In this project you will make a character that reacts to what you say . 在这个项目中你将会创造一个角色，他会对你说的话作出反应

If you compliment it, it will look happy. 如果你夸他，他会开心。

If you insult it, it will look sad.

如果你骂他，他会难过。

At first, you’ll program a list of rules for what is kind and what is mean, and learn why that approach isn’t very good.

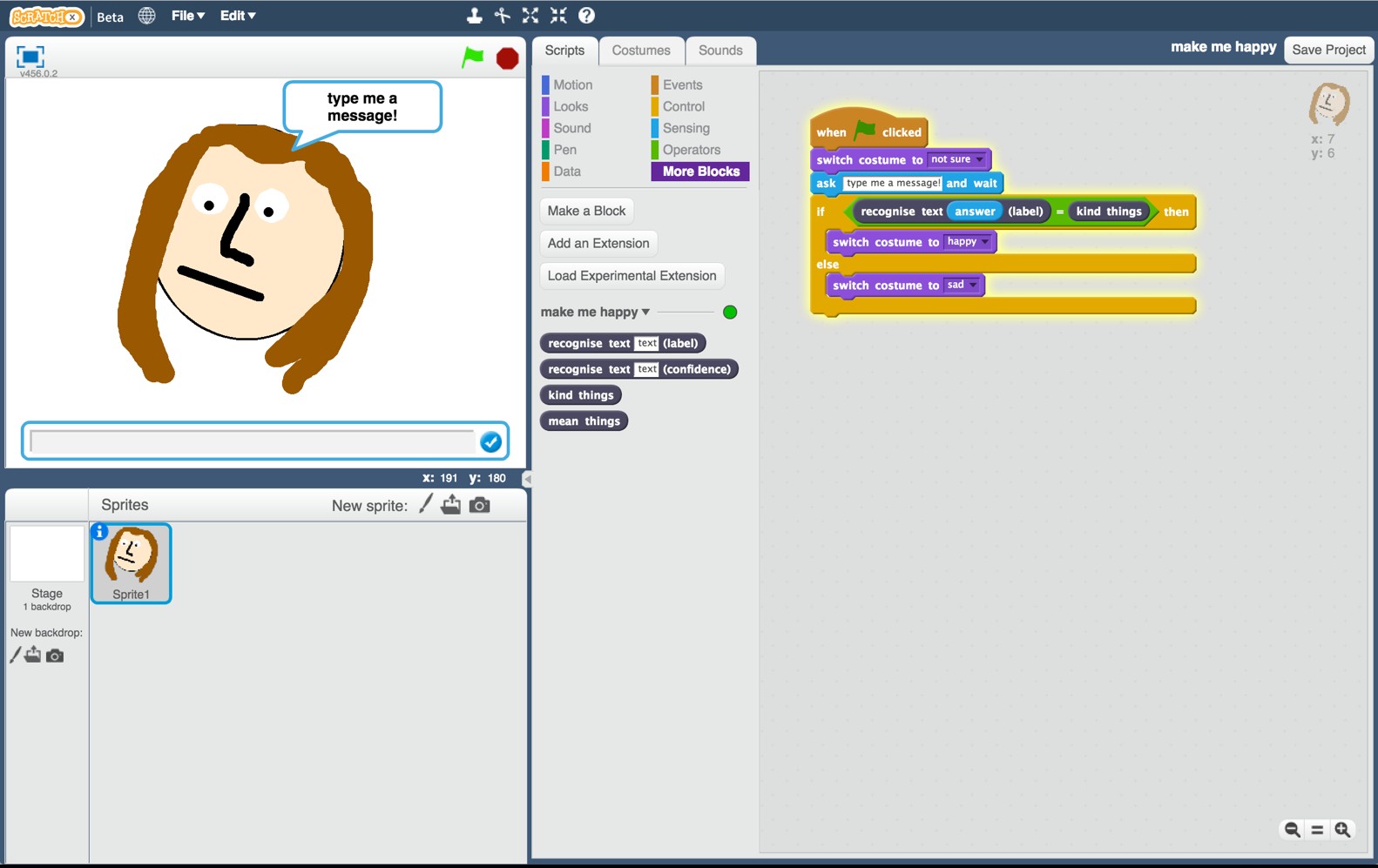
首先，你要制定一系列好话和坏话，知道为什么这个方法行不通。

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

然后，通过数据，你将教会电脑识别好坏信息。



Make me happy**让我开心**



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

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

浏览器打开网址machinelearningforkids.co.uk

1. Click on “Get started”

点击“开始”

1. 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.

点击“登录” 输入用户名与密码。如果你没有用户名，让你的老师和小组组长为你创建一个。如果你记不住用户名或者密码，让你的老师或小组组长为你重置。

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

点击菜单上方的“项目”

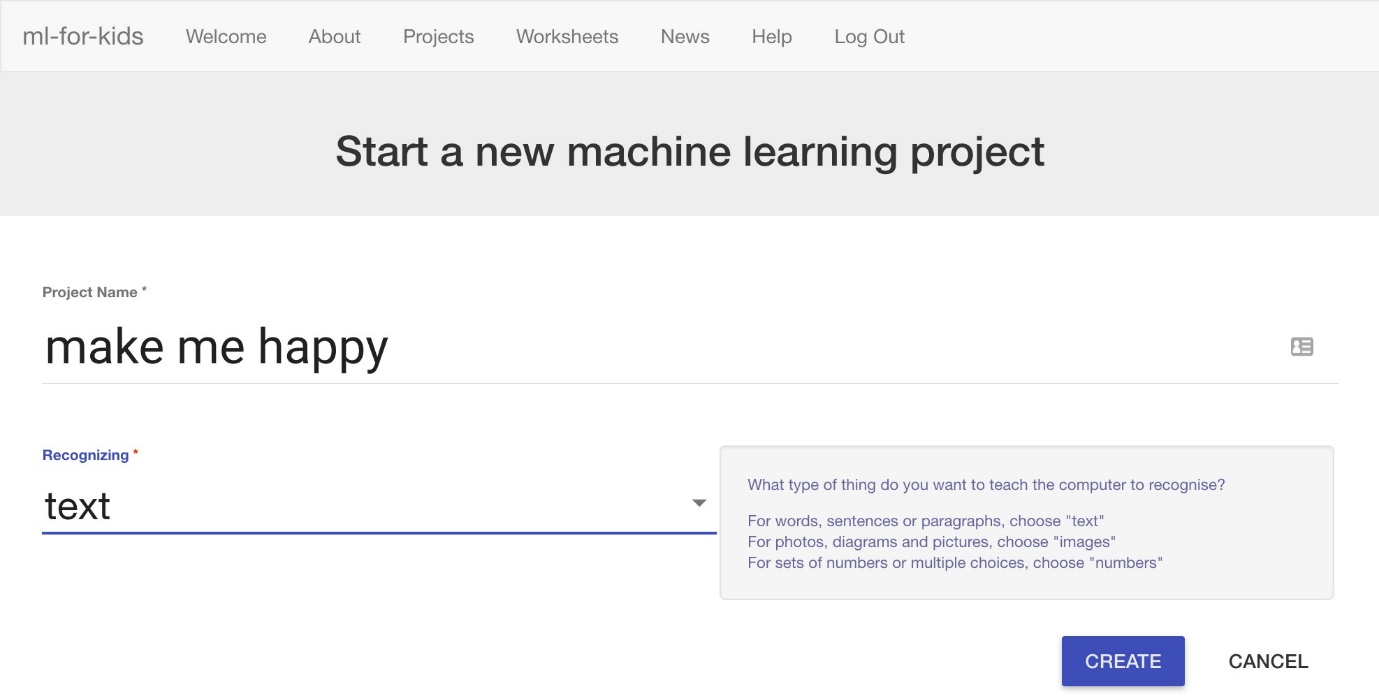
1. Click the “+ Add a new project” button.

点击“添加新项目”按钮

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

将项目命名为“让我开心”，将其设置识别为“文字”

1. Click the “Create” button

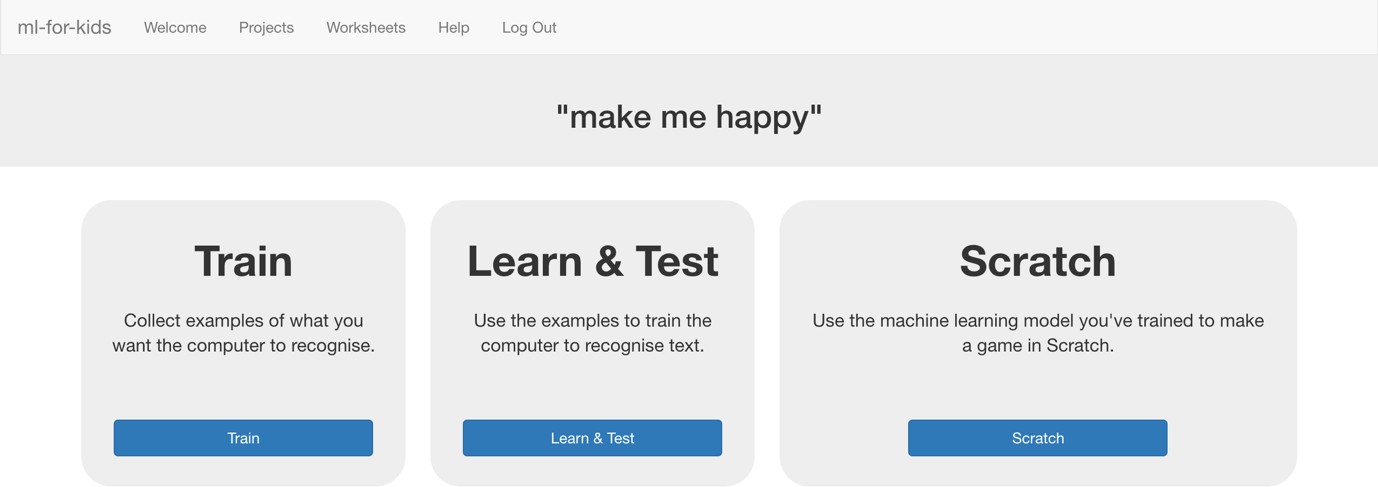


点击“创造”

1. You should now see “make me happy” in the list of your projects. Click on it.

你会在目录中看到“让我开心” 点击它

1. Start by getting a project ready in Scratch. Click 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.*

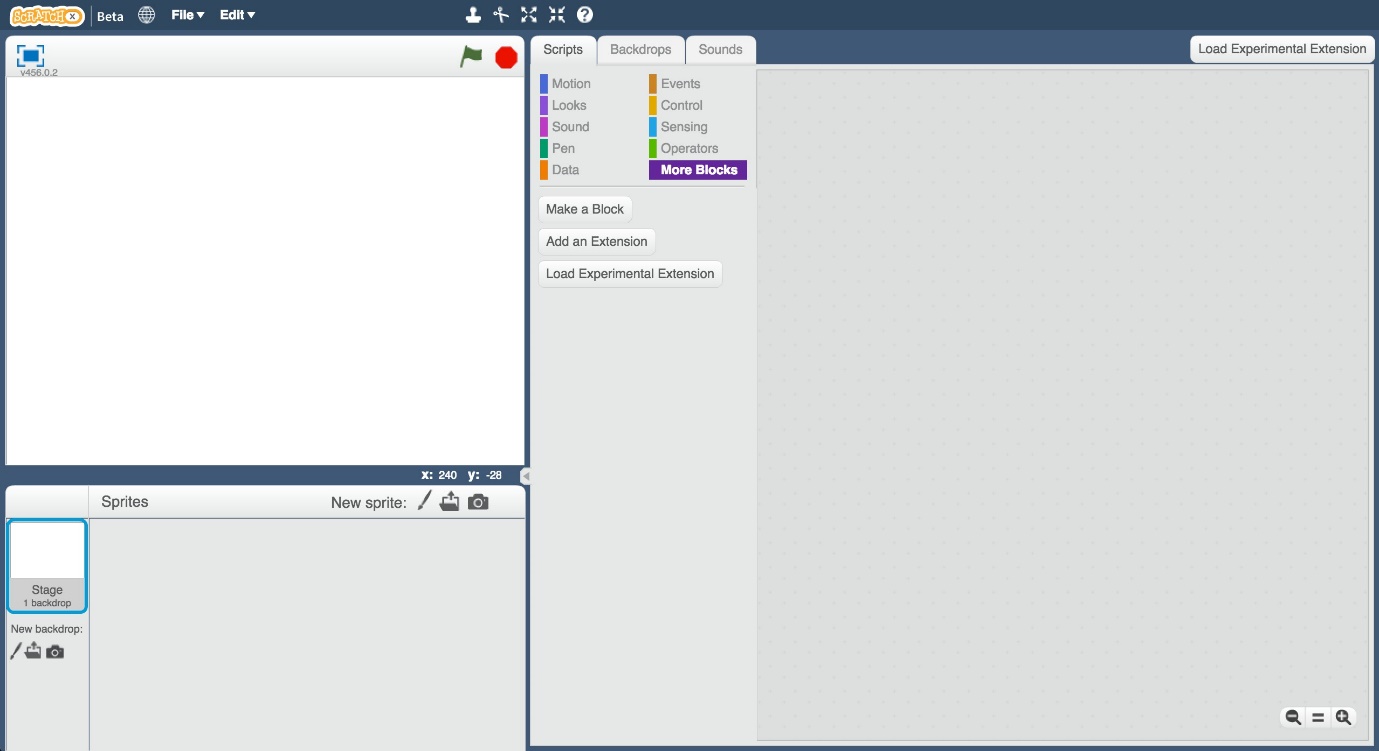


点击**脚本**按钮 见面会跳出提示你还未进行机器学习，点击**Scratch by itself** 打开脚本

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

点击绘图笔

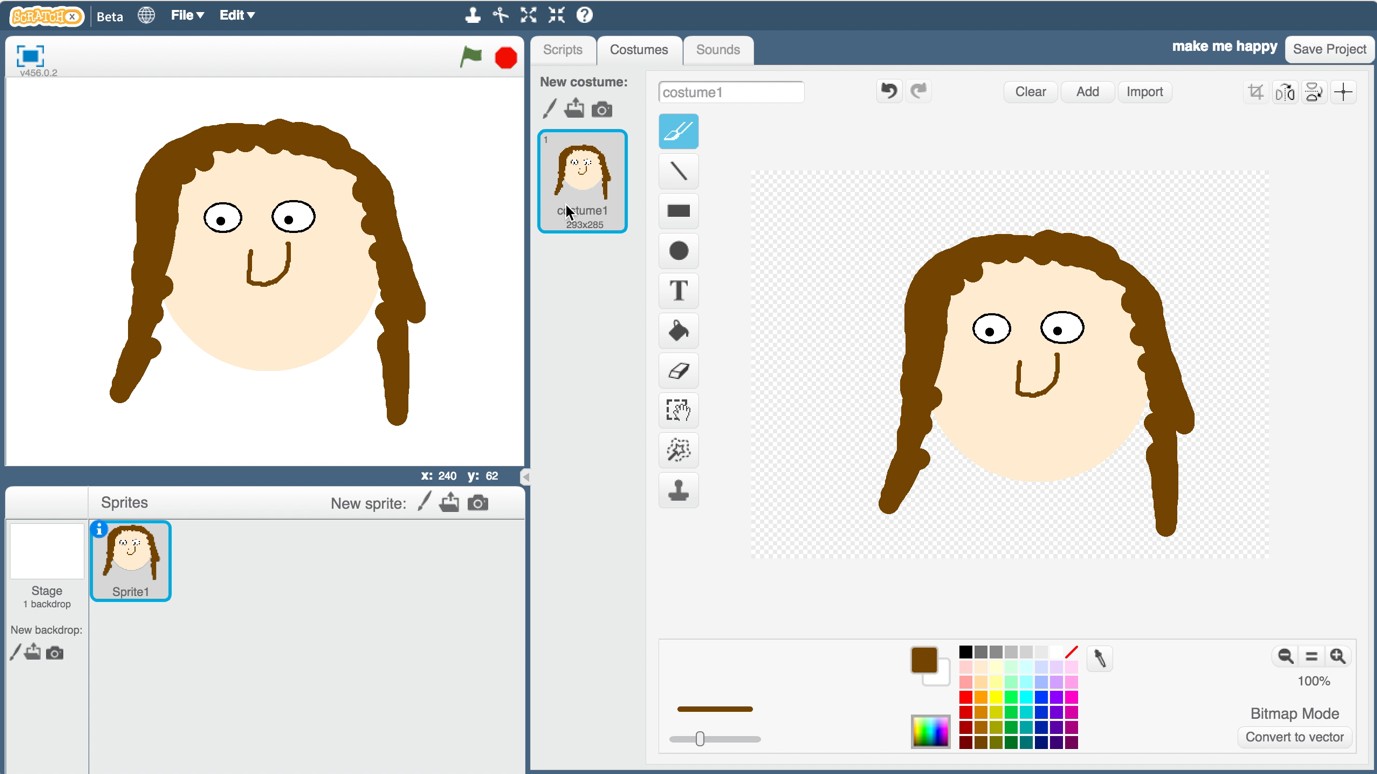
*There are a few similar looking paintbrush buttons – make sure you click the one marked below.*



*窗口中有几只相似的笔，请确认你选中下图标记的笔*

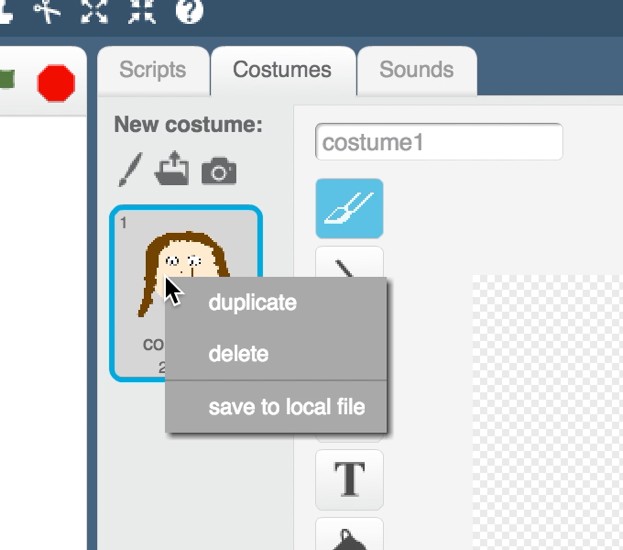
1. Draw a face, without a mouth, in the sprites editor on the right.

在右边的编辑器中画一张脸，不画嘴巴。



1. Right-click on the costume, and click “Duplicate”.Do that again so you have three copies of the costume.

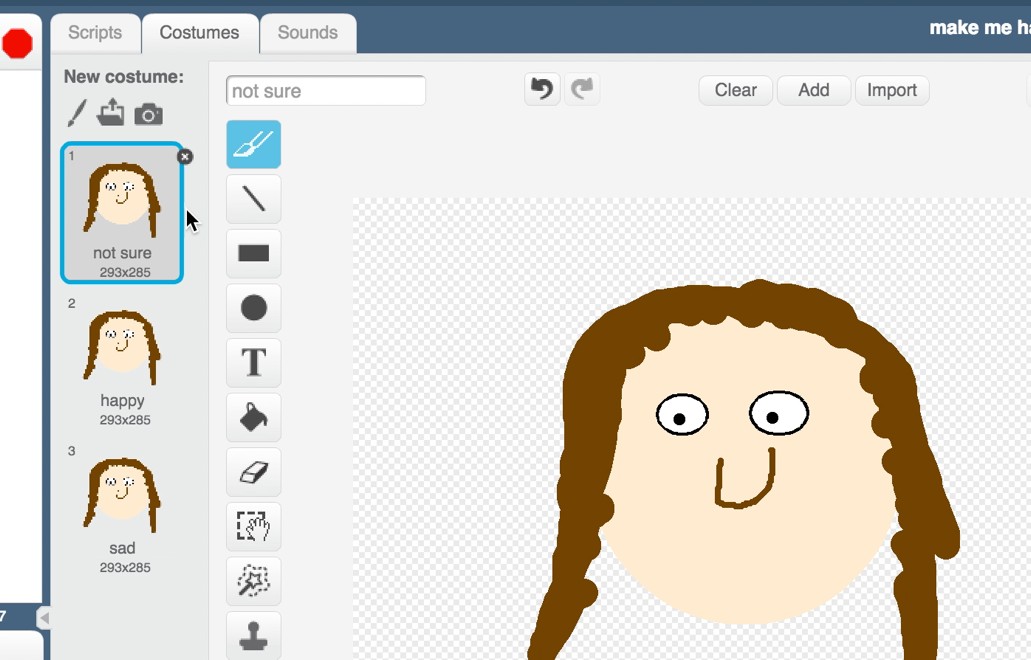
在constume中单击右键，点击“复制” 再操作一遍 以得到三份。



1. Name the three costumes “not sure”, “happy” and “sad”.Type the names into the white box shown by the arrow below.

将三份分别命名为“不确定”“快乐”“悲伤” 在下图箭头所示处输入

1. Draw a mouth on each of the costumes.



给每张脸都画上表情。

1. The “not sure” face should be a straight line.

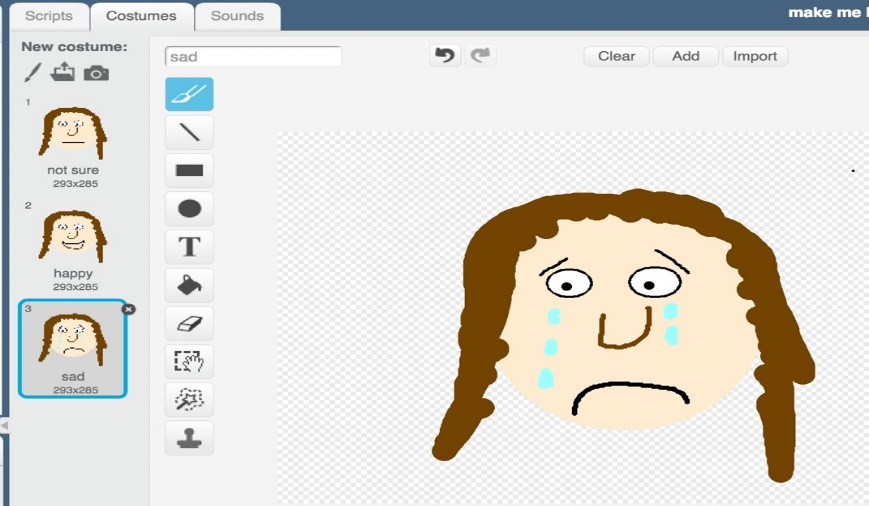
“不确定”的脸画一条直线。

1. The “happy” face should have a smile.

“快乐”的脸画一个微笑

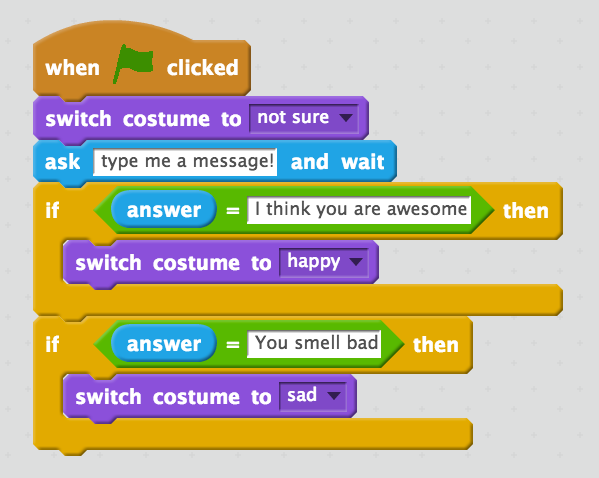
1. The “sad” face should look sad.

“悲伤”脸很悲伤



1. Click the “**Scripts**” tab, and enter the following script.

点击**“脚本”**进入以下脚本



1. Save your project.

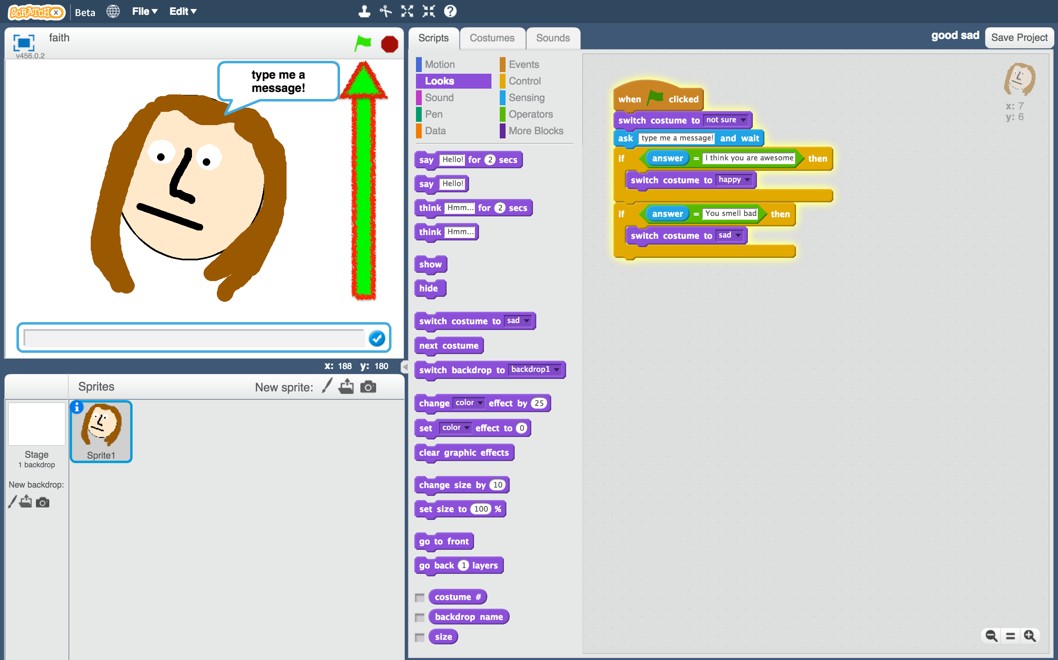
保存项目

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

点击文件 ->保存项目

1. Click the green flag to test.

点击绿色箭头测试



1. Type in a message and watch it react!

输入文字进行测试

1. Type “I think you are awesome” and press enter. The character smiles.

Click the green flag again and type “You smell bad”. The character cries. Type anything else, and the character’s face won’t change.

输入“我认为你很棒”按下回车键，人物微笑。再次点击绿色旗子，输入“你真臭”，人物哭泣。输入其他，人物没有表情。

**What have you done so far? 我们现在都做了什么**

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

你有一个能回应你的角色，并且设计程序让他微笑。

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

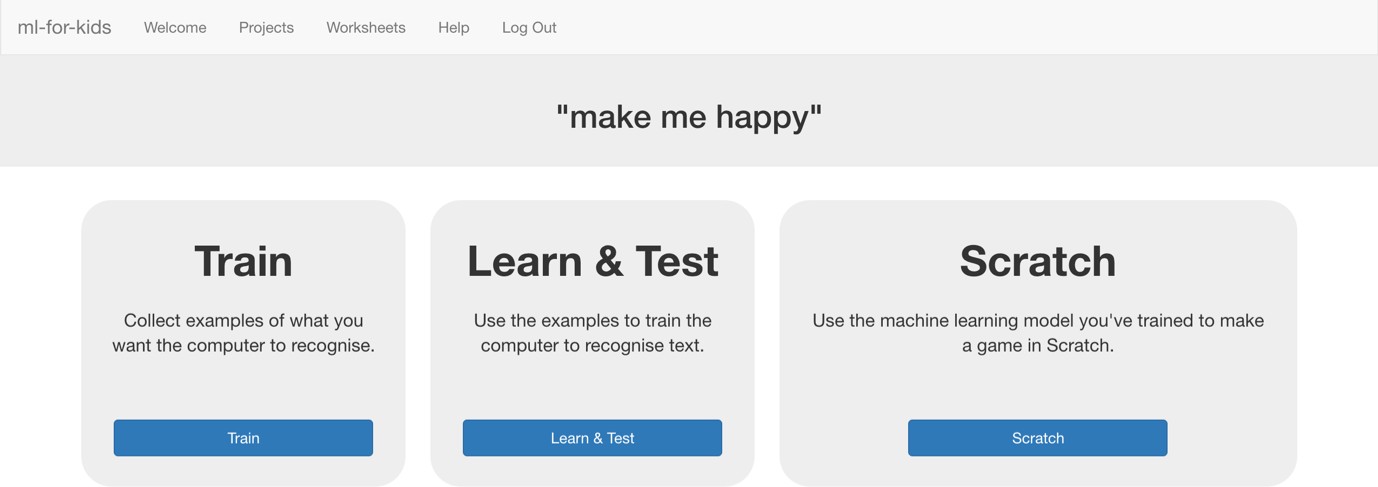
blocks. 如果你想要它有更多的回应，你应该加入更多的**if**区块。

The problem with this is that you need to predict exactly what messages the character will receive. Making a list of every possible message would take forever! 你要设计好人物接受的信息。一条一条写出来太慢了。

Next, we’ll try a better approach – teaching the computer to recognise messages for itself. 我们换一种方法—让电脑学会自己辨别消息

1. Close the Scratch window. 关闭脚本
2. You need examples to train the computer. Click the “< Back to project” link. Then click the Train button.

你需要更多数据训练计算机。点击“**回到项目”**链接，点击**“训练”**按钮

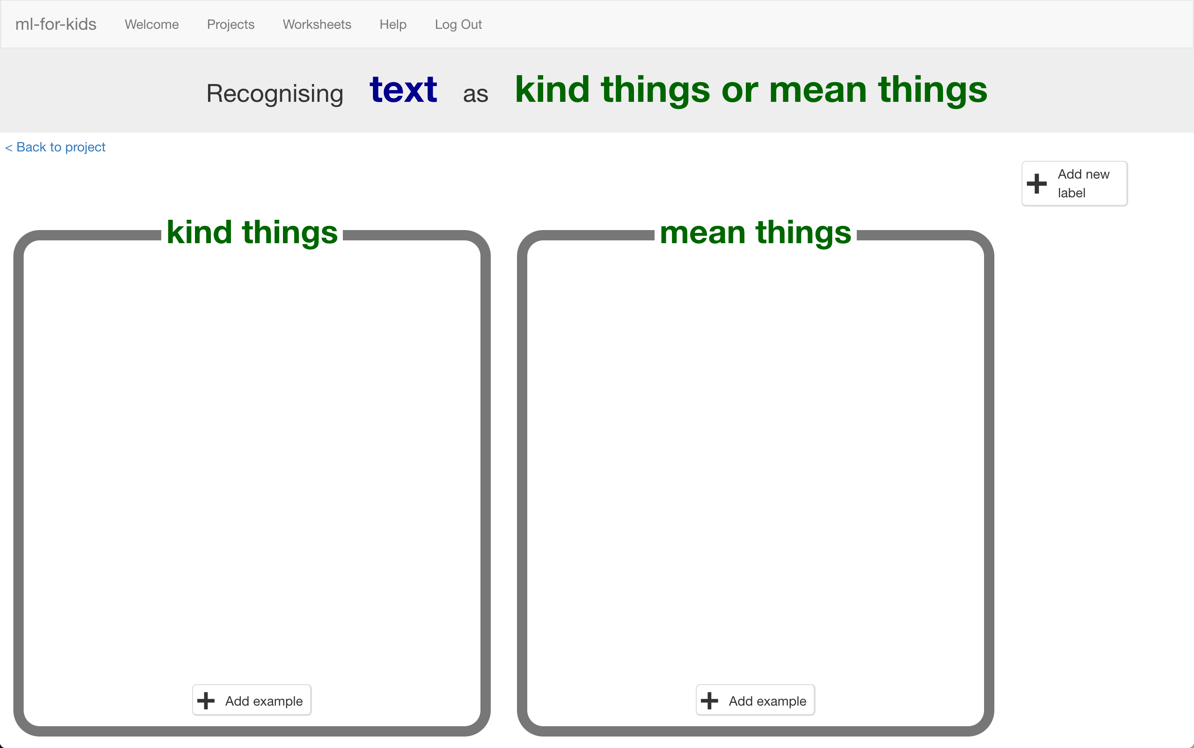


1. Click on “+ Add new label” and call it “kind things”.

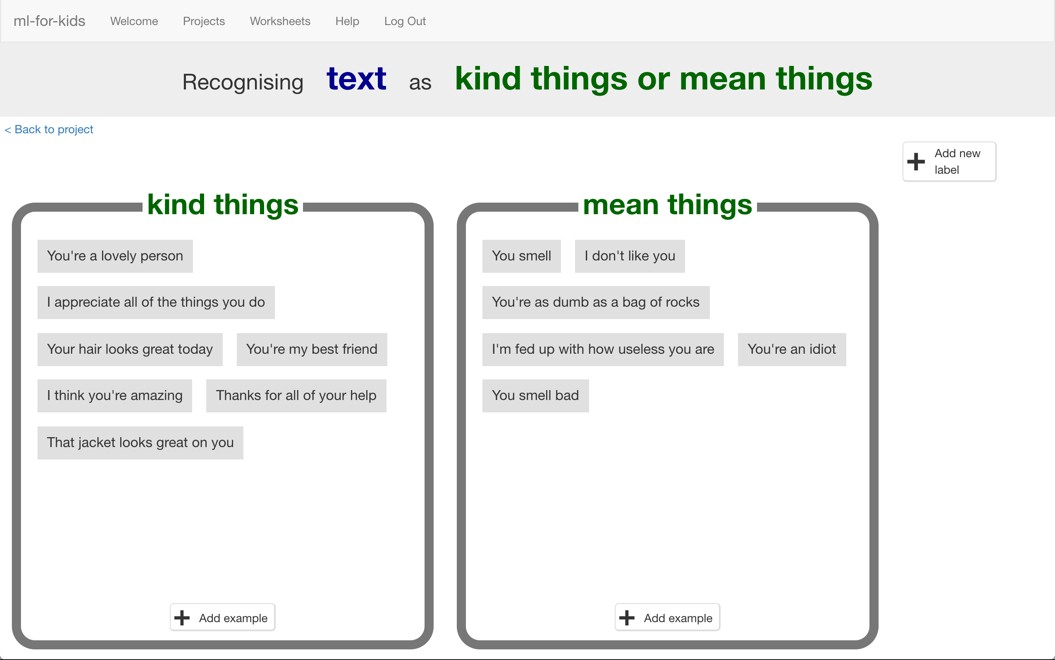
点击“添加新标签“，将其命名为”好话‘。

1. Do that again, and create a second bucket called “mean things”.

同理，命名另一个为“坏话”



1. Click the “**Add example**” button in the “kind things” bucket, and type in a kind message. 在“好话‘框中点击**“添加样本”**，输入好话。
2. Click on the “**Add example**” button in the “mean things” bucket, and type in a mean message. 在“坏话‘框中点击**“添加样本”**，输入坏话。
3. Repeat steps 21 and 22 until you’ve written at least **ten** examples of each.



重复步骤21，22知道你每个框中各有**十**句话。

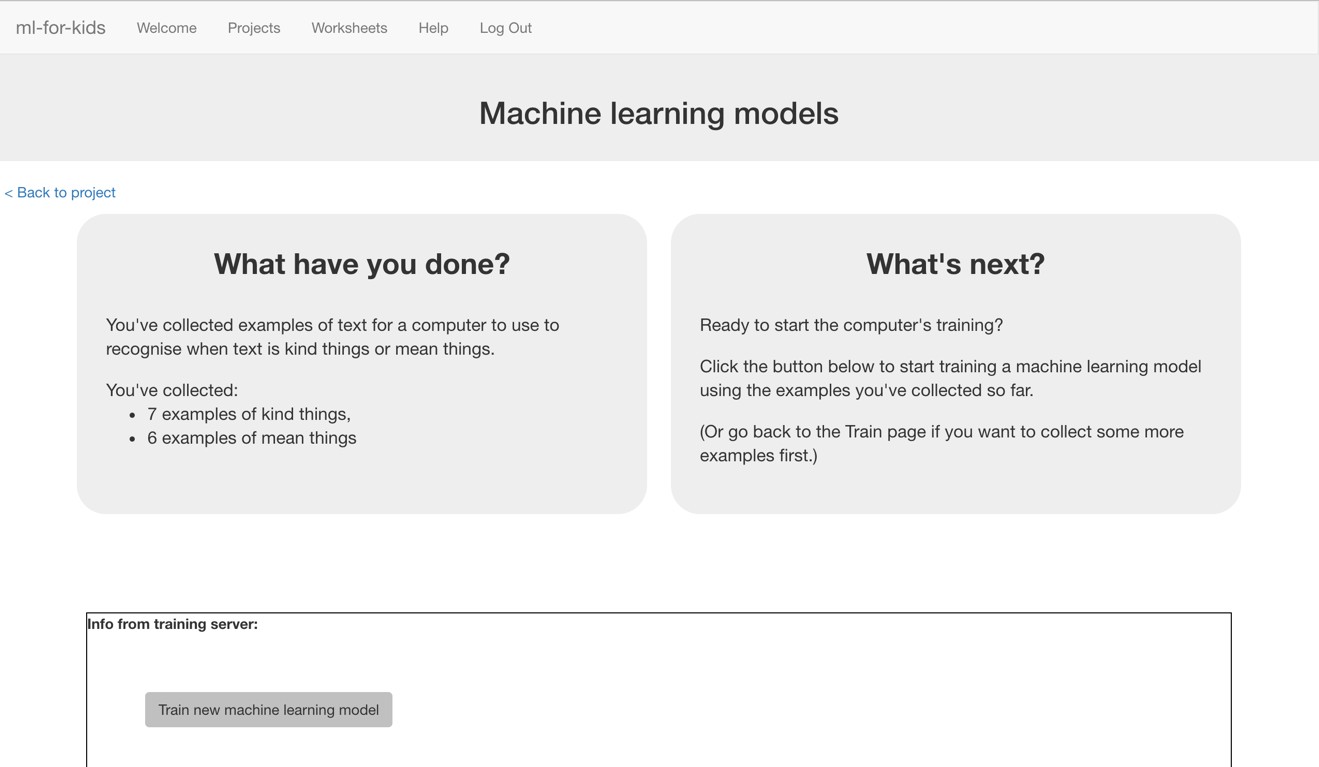
1. Click on the **“< Back to project**” link. Then click on the “**Learn & Test**” button.

点击**“回到项目”**链接，点击**“学习与测试”**

1. 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.*

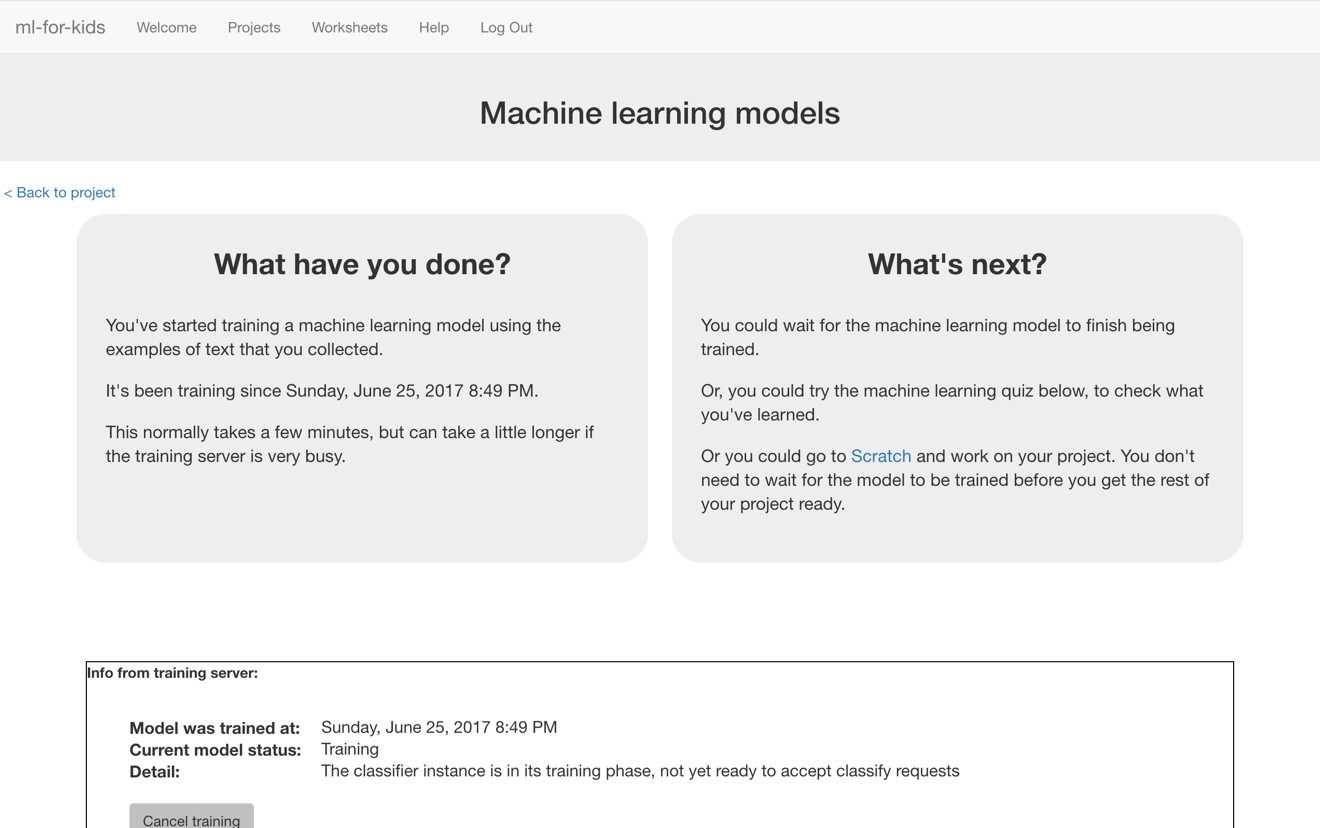


*点击“****训练新机器学习模型****“*

*只要你收集到了足够的样本，电脑就会开始自己学习*

1. Wait for the training to complete. This might take a few minutes.

等待训练完成，大概需要几分钟。



1. Once the training has completed, a Test box will be displayed. 训练完成后，会有一个信息框弹出。

Try testing your machine learning model to see what the computer has learned. 测试你的机器学习模型看看他学会了什么

Type something kind, and press enter. It should be recognised as kind. 输入好话，回车。

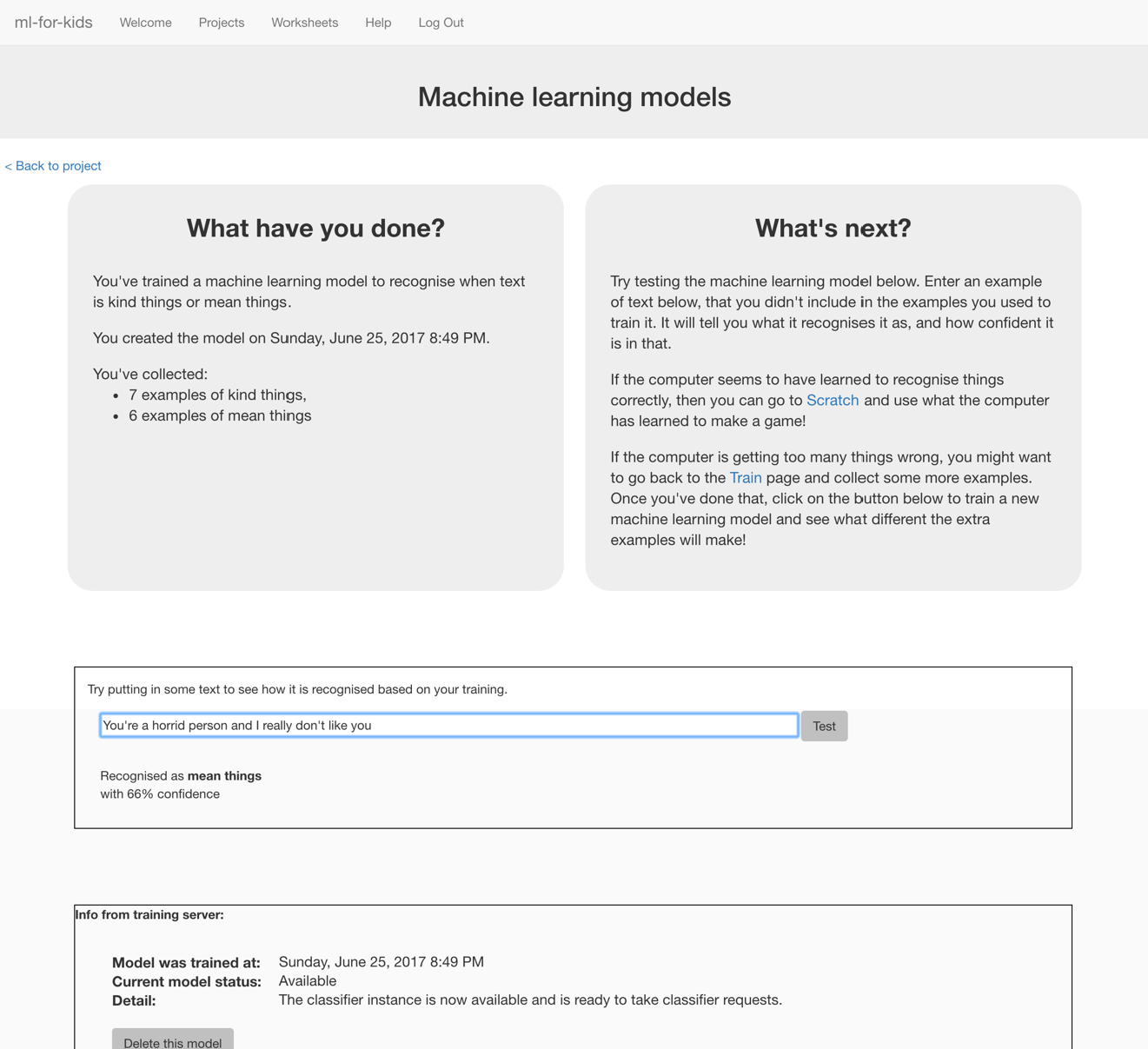
Type something mean, and press enter. It should be recognised as mean.输入坏话，回车。

*Test it with examples that you haven’t shown the computer before. 输入之前未输入框之中的话测试电脑*

*If you’re not happy with how the computer recognises the messages, go back to step 21, and add some more examples. 如果你对效果不满意回到步骤21，加入更多样本*

*Make sure you repeat step 25 to train with the new examples though!*

*请确保你重复步骤25 以新样本训练电脑*



**What have you done so far? 目前为止你做了什么**

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”. 你开始训练电脑识别好话坏话，而不是顶下规则让他按照规则来，你通过收集样本，这些样本用来训练机器学习“模型“

This is called “supervised learning” because of the way you are supervising the computer’s training. 这叫做“监督式学习“因为你在监督电脑训练

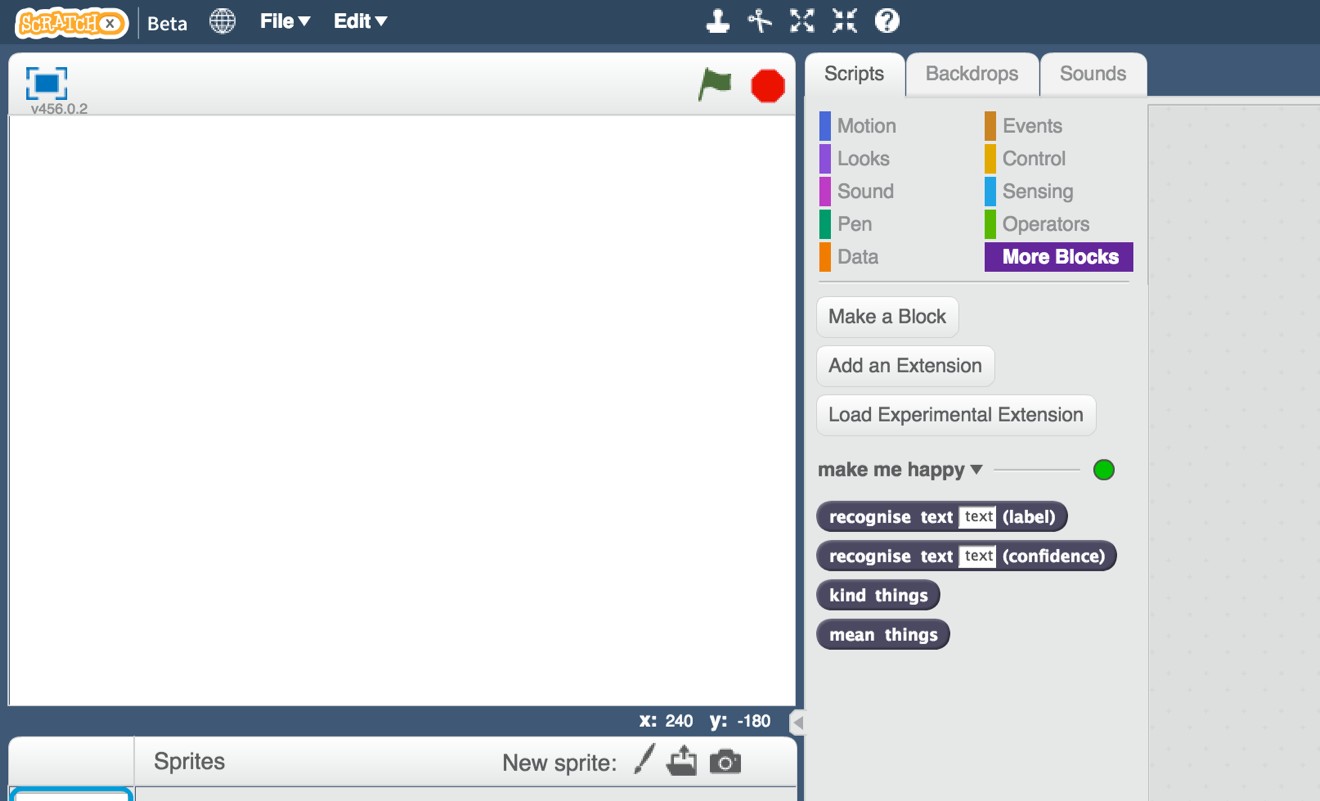
The computer will learn from patterns in the examples you’ve given it, such as the choice of words, and the way sentences are structured. These will be used to be able to recognise new messages. 电脑会从你给出的样本中学习，总结规律，比如选词，句型，这些都让电脑能够识别新信息。

1. Click the **“< Back to project**” link, then the “**Scratch**” button. *This page has instructions on how to use the new blocks in Scratch. Keep the page open if you need to check back on how to use them.*

点击**“回到项目”**链接，点击“**脚本**“按钮

1. Click the “**Open in Scratch**” button at the bottom to launch the Scratch editor. 点击页面底部“**打开脚本”**按钮启动脚本编辑器

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



*你可以在****“更多区块”****中看到四个新区块*

**Tips 提示**

**More examples! 更多样本！**

The more examples you give it, the better the computer should get at recognising whether a message is kind or mean. 样本越多电脑的识别能力越好。

**Try and be even 平均测试**

Try and come up with roughly the same number of examples for kind and mean. 好话坏话样本的数量应该差不多

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. 如果一方的样本过多，电脑可能学习到该方的更容易出现，导致出现错误。

**Mix things up with your examples 混合样本**

Try to come up with lots of different types of examples.

尽量提供不同的样本

For example, make sure that you include some long examples and some very short ones. 比如说，确保长句和短句的存在

1. Load the Scratch project you saved before. 加载之前存的项目

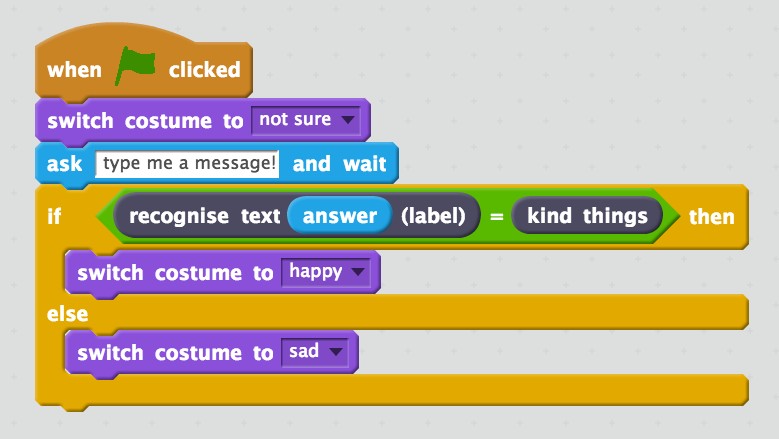
*Click on* ***File*** *->* ***Load Project*** *点击****文件 -> 加载项目***

1. Click on the “**Scripts**” tab, and update the script to use your machine learning model instead of the rules you made before. 点击**“脚本”**，更新脚本，应用机器学习，而不是你之前制定的规则。

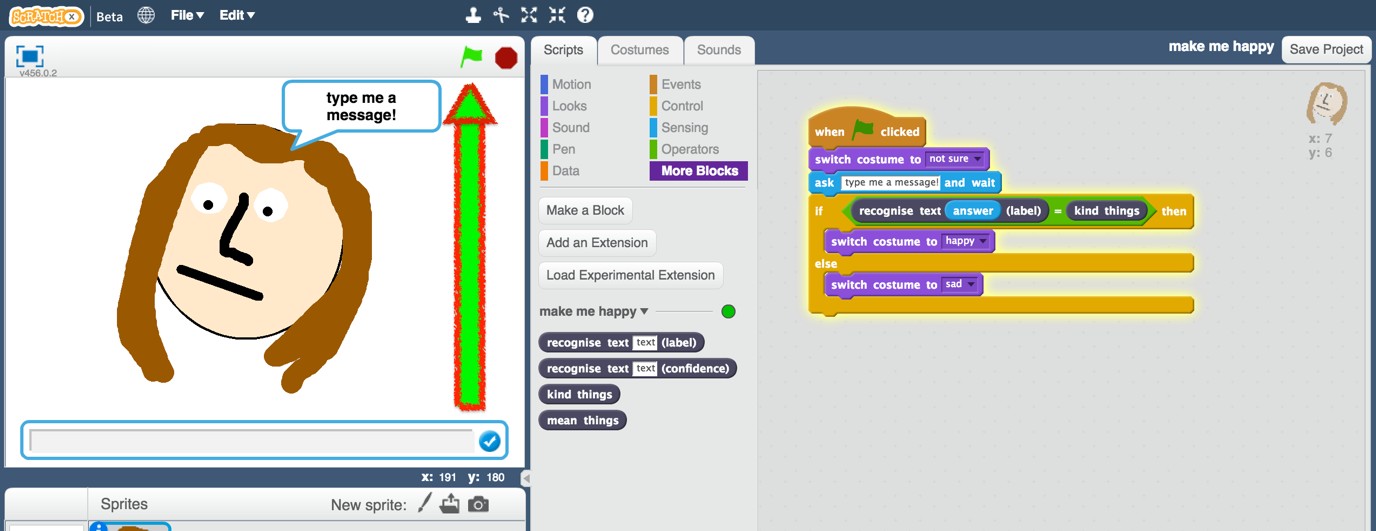
*The “recognise text … (label)” block is a new block added by your project. “recognise text … (label)”这是一个新的区块*

*”If you give it some text, it will return either “kind things” or “mean things” based on the training you’ve given to the computer. 如果你输入一些文字，他会回应好话或者坏话，基于你对于计算机的训练*

*You can use this to choose the costume to switch to. 你可以通过它控制人脸*



1. Click on the **green flag** to test again. 再次点击**绿色旗子**进行测试

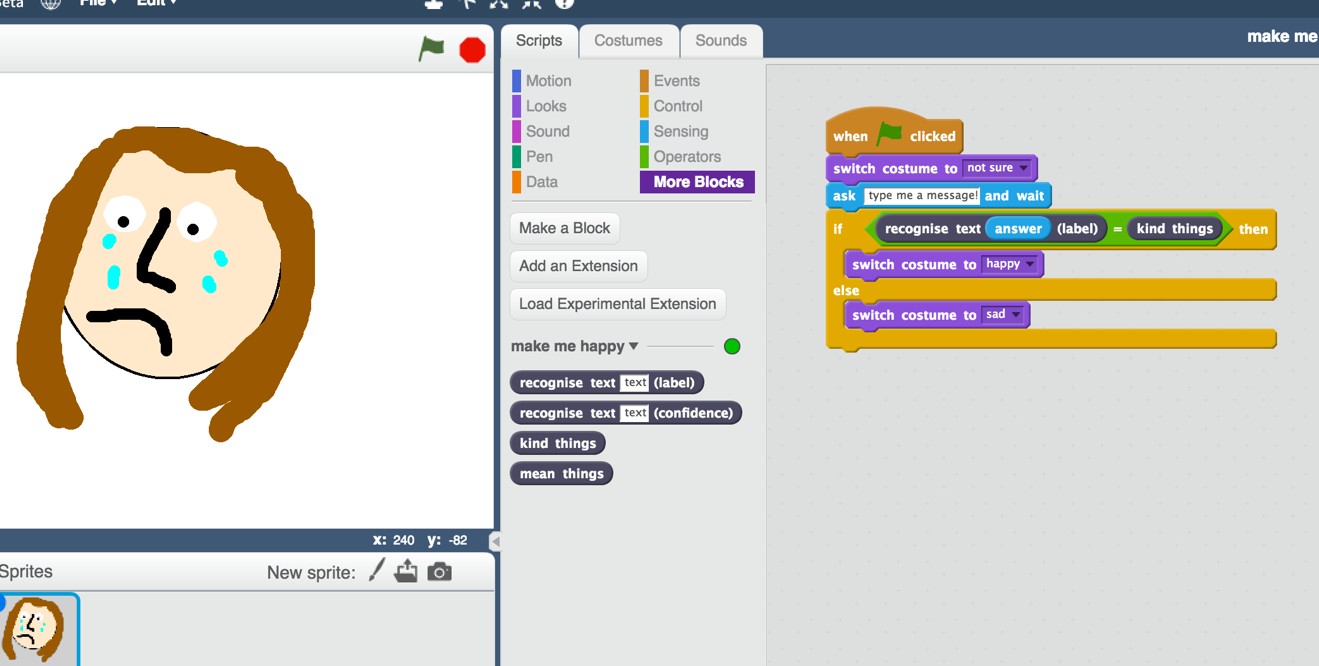


1. Test your project 测试项目

*Type a kind message and press enter. The character should smile. 输入好话，人物应该笑*

*Click the green flag again. Type a mean and unkind message and press enter. The character should look sad. 再次点击绿色旗子，输入坏话回车 人物看上去很难过*

***This should work for messages that you didn’t include in your training.这应该对于不在训练范围之中的信息也奏效。***



1. Save your project. 保存项目

*Click* ***File*** *->* ***Save project*** *点击****文件*** *->****保存项目***

**What have you done? 你已经做了什么**

You’ve modified your Scratch character to use machine learning instead of your earlier rules-based approach. 你已经通过机器学习修改脚本，而不是基于规则命令

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. 训练机器自己识别信息回比你一条一条输入可能信息快得多

The more examples you give it, the better it should get at recognising messages correctly. 样本数据越多，电脑的识别能力越好

|  |
| --- |
|  |
| **Ideas and Extensions延申** |
| Now that you’ve finished, why not give one of these ideas a try? 尝试下面的内容吧  Or come up with one of your own? 或者你有什么想法  **Write a reply 写回复**  Instead of just changing the way they look, make your character reply, based on what it recognises in the message! 除了表情回复，让你的人物用文字回复你  **Try a different character 试试不同的人物**  Instead of a person’s face, why not try something different, like an animal? 除了人脸，为什么不试试画个动物  It could react in different ways, instead of smiling. 除了微笑还可以试试其他回复方式  For example, you could make a dog that wags their tail if you say something kind to it! 比如说好话，让狗摇起尾巴  **Different emotions 试试不同表情**  Instead of kind and mean, could you train the character to recognise other types of message? 除了开心不开心，可以训练电脑识别别的信息  **Real world sentiment analysis 现实世界情绪分析**  Can you think of examples where it’s useful to be able to train a computer to recognise the emotion in writing? 你可以想出一个例子训练计算机识别写作内容的情绪吗？ |