



Unleashing the Power of Flutter and TensorFlow

Onuralp SEZER - Senior Software Engineer

Tensorflow Lite (tflite)

TensorFlow Lite, Google tarafından geliştirilen ve özellikle cihaz üzerinde çalışmak üzere tasarlanmış açık kaynaklı bir makine öğrenimi kütüphanesidir. Geliştiricilere makine öğrenimi modellerini mobil ve gömülü cihazlara entegre etme imkanı sağlar, böylece daha düşük gecikmeli ve daha verimli işleme imkanı sunar.

The Flutter logo is located on the left side of the slide. It consists of a stylized 'F' made of several overlapping geometric shapes in various shades of blue and white. The background of the slide is a dark blue with abstract light blue shapes, including circles and rectangles, scattered around the edges.

Flutter (UI Framework)

Flutter, Google tarafından geliştirilen ve özellikle mobil uygulama ve kullanıcı arayüzü (UI) tasarımı için kullanılan bir açık kaynaklı UI yazılım geliştirme kitidir. Dart programlama dilini kullanarak, tek bir kod tabanından iOS, Android ve web gibi farklı platformlara yönelik yüksek performanslı ve güzel kullanıcı arayüzleri oluşturmayı sağlar.

flutter-tflite

Flutter-TFLite, Flutter ve Dart kullanarak geliştirilen, TensorFlow Lite modellerini açabilen ve kullanabilen, platformlar arası uyumlu bir kütüphanedir. Bu kütüphane, Android, iOS, Linux, macOS ve Windows gibi çeşitli işletim sistemlerinde, Flutter uygulamalarında TensorFlow Lite modellerini yüklemek, sınıflandırmak, tahminler yapmak ve sonuçları işlemeyi sağlayan kütüphanedir.



flutter-tflite



```
dependencies:  
  tflite_flutter: ^0.10.4
```



flutter-tflite



```
$ flutter pub add tflite_flutter
```



flutter-tflite



```
import 'package:tflite_flutter/tflite_flutter.dart';
```



Load tflite model



```
import 'package:tflite_flutter/tflite_flutter.dart';

Interpreter _interpreter;

Future<void> loadModel() async {
  _interpreter = await Interpreter.fromAsset('assets/your_model.tflite');
}
```



Pre-process Input Data



```
Future<Image> pickImage() async {  
  final image_picker = ImagePicker();  
  final image_file = await image_picker.getImage(source: ImageSource.gallery);  
  
  if (image_file != null) {  
    return Image.file(File(image_file.path), width: 300, height: 300);  
  } else {  
    print('No image selected.');    return null;  
  }  
}
```

Pre-process Input Data



```
Future<Image> pickImage() async {  
  final image_picker = ImagePicker();  
  final image_file = await image_picker.getImage(source: ImageSource.gallery);  
  
  if (image_file != null) {  
    return Image.file(File(image_file.path), width: 300, height: 300);  
  } else {  
    print('No image selected.');    return null;  
  }  
}
```

Run Inference (single input)



```
Future<List> runInference(Image image) async {  
  final input = preprocessInput(image);  
  final output = List<double>.filled(1, 0).reshape([1, 1]);  
  _interpreter.run(input, output);  
  return output;  
}
```

Run Inference (multiple input)



```
var input0 = [3.14];
var input1 = [2.43];

// input: List<Object>
var inputs = [input0, input1, input0, input1];

var output0 = List<double>.filled(1, 0);
var output1 = List<double>.filled(1, 0);

// output: Map<int, Object>
var outputs = {0: output0, 1: output1};

// inference
interpreter.runForMultipleInputs(inputs, outputs);

// print outputs
print(outputs)
```

Run Async Inference



```
import 'package:tflite_flutter/tflite_flutter.dart';

final interpreter = await Interpreter.fromAsset('assets/your_model.tflite');
final isolateInterpreter =
    await IsolateInterpreter.create(address: interpreter.address);

var input0 = [3.14];
var input1 = [2.43];

// input: List<Object>
var inputs = [input0, input1, input0, input1];

var output0 = List<double>.filled(1, 0);
var output1 = List<double>.filled(1, 0);

// output: Map<int, Object>
var outputs = {0: output0, 1: output1};

await isolateInterpreter.run(input, output);
await isolateInterpreter.runForMultipleInputs(inputs, outputs);
```

Close Inference



```
interpreter.close();
```



Demo Inference



```
final demo_interpreter = await tfl.Interpreter.fromAsset('assets/demo.tflite');
```



Teşekkür Ederim



```
Onuralp SEZER - @onuralpsr  
Github: https://github.com/onuralpsr  
Linkedin: https://www.linkedin.com/in/onuralpsr  
  
interpreter.close();
```

