

coreML Day2 预习资料

昨天我们已经体验了一把 coreML 的基本使用，那么今天大家可以来具体的看一下 ML 到底是什么？这些观念掌握了之后，对于今天的课程会比较有帮助的。

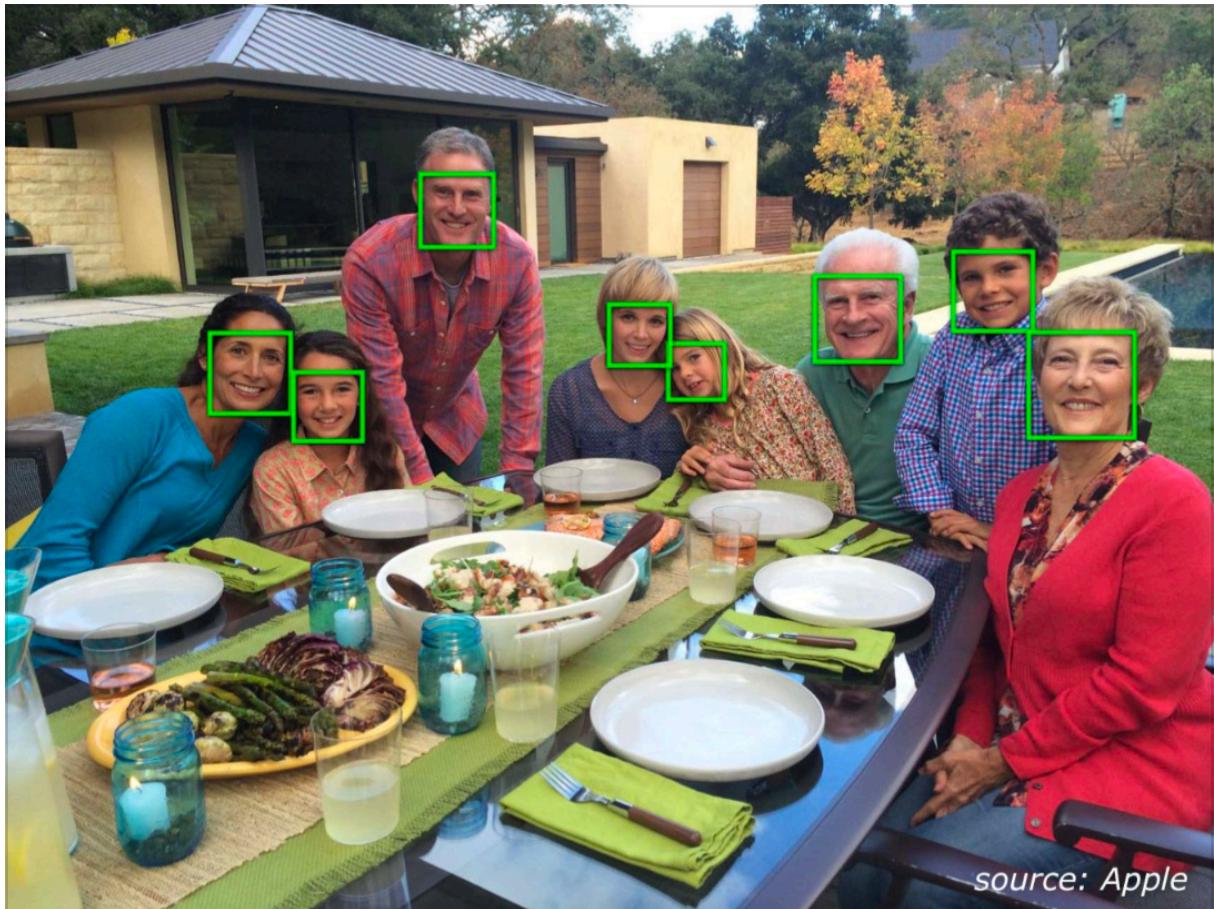
什么是 Machine Learning?

机器学习是人工智能的一个分支，他的目的在于吸收任何的数据比如说(图像，文本，语音，统计数据)然后作出预测数据当中所隐藏的特征或是行为。

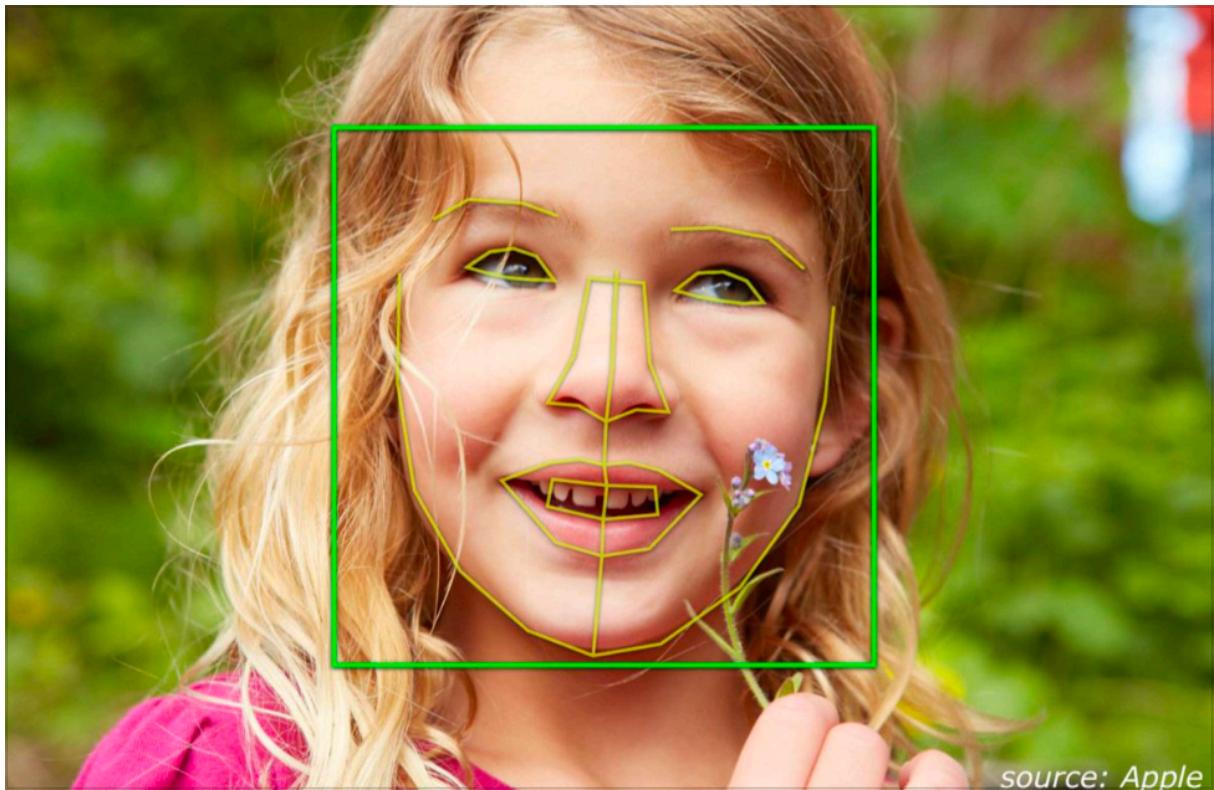
ML 也是被应用在我们的智能设备里面那些隐藏不容易被查间的地方

我们来看看 ML 现在都可以做些什么？

面部识别

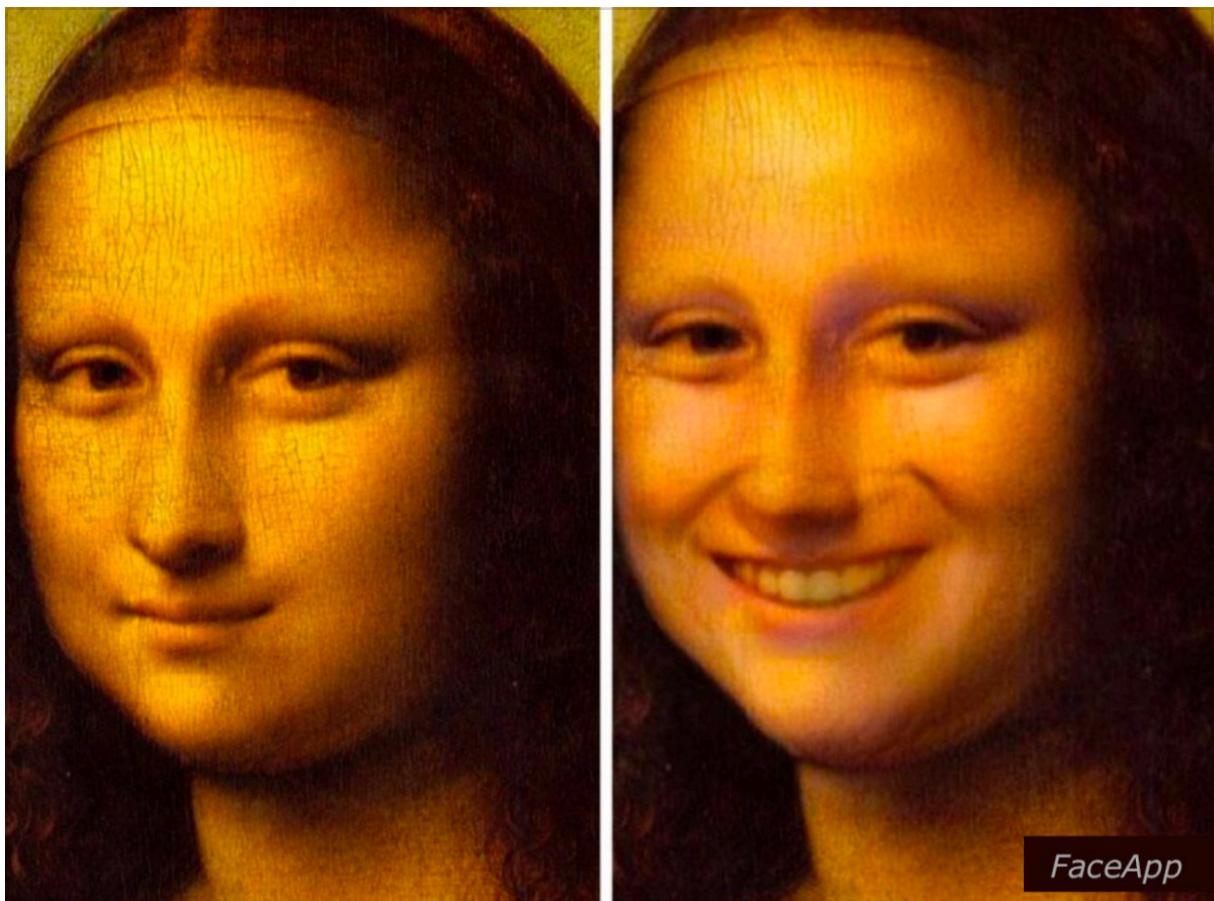


面部特征识别



source: Apple

特征检测和替换



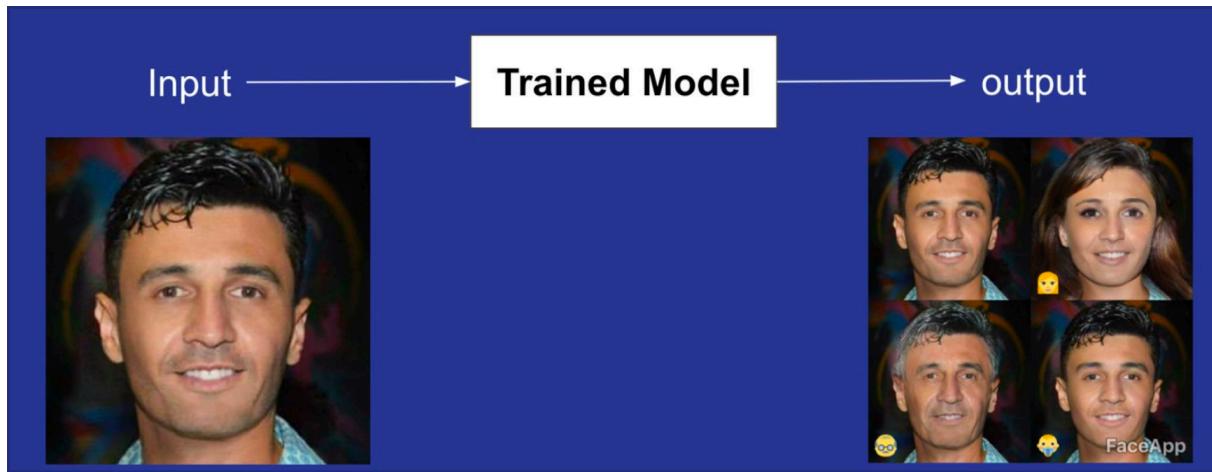
FaceApp

那麼機器學習到底是怎麼學習的？



我们使用海量的图片来训练他什么是面部特征，最终被出来的结果就叫做『训练好的模型』

训练好的模型有什么用？



有什么用～大家自己看吧～发挥一下想像力

為什麼 ML 這麼重要？

ML 可以做到我们人类做的事情，他给我们的产品增加了人性的感觉，使得我们的产品更佳的聪明

这边要注意几个地方：

ML 不是刚出来的～（虽然在 iOS. 上的使用是新的啦）

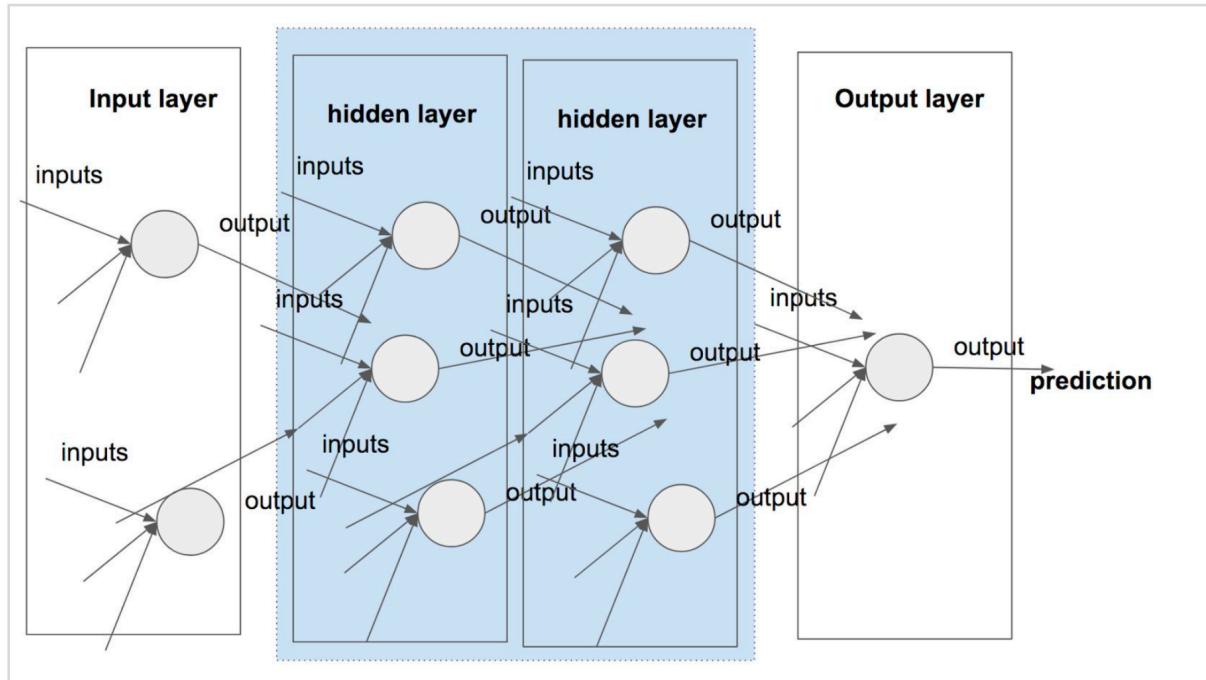
ML 的可用领域越来越多(感谢大数据)

所有的产品都有某种 ML 存在

人工神经网路 (ANN)

人工神經網路 Artificial Neuron Network 是基於生物的神經節狗和功能的一種計算模式，這邊

要注意她不是一種網路，只是一種計算模式而已啊。



上课的时候会详细说明这个图

积卷神经网路(CNN)

在机器学习中，卷积神经网路 Convolutional Neural Network 是一种 反馈形式的人工神迹网路。

目前被广泛的应用在图像识别还有分类，打开你的手机相册就知道了。现今尖端的应用在于机器人还有自驾车的视觉驱动，CNN 也被成功的识别面部，物件，还有交通号志(吐槽一下飞起的塑料袋)

递归神经网路 (RNN)

递归神经网路 Recurrent Neural Network 他是一种深层次的学习模式，他跟 CNN 有点不同，他会在许多的神经网路的隐藏层之间反覆的游走，我们甚至可以说 他是有记忆的。

人们在不同的问题可以用 RNN. 来解决，比方说 时间序列分析，自然语言处理，语音识别。现今最流行的 RNN. 应用是 LSTM.

long short-term memory model 长时间记忆模型，可以用 python 打造。

coreML 用来处理图像

今天主要的部分在於使用 Vision 還有執行 VNRequest

```

let path = Bundle.main.path(forResource: "puppy", ofType: "jpg")
let imageURL = NSURL.fileURL(withPath: path!)

let modelFile = Inceptionv3() // Resnet50() //GoogLeNetPlaces()
let model = try! VNCoreMLModel(for: modelFile.model)
let handler = VNImageRequestHandler(url: imageURL)
let request = VNCoreMLRequest(model: model, completionHandler: myResultsMethod)

try! handler.perform( [ request] )

```

然後完成了處理程序之後，我們拿出 VNRequest 的結果

```

func myResultsMethod(request: VNRequest, error: Error?) {
    guard let results = request.results as? [VNClassificationObservation] else {
        fatalError("could not get results from ML Vision request")
    }

    var bestPrediction = ""
    var bestConfidence: VNConfidence = 0

    for classification in results {
        if(classification.confidence > bestConfidence) {
            bestConfidence = classification.confidence
            bestPrediction = classification.identifier
        }
    }
    print("predicted: \(bestPrediction) with confidence of \(bestConfidence) out of 1")

    self.resultLabel.text = bestPrediction
}

```

想要直接把 UIImage 直接丢进 coreML 里面吗？请参考 jh/6
像素缓冲区里面拿到图片的方法。(上课的时候讲)

我们有很多不同的让她吃到图片的操作

```

VNImageRequestHandler (cgImage: CGImage, options: [VNImageOption : Any])
VNImageRequestHandler (cgImage: CGImage, orientation: Int32, options: [VNImageOption : Any])
VNImageRequestHandler (ciImage: CIImage, options: [VNImageOption : Any])
VNImageRequestHandler (ciImage: CIImage, orientation: Int32, options: [VNImageOption : Any])
VNImageRequestHandler (cvPixelBuffer: CVPixelBuffer, options: [VNImageOption : Any])
VNImageRequestHandler (cvPixelBuffer: CVPixelBuffer, orientation: Int32, options: [VNImageOption : Any])
VNImageRequestHandler (data: Data, options: [VNImageOption : Any])
VNImageRequestHandler (data: Data, orientation: Int32, options: [VNImageOption : Any])

```