

Onuralp SEZER - Senior Software Engineer



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.

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 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.





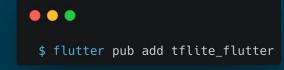


dependencies:
 tflite flutte

tflite_flutter: ^0.10.4















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;
}
```





```
var input1 = [2.43];
var inputs = [input0, input1, input0, input1];
var output0 = List<double>.filled(1, 0);
var output1 = List<double>.filled(1, 0);
var outputs = {0: output0, 1: output1};
interpreter.runForMultipleInputs(inputs, 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 output0 = List<double>.filled(1, 0);
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 - @onuralpszr
Github: https://github.com/onuralpszr
Linkedin: https://www.linkedin.com/in/onuralpszr
interpreter.close();

