new Speech to Text blocks, create the following scripts.



4. Click on the **Green Flag** and give it a try

Say “left” or “right”. The cat should move in the direction that you tell it to. Try and move it back and forth across the screen using your voice. It can be difficult to get it to work. Try to speak calmly and clearly. If it doesn’t work, modify your script to show what it thinks you’re saying:



What have you done so far?

You've used **speech recognition** to control a character in Scratch. To get this working quickly, you've used a machine learning model that has already been trained for you. This is a general machine learning model that has been trained to recognize English dictionary words.

Next, you'll train a machine learning model for yourself to see how it was done.

For the next part of the project, you'll use your voice to control an alien character that doesn't understand English! You'll invent two new words, that wouldn't be found in an English dictionary, to control your character and train a machine learning model to recognize them.

5. Invent your alien language!

You need two words – an alien word for “left” and an alien word for “right”. Invent new words that wouldn’t show up in an English dictionary. They can be random noises as long as you can repeat them in the same way every time and will be recognisably different from each other. If you don’t want to make weird noises with your voice, that’s okay - find other ways to make noises. You can click your fingers, clap your hands, squeeze a squeaky toy or do anything else you can think of!

6. Go to <https://machinelearningforkids.co.uk/>

7. 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.

8. Click on “Projects” on the top menu bar

9. Click the “+ Add a new project” button.

10. Name your project “Alien Language” and set it to learn how to recognise “sounds”.

Click the “Create” button

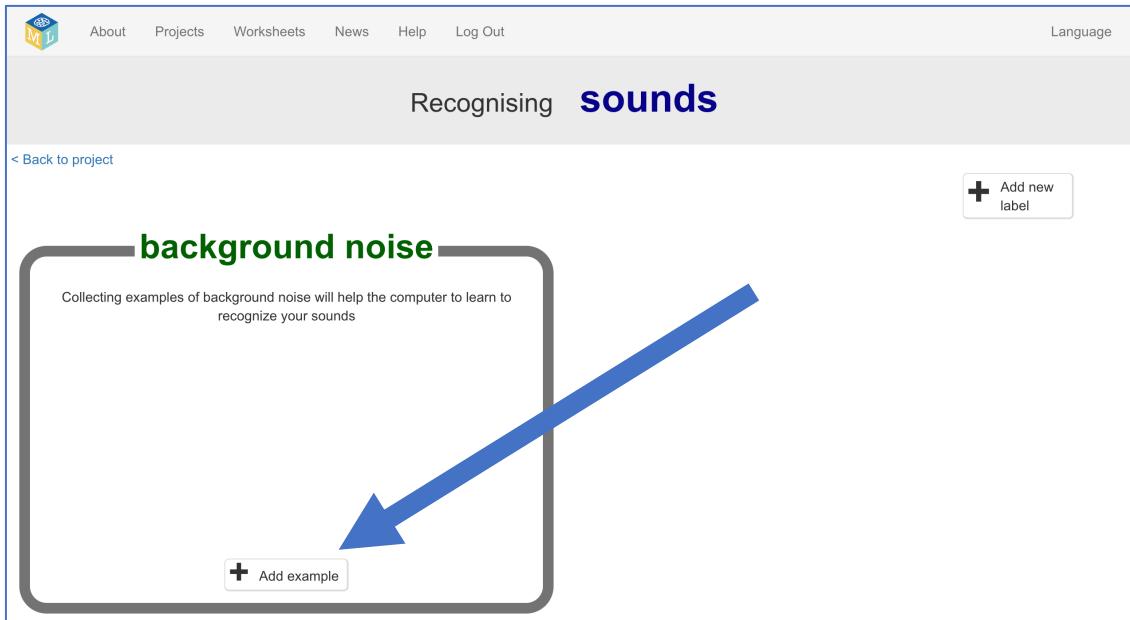
The screenshot shows a web interface for creating a machine learning project. At the top, there's a navigation bar with links for About, Projects, Worksheets, News, Help, Log Out, and Language. Below the navigation is a large button labeled "Start a new machine learning project". The main form area has fields for "Project Name *" containing "Alien Language" and "Recognising *" containing "sounds". A tooltip box appears over the "sounds" field with the text: "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 or multiple choices, choose "numbers". For voices and sounds, choose "sounds".". At the bottom right of the form are "CREATE" and "CANCEL" buttons.

11. You should now see “**Alien Language**” in the list of your projects.

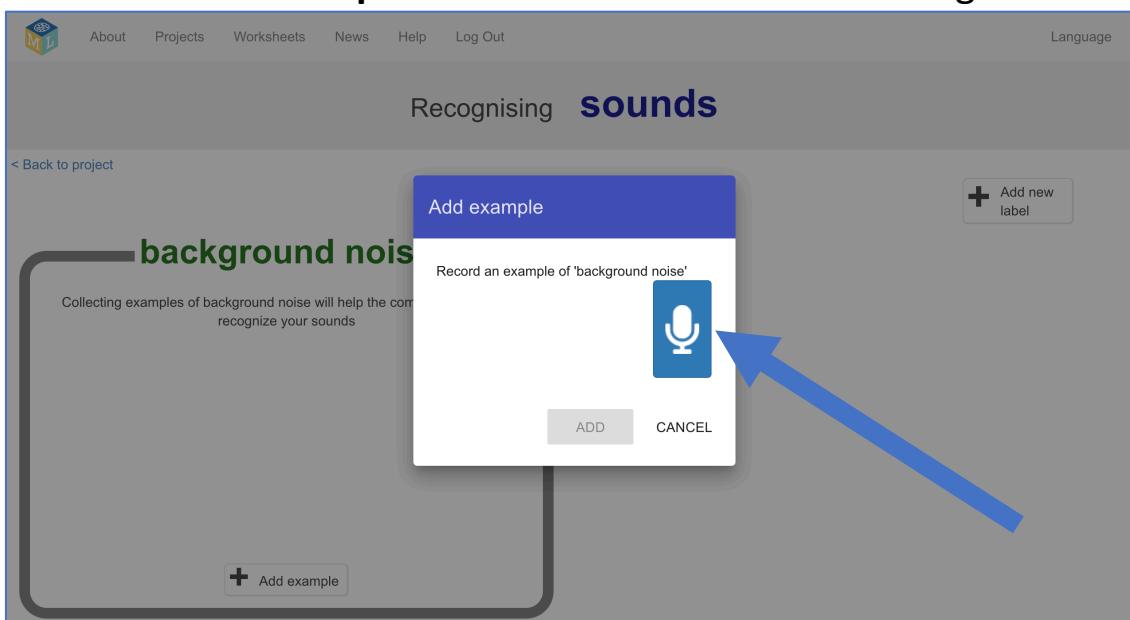
Click on it.

12. Click on the **Train** button to start collecting examples.

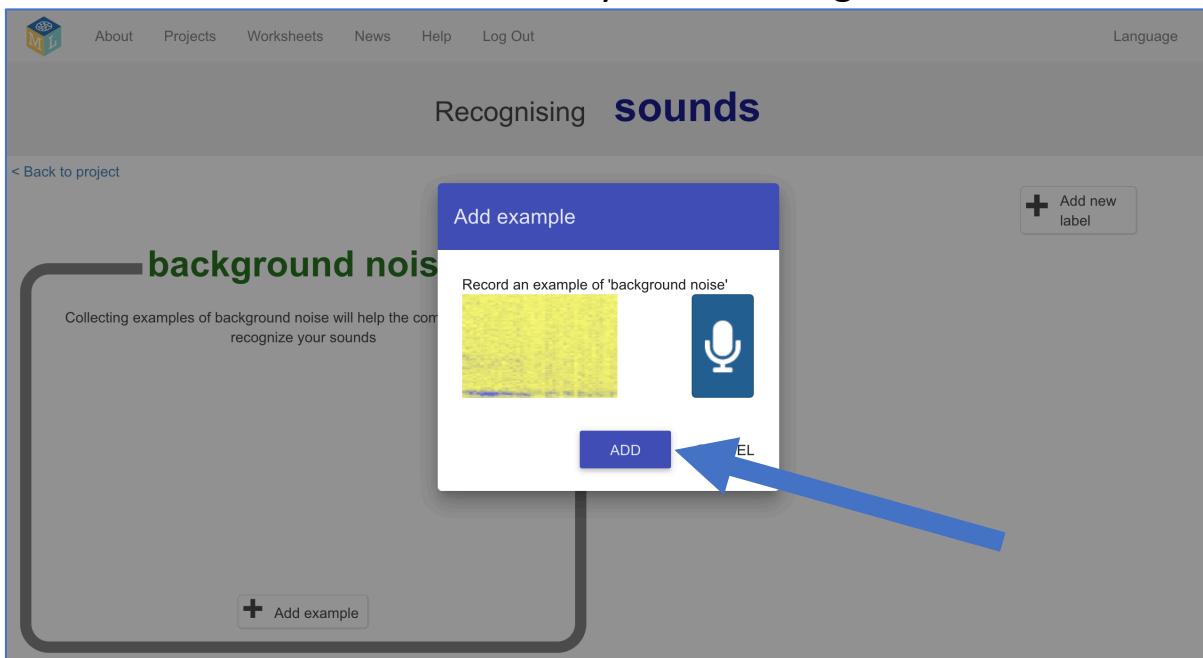
13. Click on the **Add example** button in the **background noise** bucket
Recording background noise will help your machine learning model to tell the difference between the sounds you will train it to recognize, and the background noise where you are.



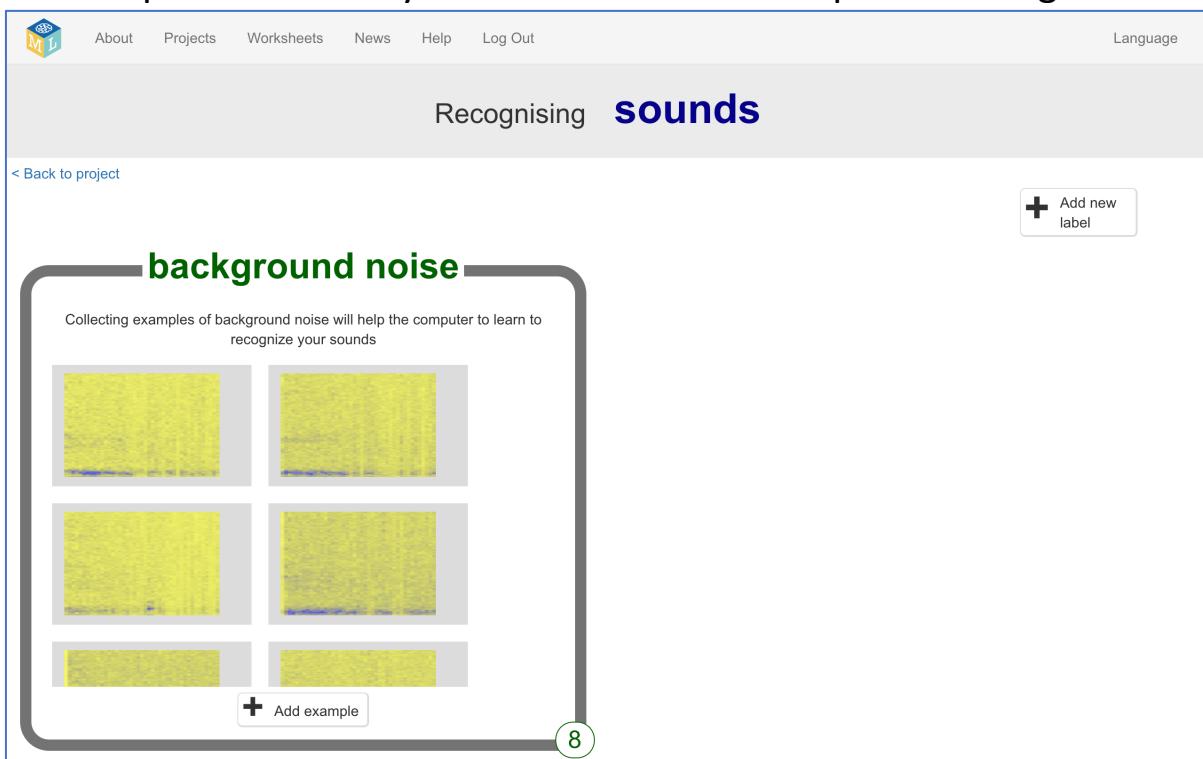
14. Click the **microphone** to record 2 seconds of background noise



15. Click the **Add** button to save your recording



16. Repeat that until you have **at least 8** examples of background noise



17. Click the **Add new label** button in the top right, and create a new training bucket called “left”

18. Click the Add example button in the new left bucket

The screenshot shows the SoundCloud interface with a project titled "Recognising sounds as left". On the left, there is a "background noise" bucket containing four spectrograms. On the right, there is a "left" bucket containing four spectrograms. A large blue arrow points from the "background noise" bucket towards the "left" bucket. In the top right corner of the "left" bucket, there is a button labeled "+ Add new label". Below each bucket, there is a green circle with the number "8" and a button labeled "+ Add example".

19. Record at least 8 examples of your alien noise for “left”

The screenshot shows the SoundCloud interface with a project titled "Recognising sounds as left". Both the "background noise" and "left" buckets now contain eight spectrograms each. The green circles with the number "8" are located below both buckets, and the "+ Add example" buttons are still present.

20. Click the Add new label button in the top right, and create a new training bucket called “right”

21. Record at least 8 examples of your alien noise for “right”

The screenshot shows a web-based application for sound recognition. At the top, there's a navigation bar with links for About, Projects, Worksheets, News, Help, Log Out, and Language. Below the navigation is the title "Recognising sounds as left or right". Underneath the title, there's a link "< Back to project". The main area is divided into three sections: "background noise", "left", and "right". Each section contains a grid of 8 spectrograms. Below each grid is a button labeled "+ Add example". A small "8" is placed at the bottom right of each section.

22. Click the “Back to project” link in the top left

23. Click the Learn & Test button

The screenshot shows a project titled "Alien Language". It features three main buttons: "Train", "Learn & Test", and "Make". A large blue arrow points directly at the "Learn & Test" button. Below each button is a brief description: "Train" says "Collect examples of what you want the computer to recognise" and has a "Train" button. "Learn & Test" says "Use the examples to train the computer to recognise sounds" and has a "Learn & Test" button. "Make" says "Use the machine learning model you've trained to make a game or app, in Scratch or in Python" and has a "Make" button.

24. Click “Train new machine learning model”

The screenshot shows a page titled "Machine learning models". It has two main sections: "What have you done?" and "What's next?". A large blue arrow points to the "Train new machine learning model" button located at the bottom of the "What have you done?" section. The "What have you done?" section includes a list of collected examples: "8 examples of _background_noise_, 8 examples of left, 8 examples of right". The "What's next?" section includes a link to start training the model.

25. Once the training is finished, click the **Start listening** button to test your machine learning model

Make one of the sounds you've trained the computer to recognize as meaning "left" or "right". If your machine learning model recognizes it, it will display what it thinks you did.

The screenshot shows the Scratch AI interface. At the top, there are two sections: 'Train' (with 8 examples of left and 8 examples of right) and 'Test' (with a note about collecting more examples if the computer is getting things wrong). Below these is a large 'Test' section containing a sound recording input field, a 'Start listening' button, a 'Stop listening' button, and a text area that says 'Recognised as right with 94% confidence'. A large blue arrow points from the 'Train & Test' section towards the recognition text.

26. If you're not happy with how the model is working, go back to the **Train** page and add more examples to all three training buckets.

27. When you're happy with your machine learning model, click on the **Make** button

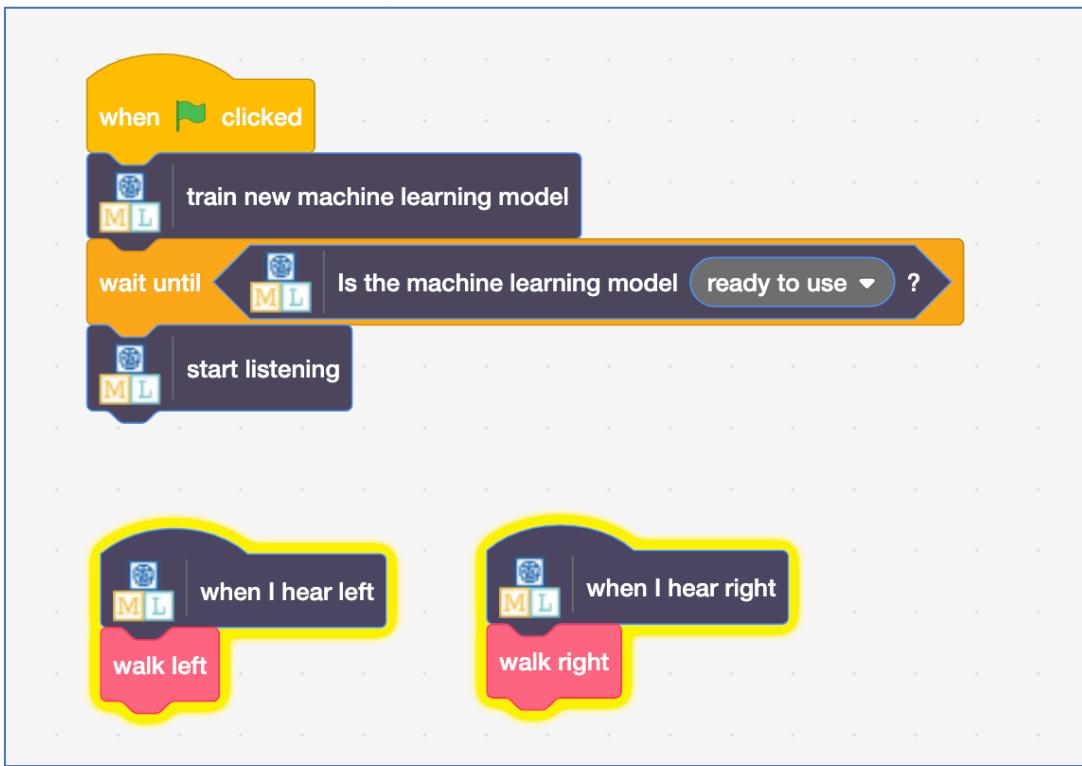
The screenshot shows the Scratch AI interface. At the top, there are three main sections: 'Train' (for collecting examples), 'Learn & Test' (for training the computer to recognise sounds), and 'Make' (for using the trained model to make a game or app). The 'Learn & Test' section contains a note about using examples to train the computer. The 'Make' section contains a note about using the trained model to make a game or app in Scratch or Python. A large blue arrow points from the 'Learn & Test' section towards the 'Make' button.

28. Click on the **Scratch 3** button and then click **Open in Scratch 3**

29. Click on the **Project templates** button at the top of the screen and open the "Alien Language" project template

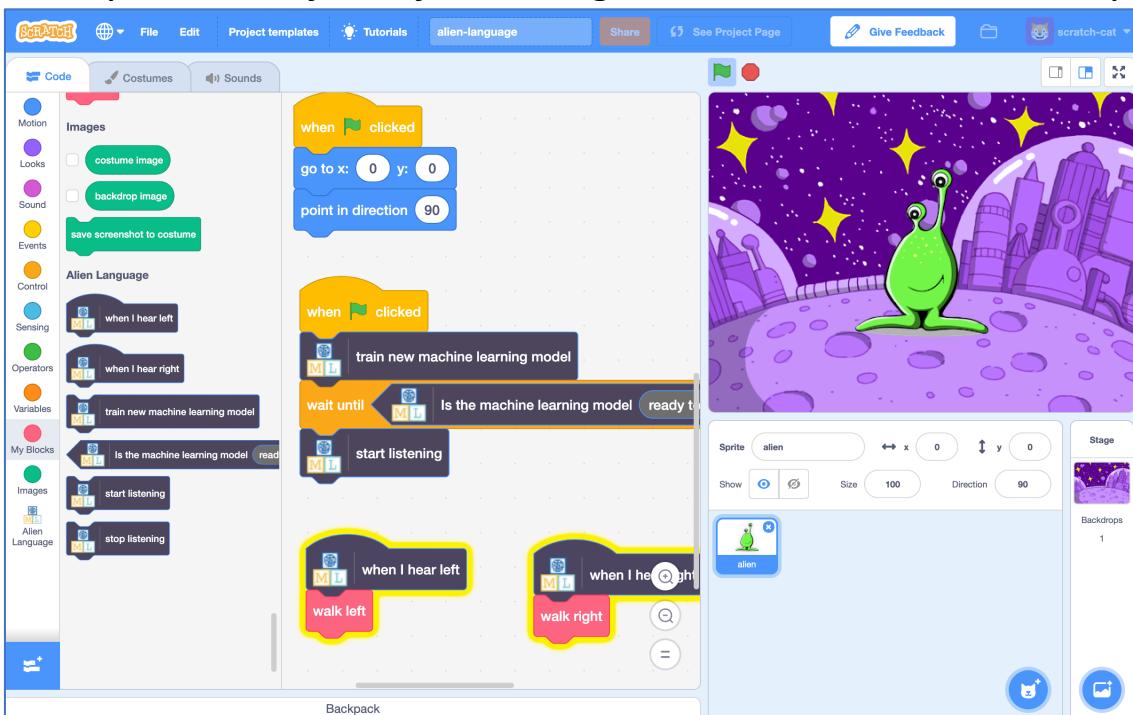
30. Add the following script to the alien sprite

*There are already some scripts in the alien sprite to put it in the right place at the start and animate how it walks. **Don't** delete these.
You can add these scripts underneath them.*



31. It's time to test! Click the Green Flag

Make your noises for "left" and "right" to tell the alien which way to walk.



What have you done?

You've trained your own machine learning model to do speech recognition. You used that to control a character in Scratch.

Unlike the pre-trained model you used before, which has been trained to recognize tens of thousands of words, you've only trained it to recognize two different words. But the principle is the same.

You've also seen the importance of training the machine learning model to work with a certain background noise.

Can you think of an example of a system like this you've seen used before? For example, some cars use speech recognition systems that have been trained to recognize the different commands you can give to the in-car computer. What other examples have you used?

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?

Add new commands

Try adding two more training buckets for “up” and “down” so you can control the alien to move in all four directions.