Core ML in Depth

Krishna Sridhar, Core ML
Zach Nation, Core ML
The easiest way to integrate machine learning models into your app.
The easiest way to integrate machine learning models into your app.
The easiest way to integrate machine learning models into your app.

macOS      iOS
The easiest way to integrate machine learning models into your app.
The easiest way to integrate machine learning models into your app.
Think of models as code
Development Flow
Development Flow

Xcode

MLMODEL
Development Flow

Xcode

MLMODEL ➔ SWIFT
Development Flow

Xcode

MLMODEL → SWIFT → Your App
Development Flow

Xcode

MLMODEL

SWIFT

Your App

exec

Your App
let model = FlowerClassifier()

if let prediction = try? model.prediction(image: image) {
    return prediction.flowerType
}
let model = FlowerClassifier()

if let prediction = try? model.prediction(image: image) {
    return prediction.flowerType
}
This Session

Use Cases
This Session

Use Cases

Hardware Optimized

<table>
<thead>
<tr>
<th>Core ML</th>
<th>Accelerate</th>
<th>MPS</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>CPU</td>
<td>GPU</td>
</tr>
</tbody>
</table>
This Session

Use Cases

Hardware Optimized

Obtaining Models

Core ML
- Accelerate
- MPS
  - CPU
  - GPU
This Session

Use Cases

Core ML is awesome. I love using it.

Hardware Optimized

<table>
<thead>
<tr>
<th>Core ML</th>
<th>Accelerate</th>
<th>MPS</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>CPU</td>
<td>GPU</td>
</tr>
</tbody>
</table>

Obtaining Models

MLMODEL
Core ML is awesome. I love using it.
Sentiment Analysis  
Object Detection  

Personalization  
Style Transfer  

Gesture Recognition  
Summarization  

Music Tagging
Pipeline

Vectorize → Neural Network → Map to String → Rose
Focus on code, not models!
Models as Functions

Numeric
Categories
Images
Arrays
Dictionaries

MLMODEL → “Rose”
Numeric and Categories

developer.apple.com/machine-learning

Numeric
- Double, Int64

Categories
- String, Int64

Images
- CVPixelBuffer

Arrays
- MLMultiArray

Dictionaries
- [String : Double], [Int64 : Double]
# Images

<table>
<thead>
<tr>
<th>Category</th>
<th>Types</th>
</tr>
</thead>
<tbody>
<tr>
<td>Numeric</td>
<td>Double, Int64</td>
</tr>
<tr>
<td>Categories</td>
<td>String, Int64</td>
</tr>
<tr>
<td>Images</td>
<td>CVPixelBuffer</td>
</tr>
<tr>
<td>Arrays</td>
<td>MLMultiArray</td>
</tr>
<tr>
<td>Dictionaries</td>
<td>[String : Double], [Int64 : Double]</td>
</tr>
</tbody>
</table>
## Multi-Dimensional Arrays

<table>
<thead>
<tr>
<th>Category</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Numeric</td>
<td>Double, Int64</td>
</tr>
<tr>
<td>Categories</td>
<td>String, Int64</td>
</tr>
<tr>
<td>Images</td>
<td>CVPixelBuffer</td>
</tr>
<tr>
<td>Arrays</td>
<td>MLMultiArray</td>
</tr>
<tr>
<td>Dictionaries</td>
<td>[String : Double], [Int64 : Double]</td>
</tr>
</tbody>
</table>
Dictionaries

- **Numeric**: Double, Int64
- **Categories**: String, Int64
- **Images**: CVPixelBuffer
- **Arrays**: MLMultiArray
- **Dictionaries**: `[String : Double], [Int64 : Double]`
Working with Text
Sentiment Analysis
CoreML is awesome. I love using it.
Sentiment Analysis

I

clear

the

this

Q W E R T Y U I O P

A S D F G H J K L

Z X C V B N M

1 2 3

space

return
Today’s lunch was terrible and disappointing.
Sentiment Analysis Application

Core ML is awesome. I love using it.
Core ML is awesome. I love using it.
Sentiment Analysis Application

Core ML is awesome.
I love using it.
Sentiment Analysis Application

Core ML is awesome. I love using it.
Using Other Formats

Word Counts

```json
{
    "Core" : 1,
    "ML" : 1,
    "is" : 1,
    "awesome" : 1,
    "I" : 1,
    "love" : 1,
    "using" : 1,
    "it" : 1,
}
```
Using Other Formats

Word Counts

```json
{
    "Core": 1,
    "ML": 1,
    "is": 1,
    "awesome": 1,
    "I": 1,
    "love": 1,
    "using": 1,
    "it": 1,
}
```
Using Other Formats

Word Counts

```json
{
    "Core" : 1,
    "ML" : 1,
    "is" : 1,
    "awesome" : 1,
    "I" : 1,
    "love" : 1,
    "using" : 1,
    "it" : 1,
}
```
Use NLP to Process Text

Word Counts

```json
{
  "Core" : 1,
  "ML" : 1,
  "is" : 1,
  "awesome" : 1,
  "I" : 1,
  "love" : 1,
  "using" : 1,
  "it" : 1,
}
```
Sentiment Analysis Application

Use NLP (NSLinguisticTagger)

Use Core ML

Core ML is awesome. I love using it.
Core ML is awesome. I love using it.
Processing Text

```swift
func tokenizeAndCountWords(rawTxt: String) -> [String : Double] {
    // Tokenize using NSLinguisticTagger
    ...

    // Count occurrences of each token
}
```
let model = SentimentAnalysis()
if let prediction = try? model.prediction(wordCounts: counts) {
    print("Sentiment: \(prediction.sentiment)"
}
Text Gotchas

Most models don’t work directly on raw text.
What do you think about having dinner out at Point Reyes tomorrow?

I'm not sure if Oliver will eat oysters, but he will.
I’m not sure Oliver will eat oysters, but he will love...
I’m not sure Oliver will eat oysters, but he will.

Task - Next word prediction
Shall I compare
Language Models

Shall I compare

What do you think about having dinner out at Point Reyes tomorrow?

I'm not sure if Oliver will eat oysters, but he will.
Shall I compare thee to a summer's day?
Shakespeare Keyboard

Shall I compare

Current Word

thee | summers | day
Shakespeare Keyboard

Current Word

Next Word Choices

State

MLMODEL

Thee | Summers | Day

Shall I compare
Shakespeare Keyboard

Current Word → Next Word Choices → State

Next Word → Current Word

thee | summers | day
Shakespeare Keyboard

Current Word → Next Word

Next Word Choices → State

MLMODEL

thee
summers
day
Shakespeare Keyboard

Current Word

Next Word

MLMODEL

State

MLMODEL

State

Next Word Choices

Shall I compare

thee    summers    day
if let output = try? model.prediction(input: input) {
    // Send the best 3 words to the user
    displayTopPredictions(output.nextWordProb)
}

// Update state for next prediction
input.state = output.state
input.currentWord = getNextWordFromUser()
if let output = try? model.prediction(input: input) {
    // Send the best 3 words to the user
    displayTopPredictions(output.nextWordProb)
}

// Update state for next prediction
input.state = output.state
input.currentWord = getNextWordFromUser()
if let output = try? model.prediction(input: input) {
    // Send the best 3 words to the user
    displayTopPredictions(output.nextWordProb)
}

// Update state for next prediction
input.state = output.state
input.currentWord = getNextWordFromUser()
if let output = try? model.prediction(input: input) {

    // Send the best 3 words to the user
    displayTopPredictions(output.nextWordProb)
}

// Update state for next prediction
input.state = output.state
input.currentWord = getNextWordFromUser()}
Come to the Labs

Thursday 11:00–3:30
Friday 1:30–4:00
dining table
35.8 %
Built on Performance Primitives

Your app

Vision
NLP

Core ML

ML Performance Primitives

Accelerate
MPS

Hardware

CPU
GPU
Optimized for Hardware

Runs on GPU

vase

rose

MPS
Optimized for Hardware

- Runs on GPU
  - MPS
- Runs on CPU
  - Accelerate
Optimized for Hardware
Optimized for Hardware

Runs on Core ML
Context Switching

Runs on GPU

Runs on CPU

Dog playing in a soccer field
This Session

Use Cases

Hardware Optimized

<table>
<thead>
<tr>
<th>Core ML</th>
<th>Accelerate</th>
<th>MPS</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPU</td>
<td>GPU</td>
<td></td>
</tr>
</tbody>
</table>

Obtaining Models
Deploying Core ML Models

- Xcode
- SWIFT
- Your App
Deploying Core ML Models

Where do models come from?
Example Models

Task specific Core ML models

Places205-GoogLeNet
Detects the scene of an image from 205 categories such as an airport terminal, bedroom, forest, coast, and more.
View original model details ›
Download Core ML Model
File size: 24.8 MB

ResNet50
Detects the dominant objects present in an image from a set of 1000 categories such as trees, animals, food, vehicles, people, and more.
View original model details ›
Download Core ML Model
File size: 102.6 MB

developer.apple.com/machine-learning
Tap into ML Community

Popular ML libraries

Many models

Thriving communities

Caffe

Keras

XGBoost

LIBSVM
Tap into ML Community

Popular ML libraries

Many models

Thriving communities
Core ML Tools
Core ML Tools

Open Source
Conversion Workflow

Xcode

MLMODEL → SWIFT → Your App

exec

ML

Swift

Your App

ML

ML

Swift

Your App
Conversion Workflow

Model Source

Xcode

Your App

e.g. Caffe

MLMODEL

SWIFT

exec

Your App
Conversion Workflow

Model Source

e.g. Caffe

Xcode

MLMODEL

SWIFT

Your App

exec
Getting Started

### Download and install python package

> pip install coremltools
What Is coremltools?

Converters

- Core ML Bindings
- Converter Library

Core ML Specification

Convert from other formats
What Is coremltools?

Converters

Core ML Bindings  Converter Library

Core ML Specification

Convert from other formats
Build your own converter
Compatible and Extensible

Compatible

Converters

Core ML Bindings

Converter Library

Core ML Specification

Caffe Keras dmlc XGBoost LIBSVM learn
Core ML Model
developer.apple.com/machine-learning

Single document
Encapsulates
- Functional description (inputs → outputs)
- Trained parameters

Public format
Xcode Model View

**Machine Learning Model**
- **Name**: FlowerClassifier
- **Type**: Neural Network Classifier
- **Size**: 41.6 MB
- **Author**: Lizzi Ottens
- **License**: MIT
- **Description**: Identify the type of flower present in an image.

**Model Class**
- FlowerClassifier (Swift generated source)

**Model Evaluation Parameters**

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>flowerimage</td>
<td>Image&lt;RGB,227,227&gt;</td>
<td>Input image of a flower</td>
</tr>
<tr>
<td>flowerType</td>
<td>String</td>
<td>Most likely flower type in image</td>
</tr>
<tr>
<td>flowerTypeProbs</td>
<td>Dictionary&lt;String,Double&gt;</td>
<td>Probability of each flower type</td>
</tr>
</tbody>
</table>
Xcode Model View

**Machine Learning Model**

- **Name**: FlowerClassifier
- **Type**: Neural Network Classifier
- **Size**: 41.6 MB
- **Author**: Lizi Ottens
- **License**: MIT
- **Description**: Identify the type of flower present in an image.

**Model Class**

- FlowerClassifier (Swift generated source)

**Model Evaluation Parameters**

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>flowerimage</td>
<td>Image&lt;RGB,227,227&gt;</td>
<td>Input image of a flower</td>
</tr>
<tr>
<td>flowerType</td>
<td>String</td>
<td>Most likely flower type in image</td>
</tr>
<tr>
<td>flowerTypeProbs</td>
<td>Dictionary&lt;String,Double&gt;</td>
<td>Probability of each flower type</td>
</tr>
</tbody>
</table>
Core ML Converters

Unified APIs to convert models from various formats to Core ML

Model Source → Convert → In Xcode

e.g. Caffe → MLMODEL

Converters
- Core ML Bindings
- Converter Library
- Core ML Specification
Flower Classification in Caffe
Flower Classification in Caffe

Caffe

.caffemodel → .protocxt → labels.txt

74 → "Rose"
Flower Classification in Caffe

Caffe

-needed for conversion to Core ML format
Demo
import coremltools

caffe_model = ('flowers.caffemodel', 'flowers.prototxt')
model = coremltools.converters.caffe.convert(
    caffe_model,
    image_input_names = 'data',
    class_labels = 'labels.txt')

model.save('FlowerClassifier.mlmodel')
import coremltools

keras_model = 'flowers.h5'
model = coremltools.converters.keras.convert(keras_model,
                                          image_input_names = 'data',
                                          class_labels = 'labels.txt')

model.save('FlowerClassifier.mlmodel')
Supported Packages

- Caffe
- Keras
- Neural Networks

- Pipelines
- Scikit-learn
- Tree Ensembles

- XGBoost
- Linear Models
- Support Vector Machines
- LIBSVM
Obtaining Models

Summary

**ResNet50**
Detects the dominant objects present in an image from a set of 1000 categories such as trees, animals, food, vehicles, people, and more.

[View original model details](#)

- Download Core ML Model
  - File size: 102.6 MB

developer.apple.com/machine-learning
Summary
Summary

Easy integration of ML models
Summary

Easy integration of ML models

Rich datatype support
Summary

Easy integration of ML models
Rich datatype support
Hardware optimized
Summary

Easy integration of ML models

Rich datatype support

Hardware optimized

Compatible with popular formats
More Information

https://developer.apple.com/wwdc17/710
## Related Sessions

<table>
<thead>
<tr>
<th>Session Title</th>
<th>Location</th>
<th>Date/Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>Introducing Core ML</td>
<td></td>
<td>WWDC 2017</td>
</tr>
<tr>
<td>Vision Framework: Building on Core ML</td>
<td></td>
<td>WWDC 2017</td>
</tr>
<tr>
<td>Natural Language Processing and your Apps</td>
<td></td>
<td>WWDC 2017</td>
</tr>
<tr>
<td>Accelerate and Sparse Solvers</td>
<td>Grand Ballroom A</td>
<td>Thursday 10:00AM</td>
</tr>
<tr>
<td>Using Metal 2 for Compute</td>
<td>Grand Ballroom A</td>
<td>Thursday 4:10PM</td>
</tr>
<tr>
<td>Lab Name</td>
<td>Location</td>
<td>Time</td>
</tr>
<tr>
<td>----------------------------------------------</td>
<td>-------------------</td>
<td>-----------------</td>
</tr>
<tr>
<td>Core ML and Natural Language Processing Lab</td>
<td>Technology Lab D</td>
<td>Thu 11:00AM-3:30PM</td>
</tr>
<tr>
<td>Core ML and Natural Language Processing Lab</td>
<td>Technology Lab D</td>
<td>Fri 1:50PM-4:00PM</td>
</tr>
</tbody>
</table>