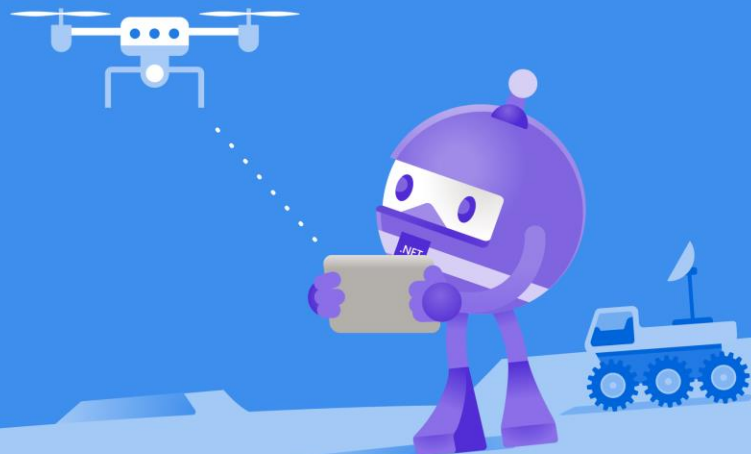


.NET Conf

探索 .NET 新世界



投影片連結：

<https://www.slideshare.net/KoKo164/mlnet-240425254>

ML.NET 在遷移式學習的 應用與挑戰

Ko Ko (柯克)

2020/12/19 .NET Conf @ 台灣大學



自介

本名柯克，也可以叫我科科或是Ko Ko。

現任 Microsoft MVP。

曾經是個自由接案者，目前在電信業，致力於製造業的AI導入。

主要活動在中部社群，也曾在COSCUP、MOPCON等年會演講。

部落格：大魔術熊貓工程師

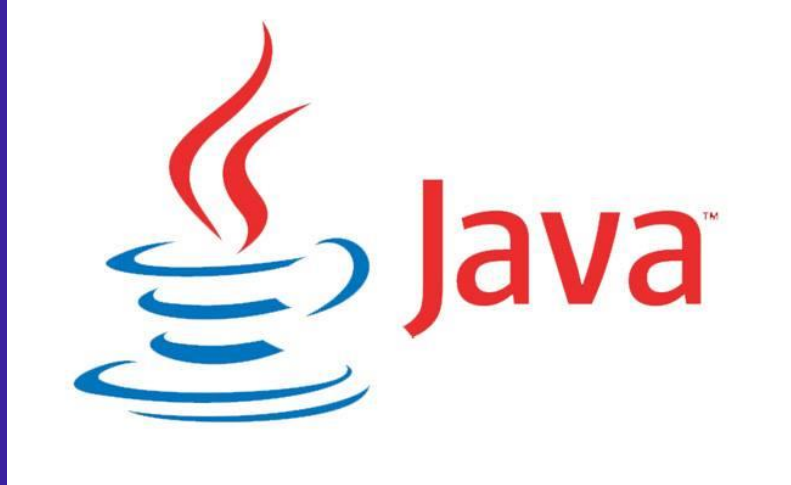
網站：KoKo Mexcelsa演講紀錄



什麼是遷移式學習(Transfer learning)

- 簡單來說就是把已經訓練好的模型，轉移到新的模型上面，以解決不同的問題。
- 用軟體工程師學程式語言來比喻

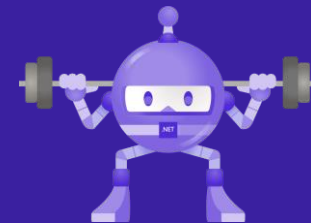






Transfer learning的優勢

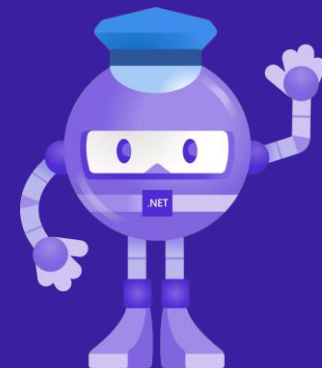
- 使用相對少的資料
- 訓練時間更快，吃的資源更少
- 準確度可以更高





遷移式學習的技巧

- Pre-trained models
- Feature extract
- Fine-tuning

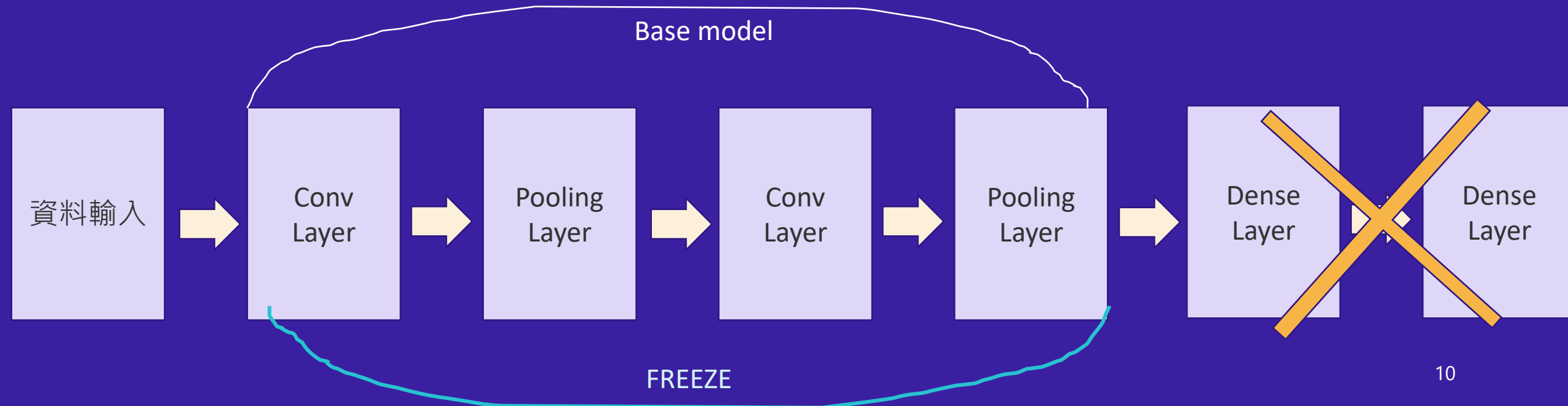


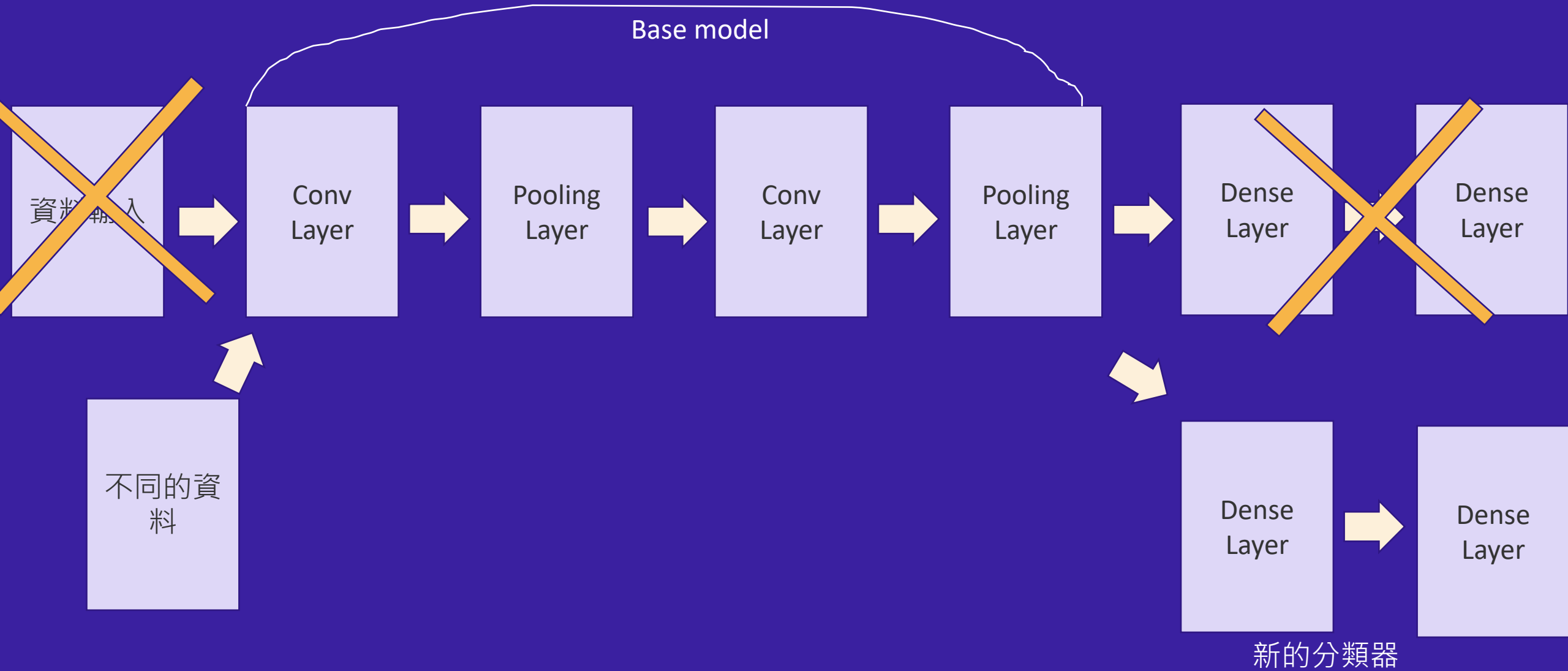
Pre-trained models

- 直接拿 train 好的模型來使用。
- 不更改後面的分類器。
- 該模型在訓練時必須包含你需要預測的東西，否則就沒有意義。

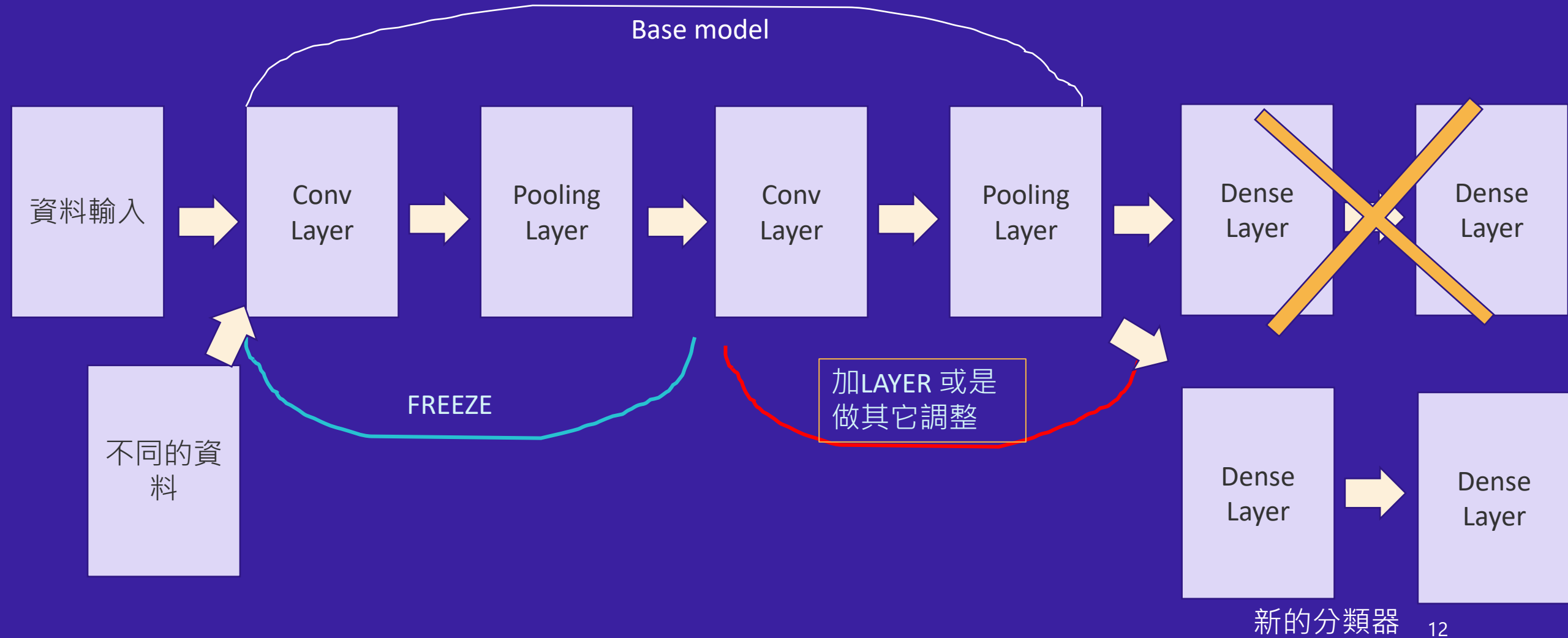
Feature extract

- Base model 是扣掉後面分類器的那幾層。
- 把中間那幾層固定住，一般稱為Freeze，抽換掉前後





Fine-tuning



Style-transfer



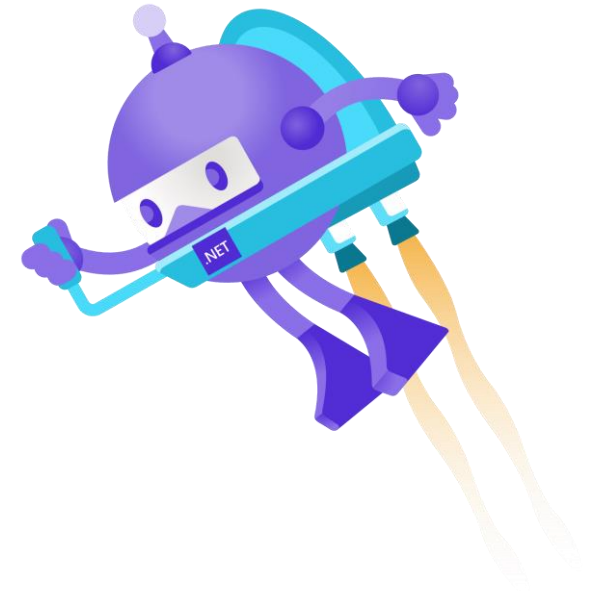
Tensorflow Hub

- Github 放程式碼，Tensorflow Hub放已訓練的AI模型。
- 可以很容易的下載到 pre-trained model 。
- 注意閱讀說明，不是每個模型可以fine-tune 。



Live Demo---Tensorflow Hub

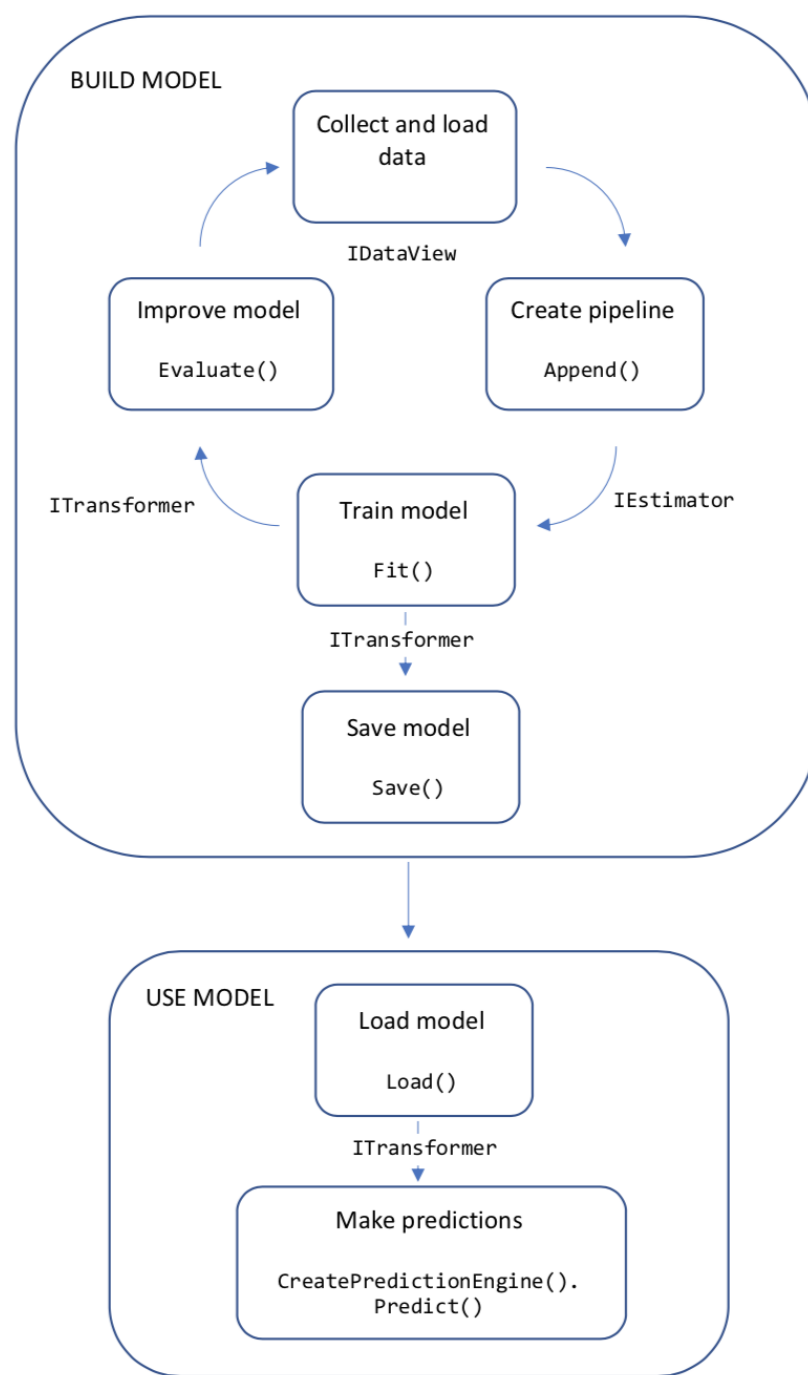
Ko Ko



什麼是ML.NET

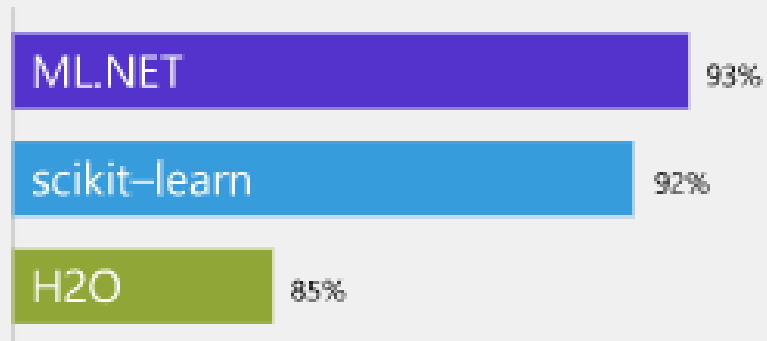
- 微軟一套開源的、跨平台、AI開發框架。
- .NET 開發者可以輕鬆使用C# F# 來做AI 的開發。
- 有很多內建好的算法讓你快速做出成果。



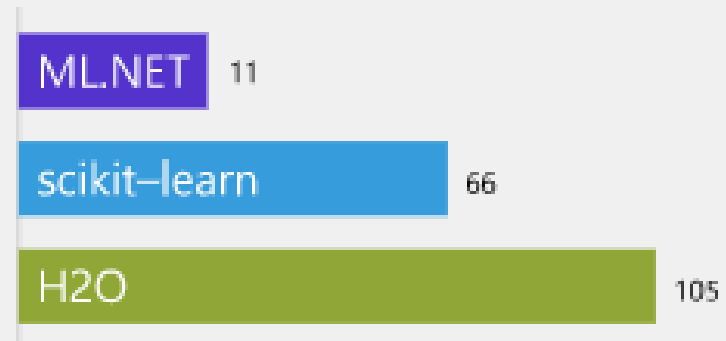


ML.NET的優勢

Accuracy (AUC)



Runtime: Training + Testing (minutes)



Data sourced from [Machine Learning at Microsoft with ML.NET](#) paper. Results for sentiment analysis, using ~900 MB of an Amazon review dataset. Higher accuracy and lower runtime are better.

ML.NET的優勢



Sentiment analysis

Analyze the sentiment of customer reviews using a binary classification algorithm.



Product recommendation

Recommend products based on purchase history using a matrix factorization algorithm.



Price prediction

Predict taxi fares based on parameters such as distance traveled using a regression algorithm.



Customer segmentation

Identify groups of customers with similar profiles using a clustering algorithm.



Object detection

Recognize objects in an image using an ONNX deep learning model.



Fraud detection

Detect fraudulent credit card transactions using a binary classification algorithm.



Sales spike detection

Detect spikes and changes in product sales using an anomaly detection model.



Image classification

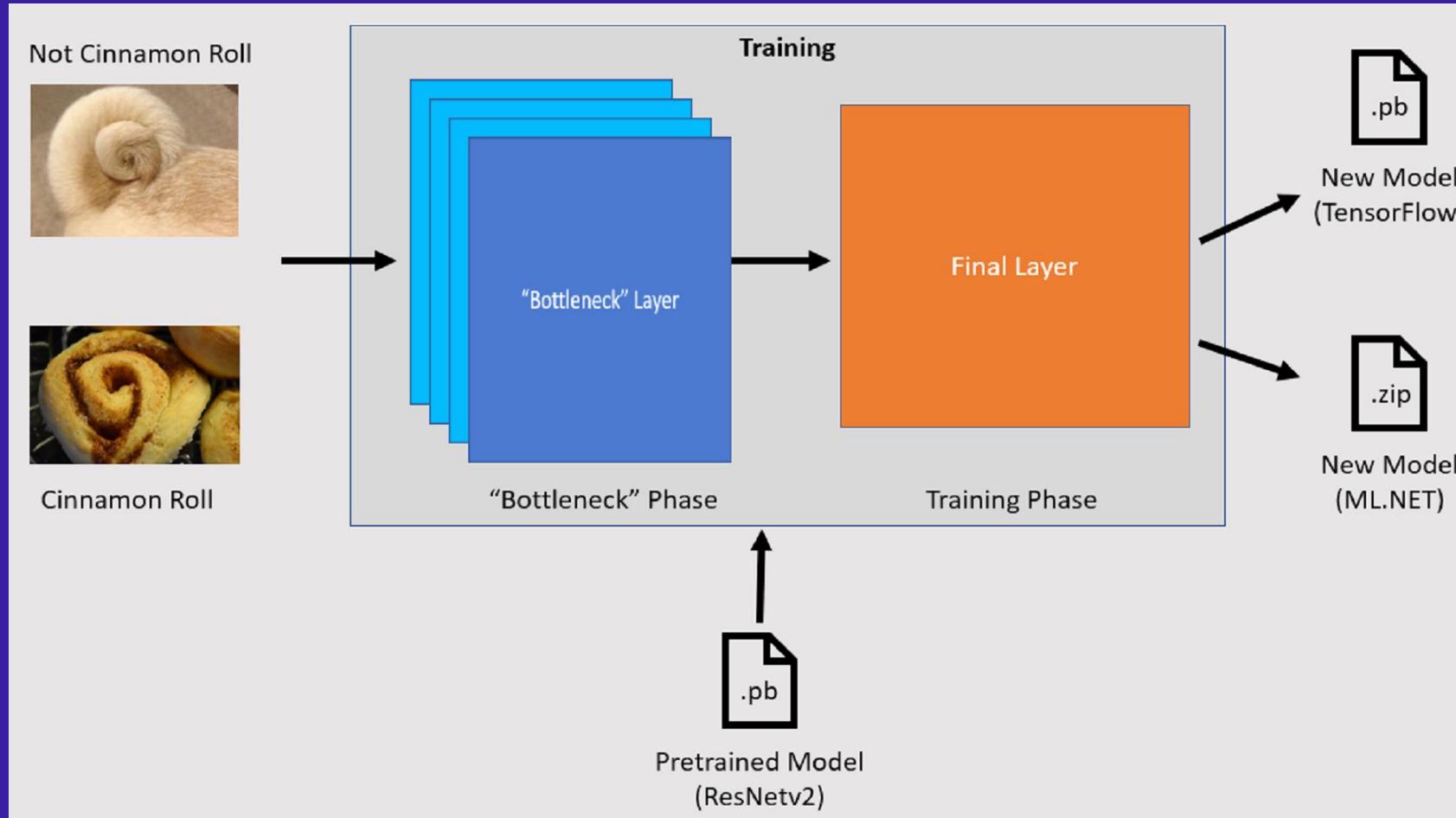
Classify images (for example, broccoli vs. pizza) using a TensorFlow deep learning model.



Sales forecasting

Forecast future sales for products using a regression algorithm.

遷移式學習在ML.NET



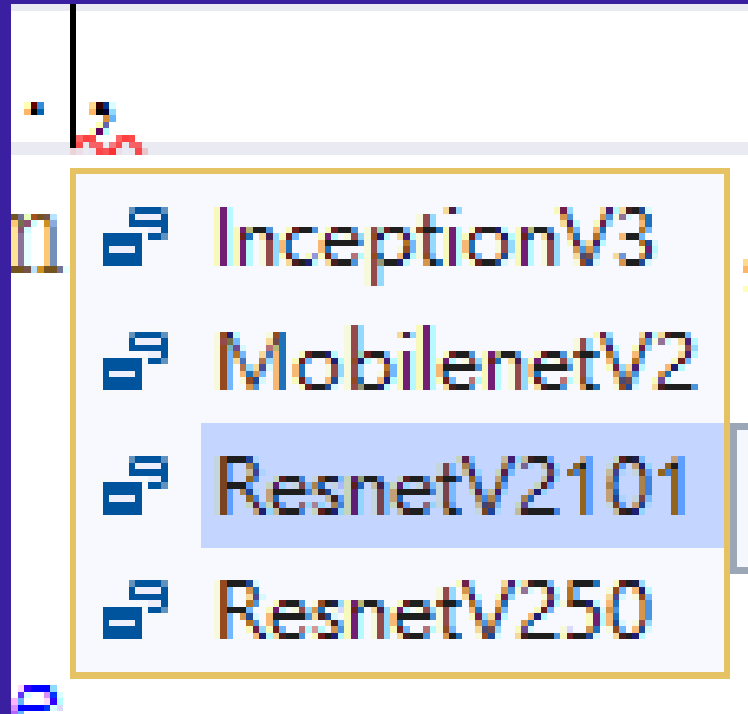
Constructors

`ImageClassificationTrainer.Options()`

Fields

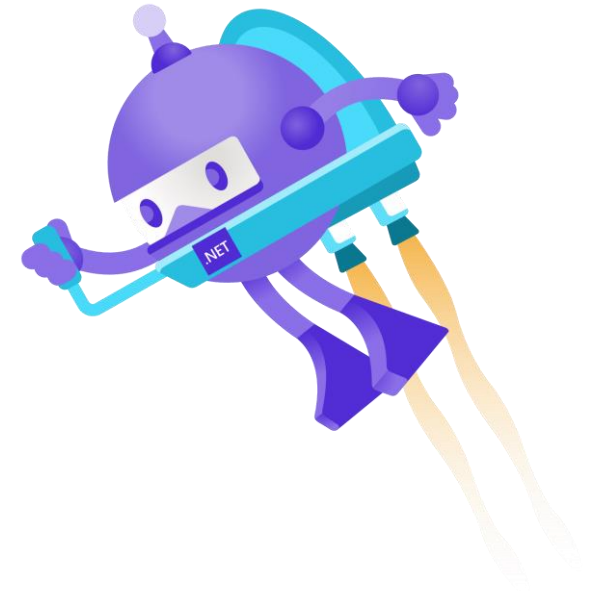
`Arch`

Specifies the model architecture to be used in the case of image classification training using transfer learning. The default Architecture is Resnet_v2_50.



Live Demo---ML.NET 植物病害

Ko Ko



```
C:\Users\koko\Desktop\programming\AIDemo\AIDemo\bin\Debug\netcoreapp3.1\AIDemo.exe
0.40811646
Phase: Training, Dataset used: Validation, Batch Processed Count: 3, Epoch: 8, Accuracy: 0.7916667, Cross-Entropy: 0.4225216
Phase: Training, Dataset used: Validation, Batch Processed Count: 3, Epoch: 9, Accuracy: 0.7916667, Cross-Entropy: 0.43651974
Phase: Training, Dataset used: Validation, Batch Processed Count: 3, Epoch: 10, Accuracy: 0.7916667, Cross-Entropy: 0.4443814
Phase: Training, Dataset used: Validation, Batch Processed Count: 3, Epoch: 11, Accuracy: 0.7916667, Cross-Entropy: 0.45518538
Phase: Training, Dataset used: Validation, Batch Processed Count: 3, Epoch: 12, Accuracy: 0.7916667, Cross-Entropy: 0.46018133
Phase: Training, Dataset used: Validation, Batch Processed Count: 3, Epoch: 13, Accuracy: 0.7916667, Cross-Entropy: 0.4685061
Phase: Training, Dataset used: Validation, Batch Processed Count: 3, Epoch: 14, Accuracy: 0.7583334, Cross-Entropy: 0.47164783
Phase: Training, Dataset used: Validation, Batch Processed Count: 3, Epoch: 15, Accuracy: 0.7583334, Cross-Entropy: 0.47817782
Phase: Training, Dataset used: Validation, Batch Processed Count: 3, Epoch: 16, Accuracy: 0.7583334, Cross-Entropy: 0.48050952
Phase: Training, Dataset used: Validation, Batch Processed Count: 3, Epoch: 17, Accuracy: 0.7583334, Cross-Entropy: 0.48600042
Phase: Training, Dataset used: Validation, Batch Processed Count: 3, Epoch: 18, Accuracy: 0.7583334, Cross-Entropy: 0.48811364
Phase: Training, Dataset used: Validation, Batch Processed Count: 3, Epoch: 19, Accuracy: 0.7583334, Cross-Entropy: 0.49298063
Phase: Training, Dataset used: Validation, Batch Processed Count: 3, Epoch: 20, Accuracy: 0.7583334, Cross-Entropy: 0.4950571
```

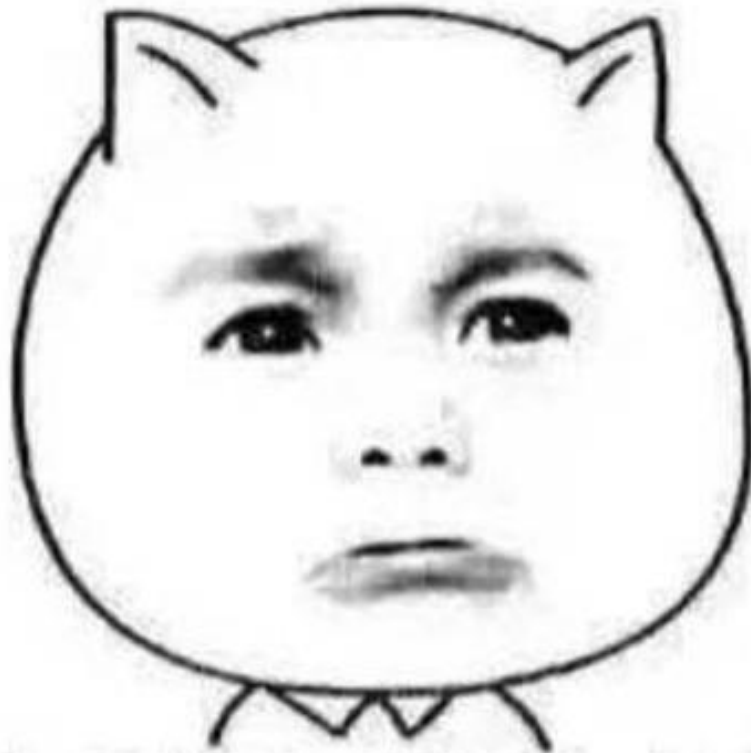
```
Image: IMG_0322.jpg | Actual Value: PowderyMildew | Predicted Value: PowderyMildew
Source=ImageClassificationTrainer; MultiClassClassifierScore; Cursor, Kind=Trace] Channel disposed
Image: IMG_0310.jpg | Actual Value: PowderyMildew | Predicted Value: PowderyMildew
Image: IMG_0297.jpg | Actual Value: PowderyMildew | Predicted Value: PowderyMildew
Image: IMG_0359.jpg | Actual Value: PowderyMildew | Predicted Value: PowderyMildew
Image: IMG_0362.jpg | Actual Value: PowderyMildew | Predicted Value: Health
Image: Powdery Mildew Control.jpg | Actual Value: PowderyMildew | Predicted Value: PowderyMildew
```

```
C:\Users\koko\Desktop\programming\AIDemo\AIDemo\bin\Debug\netcoreapp3.1\AIDemo.exe (process 3548) exited with code 0.
Press any key to close this window . . .
```


看看Tensorflow 怎麼做

Layer (type)	Output Shape	Param #
mobilenetv2_1.00_224 (Model)	(None, 7, 7, 1280)	2257984
conv2d_2 (Conv2D)	(None, 5, 5, 32)	368672
global_average_pooling2d (G1	(None, 32)	0
dense_2 (Dense)	(None, 5)	165

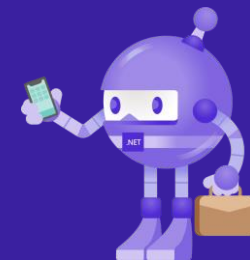
=====
Total params: 2,626,821
Trainable params: 2,231,429
Non-trainable params: 395,392
=====



宝宝委屈 宝宝心里苦 宝宝不说

所以什麼時候用ML.NET

- 把寫TF的同事訓練好的模型，整合到 .NET 的 app 時
- 需要快速的概念驗證
- 不懂 AI 但是要做出有 AI 功能的程式



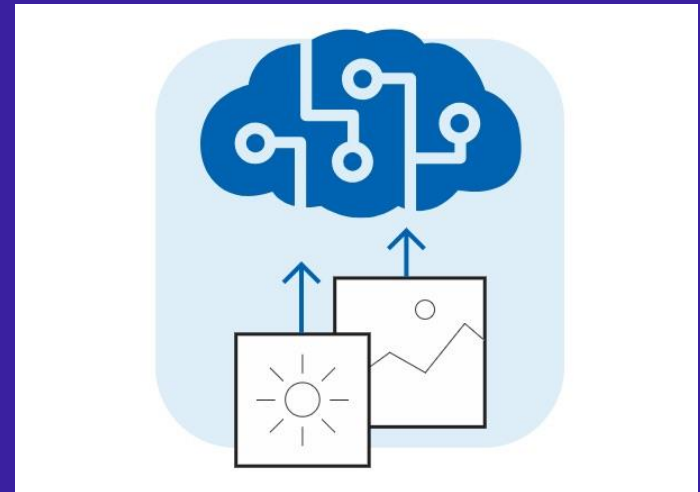
TensorFlow Lite

- 可以做出很小很輕量模型
- 用在手機上或是樹莓派上做 inference
- ML.NET 不支援tflite，不過沒關係，我們有 Xamarin



搭配 Azure Custom vision 使用

- 這是一個Azure 上的服務
- 可以上傳你的影像，很簡單製作自己的AI Model
- 有免費的額度



Project Types ⓘ

- ☒ Classification
- ☐ Object Detection

Classification Types ⓘ

- ☐ Multilabel (Multiple tags per image)
- ☒ Multiclass (Single tag per image)

Domains:

- ☒ General
- ☐ Food
- ☐ Landmarks
- ☐ Retail
- ☐ General (compact)
- ☐ Food (compact)
- ☐ Landmarks (compact)
- ☐ Retail (compact)

Pick the domain closest to your scenario. Compact domains are lightweight models that can be exported to iOS/Android and other platforms. [Learn More](#)



azureshow20200920k...

Training Images Performance Predictions Train

Filter

Iteration

Workspace


Tags

Tagged Untagged

Showing: all tagged images

Search For Tags:

Add images Delete Tag images Select all



Looks like you don't have any images here!

Go ahead and browse for images to upload to your project, tag them, and they will be ready to use.

Add images

JPG, .PNG, .BMP format, up to 6 MB per image

Choose your platform



TensorFlow

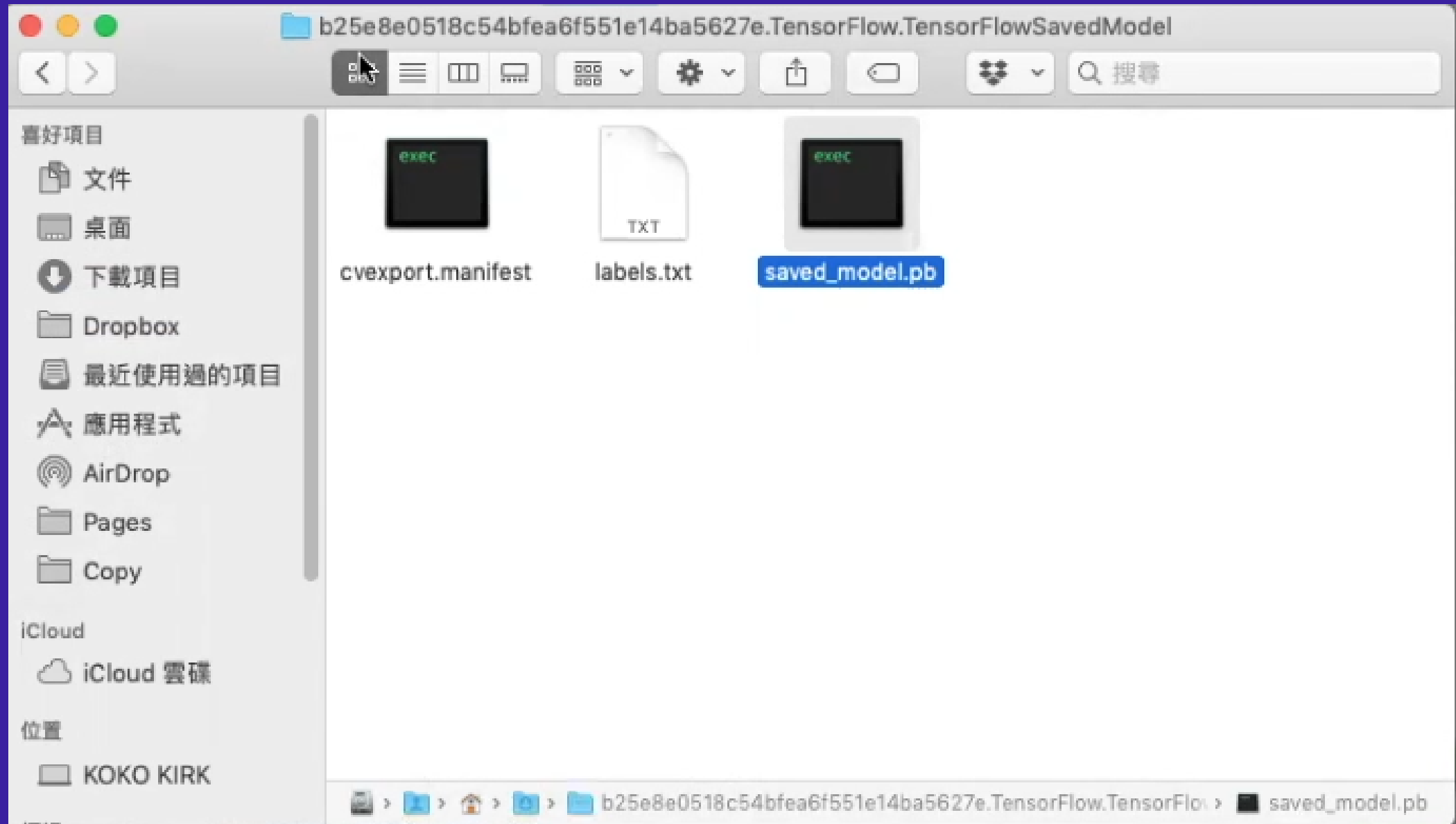
SavedModel



Download

How to use?

[TensorFlow documentation](#)



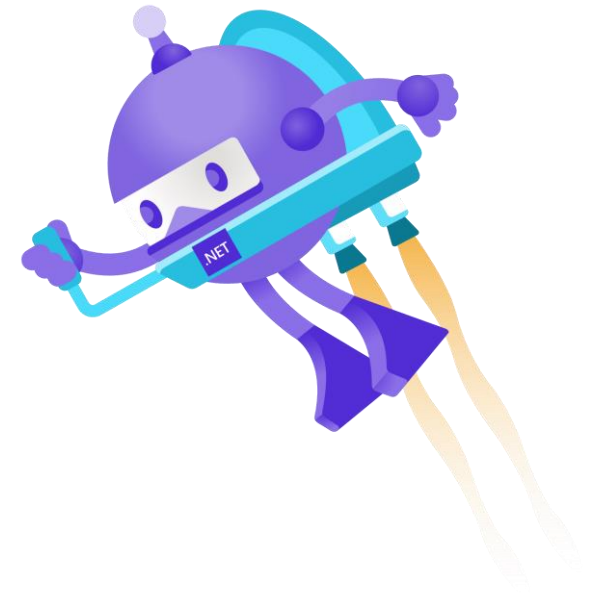
新一代AI開發工具---Model Builder

- 這是一個自動化幫你產生AI 模型的工具
- 主要是使用 AutoML 的技術來幫你自動化產生
- 不用寫一行 code 就幫你產生Model，自動幫你產生code
- 免費！！



Live Demo---Model Builder在 植物病害

Ko Ko



Scenario

Environment

Data

Train

Evaluate

Consume

Next steps

Evaluate

Results of training for your model can be found below.
[How do I understand my model performance?](#)

Best model:

Accuracy: 71.44%

Model: DNN + ResNet50

Try your model



Results

無白粉病	94%
有白粉病	6%

Scenario

Environment

Data

Train

Evaluate

Consume

Next steps

Evaluate

Results of training for your model can be found below.

[How do I understand my model performance?](#)

Best model:

Accuracy: 71.44%

Model: DNN + ResNet50

Try your model



Results

無白粉病	100%
有白粉病	< 1%

ML.NET Model Builder

Program.cs

Scenario

Environment

Data

Train

Evaluate

Consume

Next steps

Consume the model

Let's add the model and necessary projects and references to the solution. Once added, open AIDemoML.ConsoleApp to see how to consume your model. You can then copy the code from this generated console app to the app where you want to consume your model and make predictions.

Add to solution

Next step

Solution Explorer

Search Solution Explorer (Ctrl+;)

Solution 'AIDemo' (3 of 3 projects)

AIDemo

Dependencies

Program.cs

AIDemoML.ConsoleApp

Dependencies

ModelBuilder.cs

Program.cs

AIDemoML.Model

Dependencies

ConsumeModel.cs

MLModel.zip


ModelInput.cs

ModelOutput.cs

Solution Explorer

Team Explorer

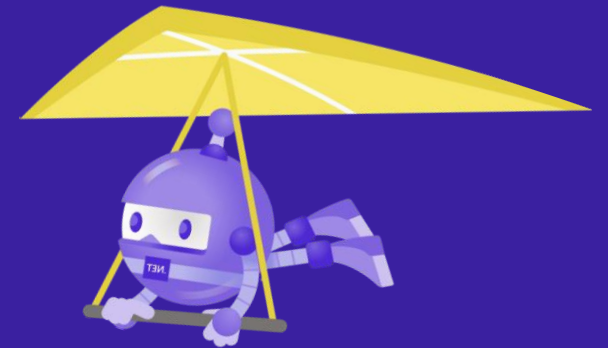
Properties

 Feedback

38

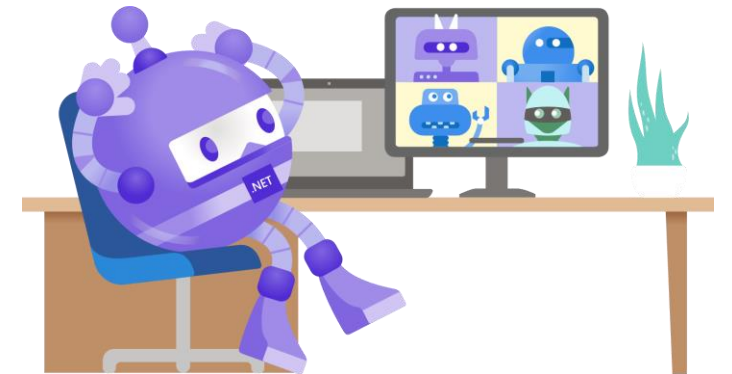
關於做AI的其他心得分享

- 溝通！溝通！溝通！
- 資料！資料！資料！
- 過早要求成功等於立刻失敗
- 專案初期建議用地端來train model 比較省錢
- 後期再用雲端開超大的機器快速訓練模型



重點回顧

- Strategy of transfer learning
- Tensorflow hub
- Advantage of TF lite
- Advantage of ML.NET
- Custom Vision
- Model builder



Thanks for joining!

Ask questions on Twitter using #dotNETConf



.NET Conf
2020

特別感謝

91APP
Technical Network



KKKTIX



HackMD



STUDY4
為 學 習 而 生

以及各位參與活動的你們

