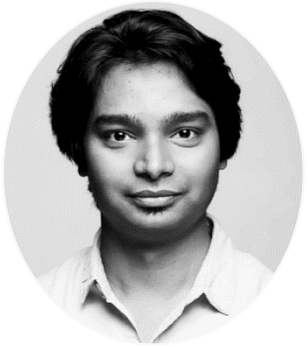
Building Machine Learning Solutions with TensorFlow.js

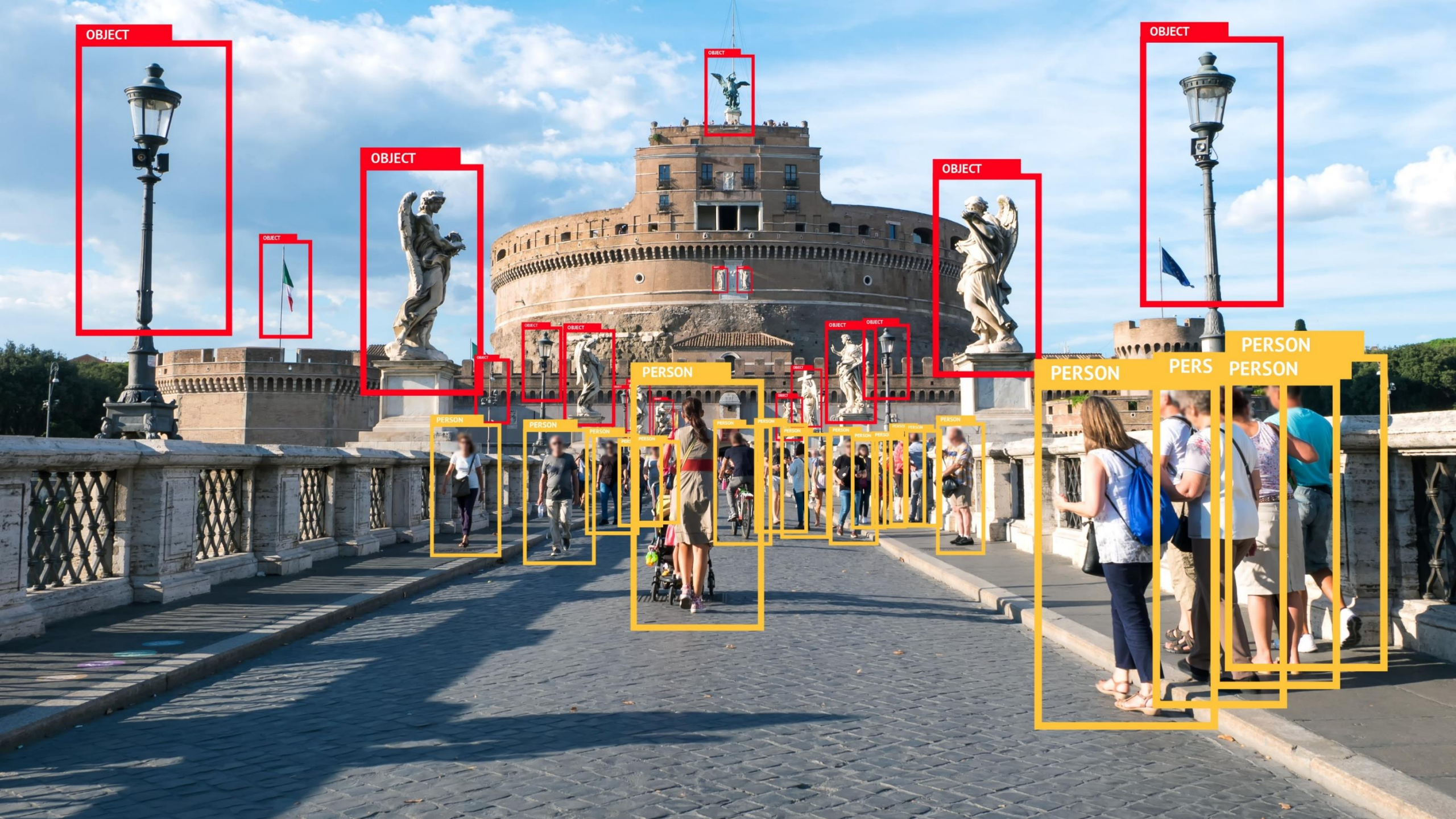
INTRODUCTION

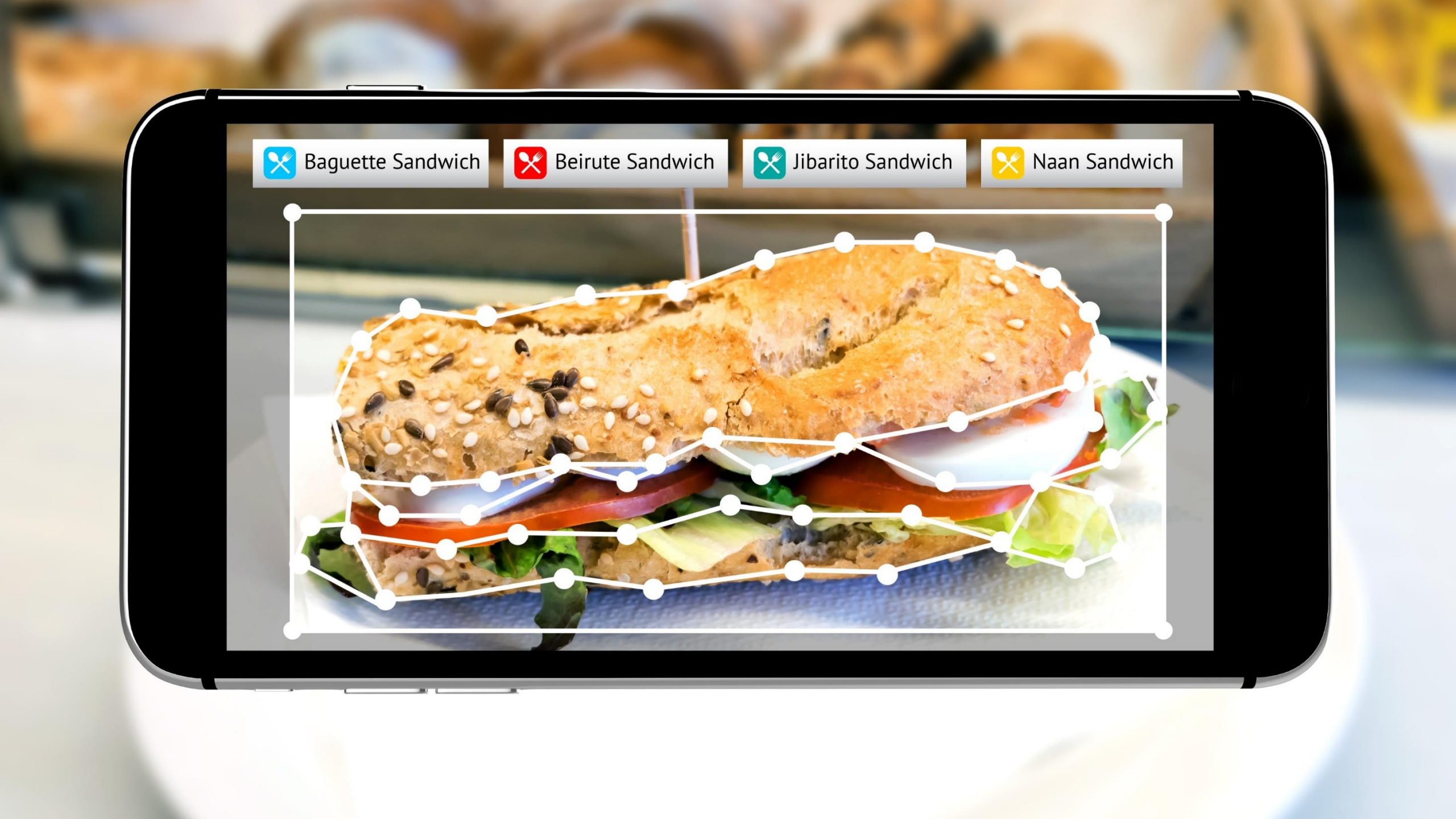


Abhishek Kumar
DATA SCIENTIST | AUTHOR | SPEAKER

@meabhishekkumar


0101 010
1001 0
0010101 0
101 010
01 010 1011 0010
1001 001


OBJECT
OBJECT
OBJECT
OBJECT
OBJECT
PERSON
PERSON
PERSON
PERS
PERSON

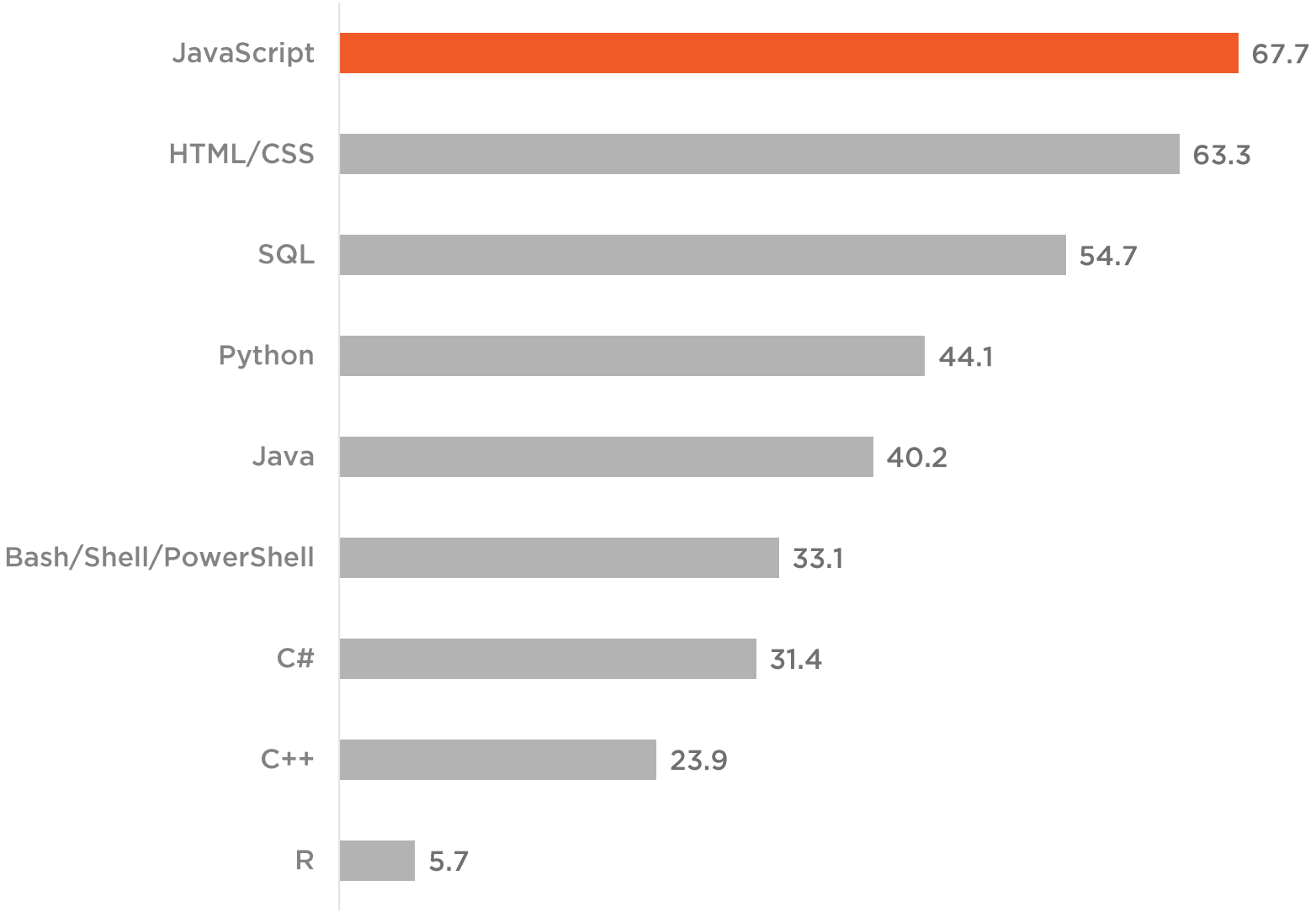

Baguette Sandwich
Beirute Sandwich
Jibarito Sandwich
Naan Sandwich


Hello,

Core Languages

Python	R
C/C++	Julia, Scala, Go, C#

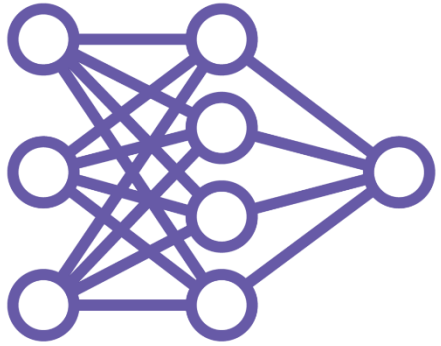
What about JavaScript?



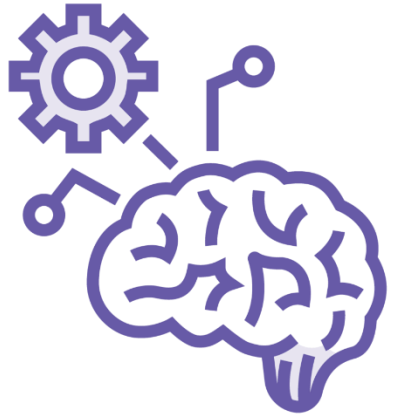
Stack Overflow developer summary report 2020

* Source : https://insights.stackoverflow.com/survey/2020#most-popular-technologies
** Some language are omitted for the sake for brevity

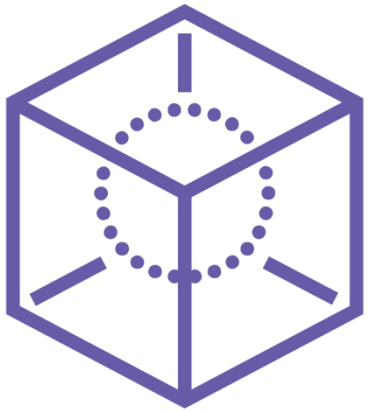
TensorFlow.js



Build



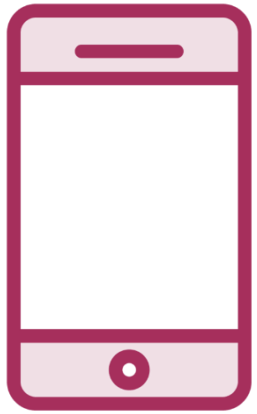
Train



Deploy

Supported Platforms

Browser

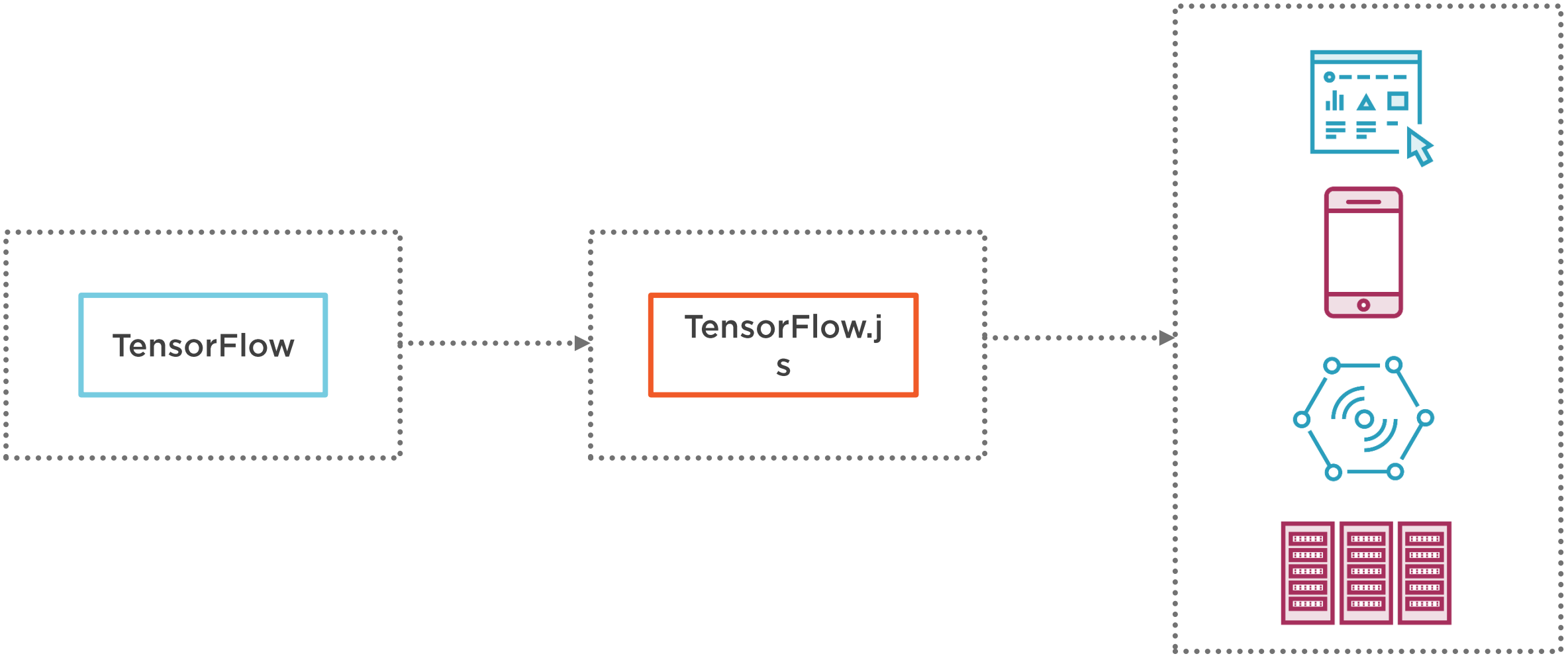


Mobile

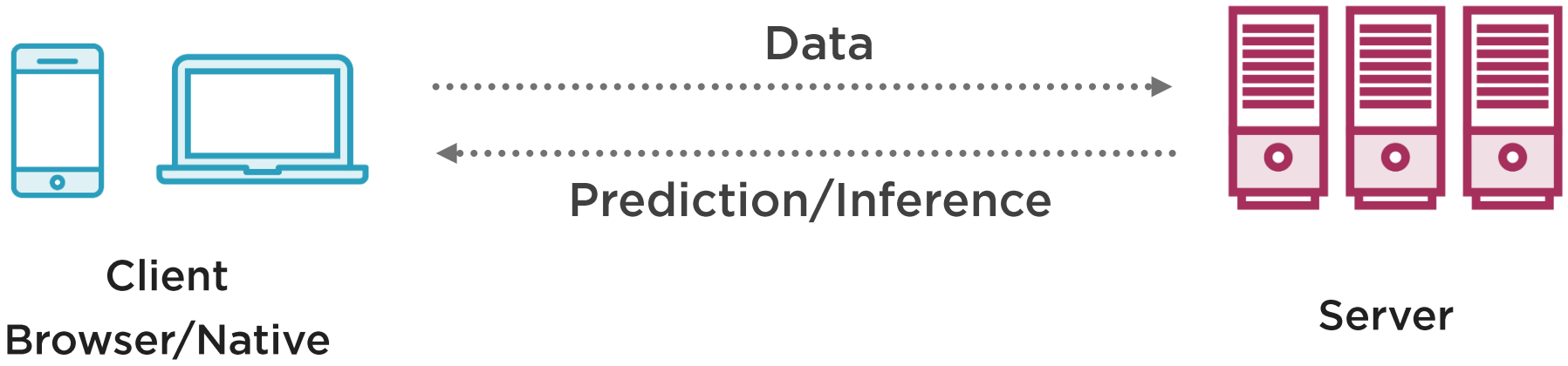
Desktop and IOT

Server

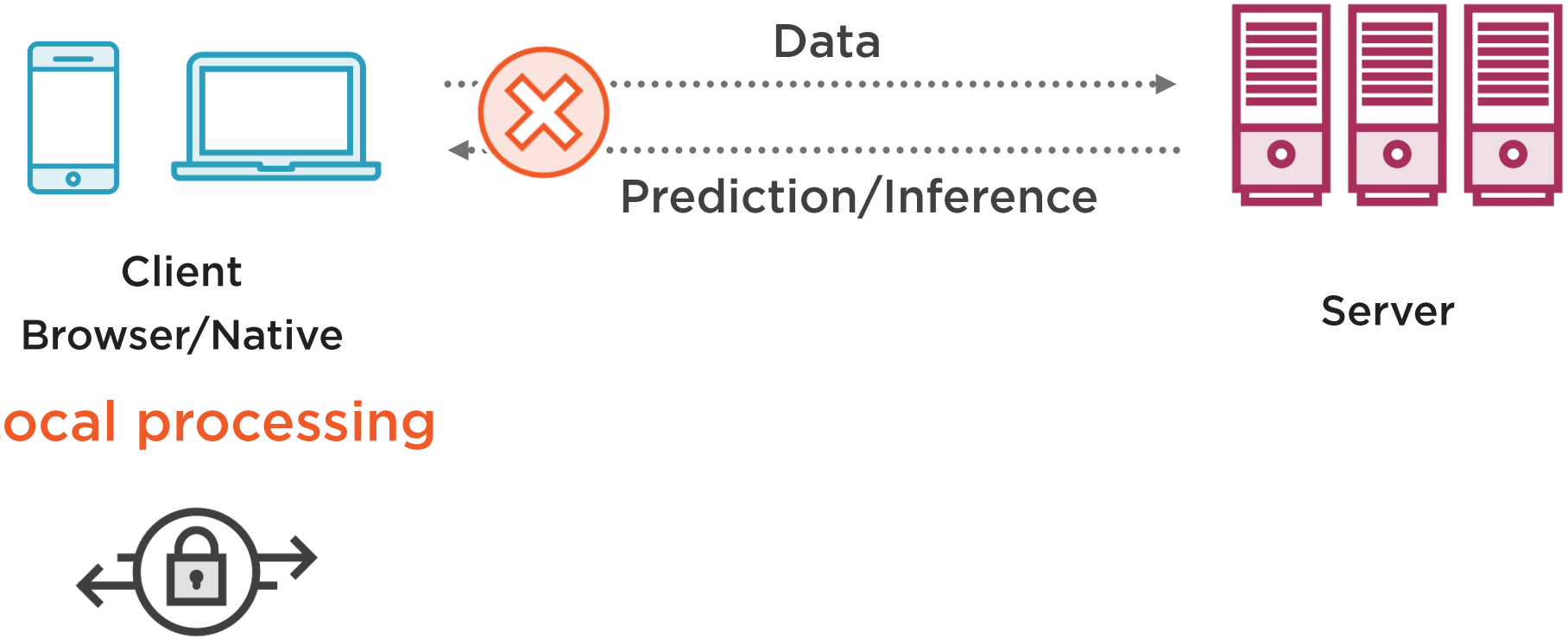
Is it only for JavaScript developers?


TensorFlow
TensorFlow.j
s

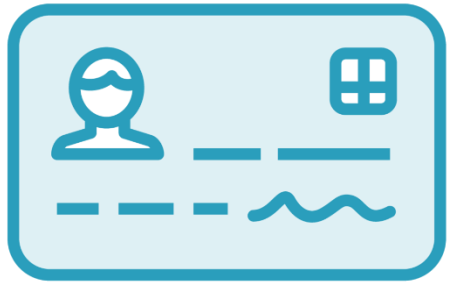
Machine Learning in Browser



Privacy


Data
Prediction/Inference
Client
Browser/Native
Server
Local processing

Privacy

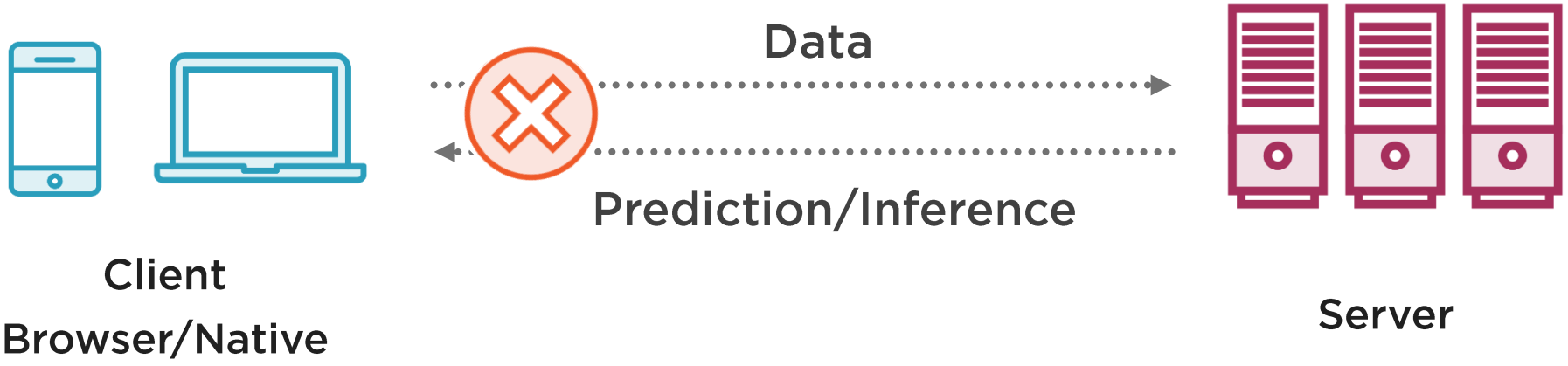


Detecting sensitive document

Health diagnosis

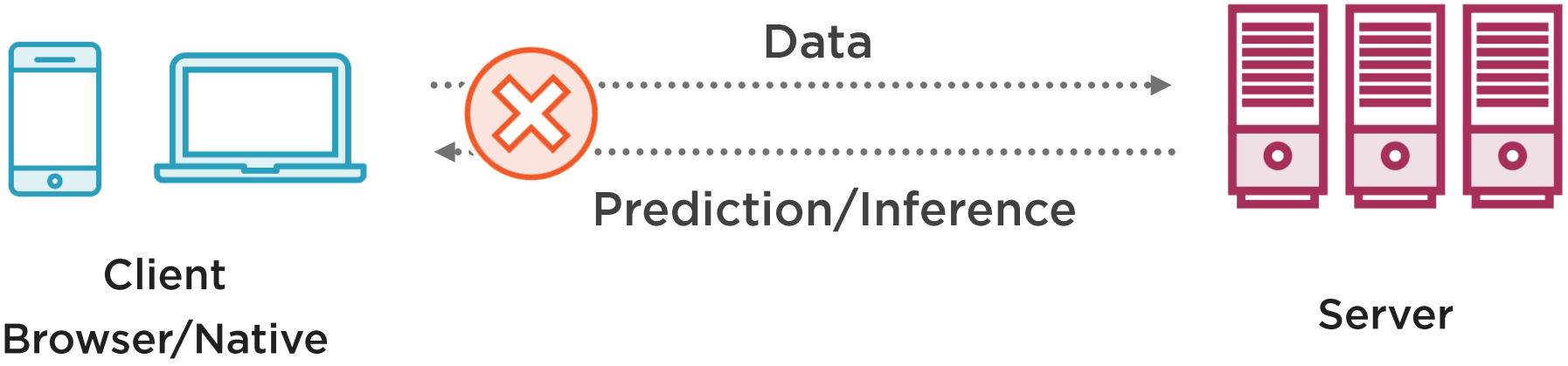
Legal documents

Latency



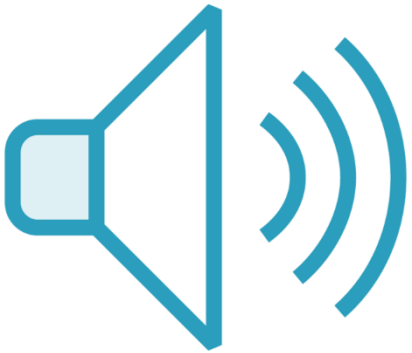
Local processing

Interactivity

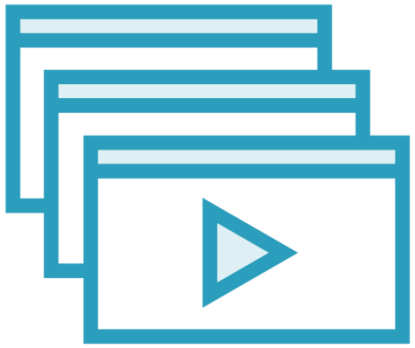


Local processing

Interactivity



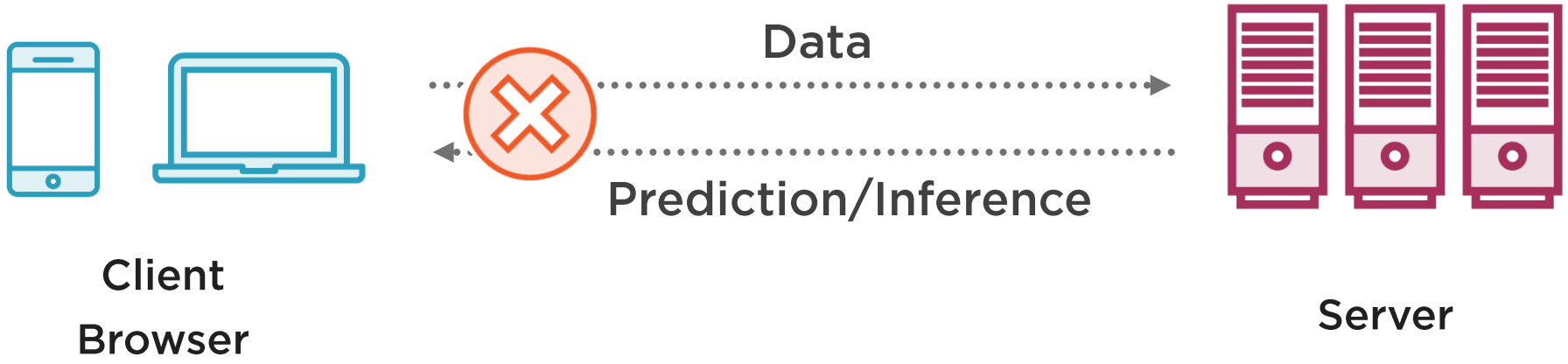
Audio input



Images and video feeds

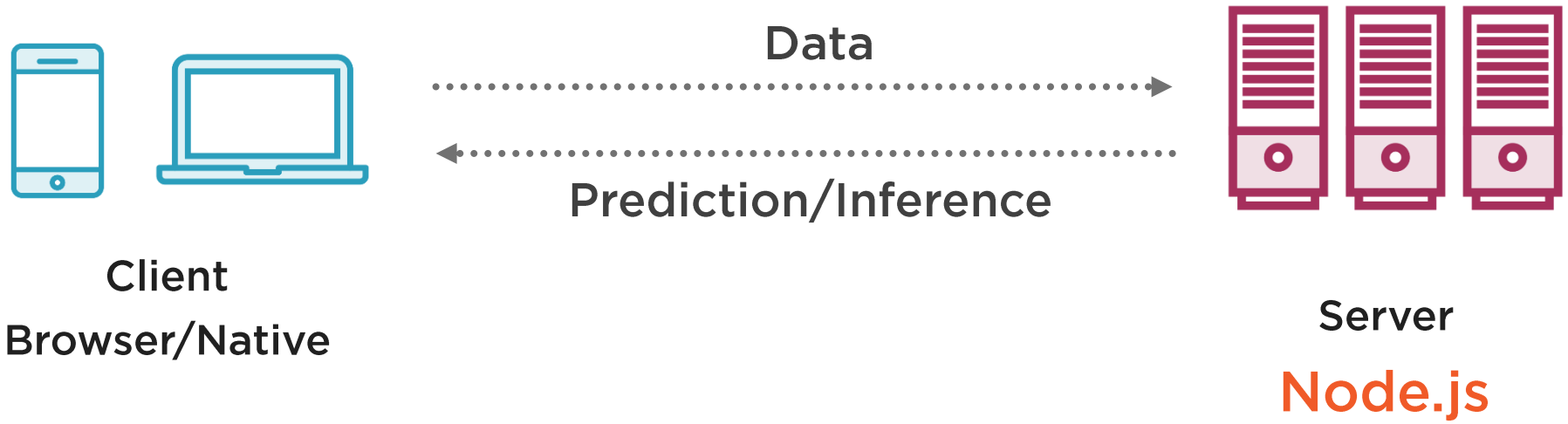
On-device sensors

Distribution

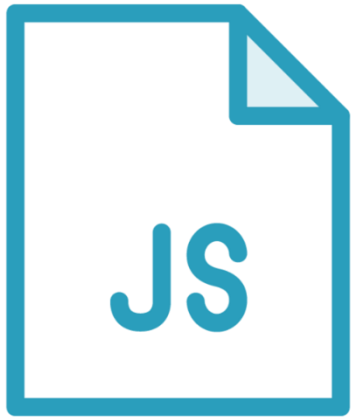


No install needed

TensorFlow.js on Server-side



TensorFlow.js on Server-side



Consistent tech stack

Use existing JS skills

All this goodness at the cost of performance?

Backend

CPU

Plain vanilla JS

WebGL

GPU based

WASM

CPU acceleration with WebAssembly

Node

TensorFlow C API

WebGPU

Near zero overhead execution

WebGL, supported on 97% of devices, can lead to 100x performance improvement.

https://caniuse.com/?search=webgl

https://www.tensorflow.org/js/guide/platform_environment

WASM, supported on 90% of devices, can lead to 10-30x performance improvement than vanilla JS CPU.

https://caniuse.com/?search=wasm

https://www.tensorflow.org/js/guide/platform_environment

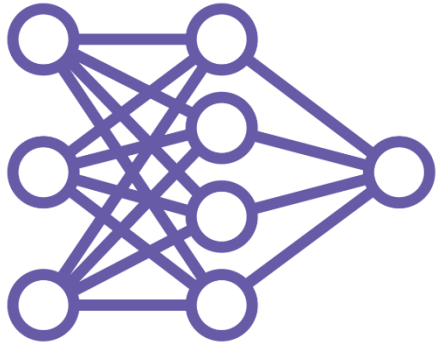
Node backend comparable to Python-based TensorFlow.

https://blog.tensorflow.org/2020/05/how-hugging-face-achieved-2x-performance-boost-question-answering.html

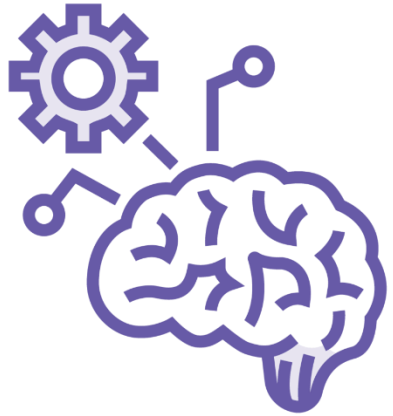
TensorFlow.js

Open source machine learning framework for JavaScript.

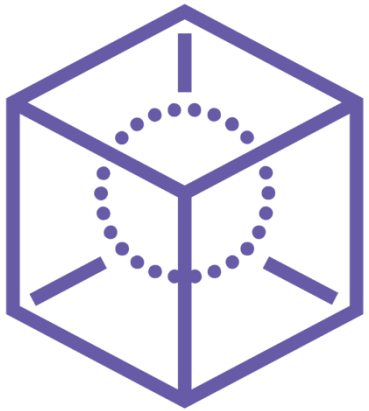
TensorFlow.js



Build



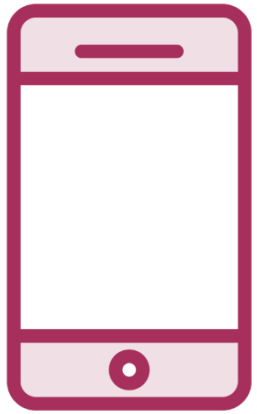
Train



Deploy

Supported Platforms

Browser



Mobile

Desktop and IOT

Server

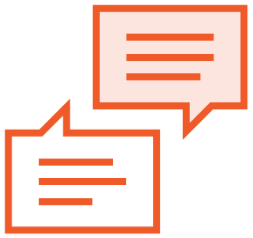
Pre-built Models

Image classification

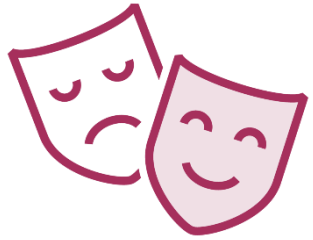
Object segmentation

Pose detection

Speech recognition



Question answering



Text toxicity

```
<script src="https://cdn.jsdelivr.net/npm/@tensorflow/tfjs"> </script>

<script src="https://cdn.jsdelivr.net/npm/@tensorflow-models/qna"> </script>
```

Using Pre-built Model

With just few lines of code

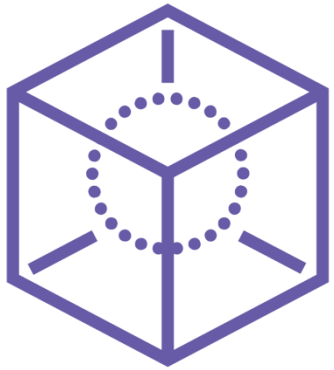
```
npm install @tensorflow/tfjs

npm install @tensorflow-models/qna
```

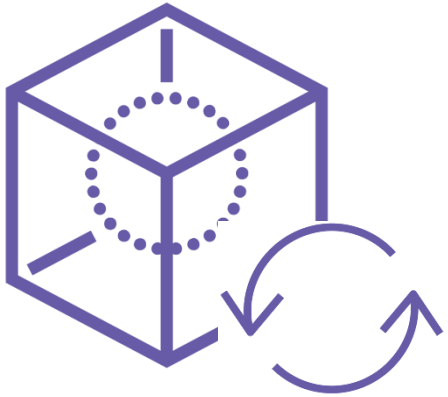
Using Pre-built Model

With just few lines of code

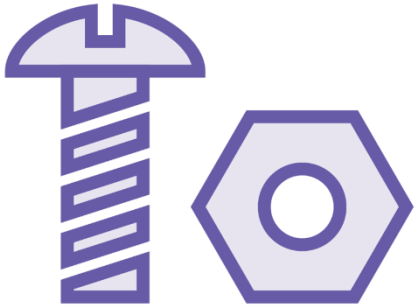
TensorFlow.js



Use existing (pre-trained, converted) model

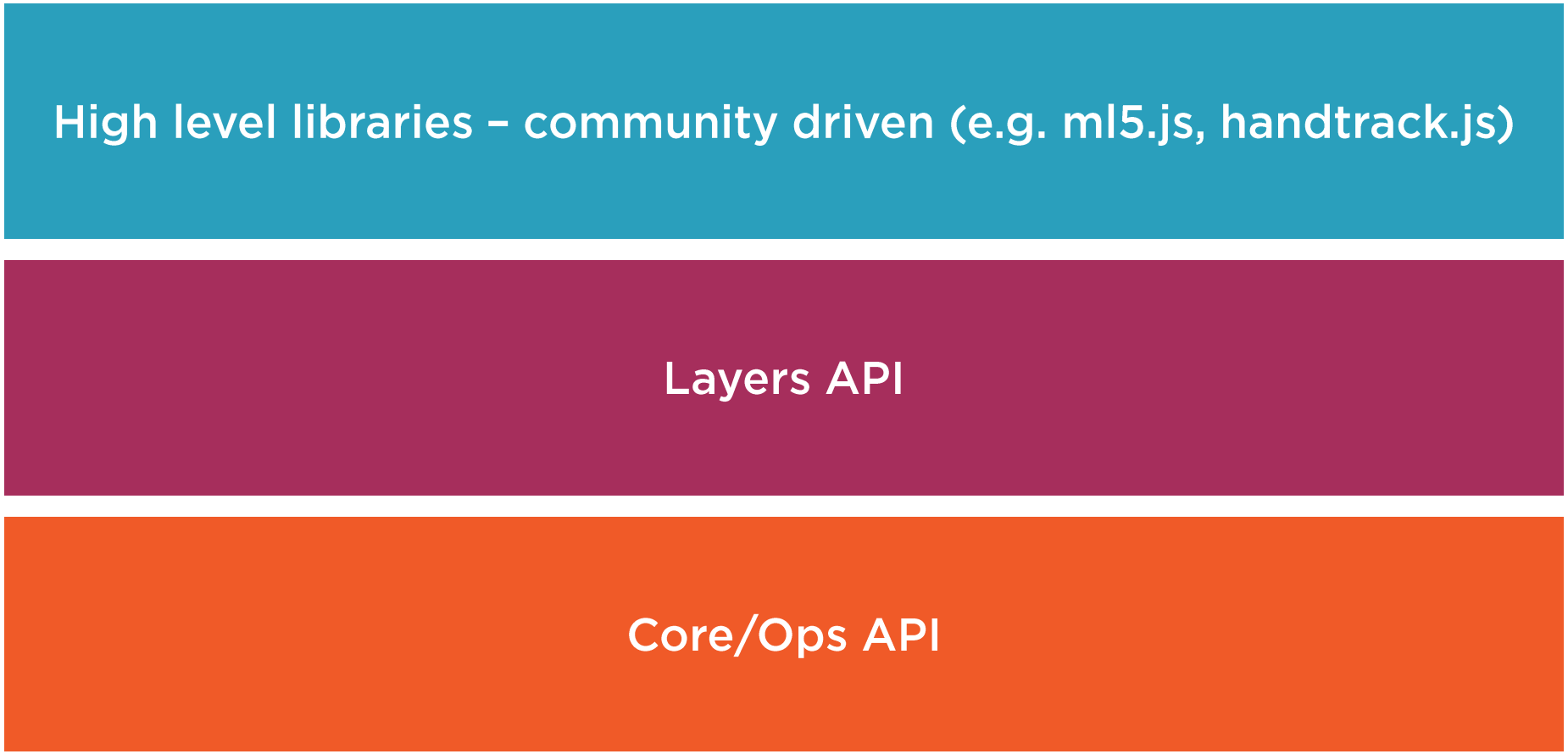


Retrain model on your own data



Build models from scratch

TensorFlow.js APIs

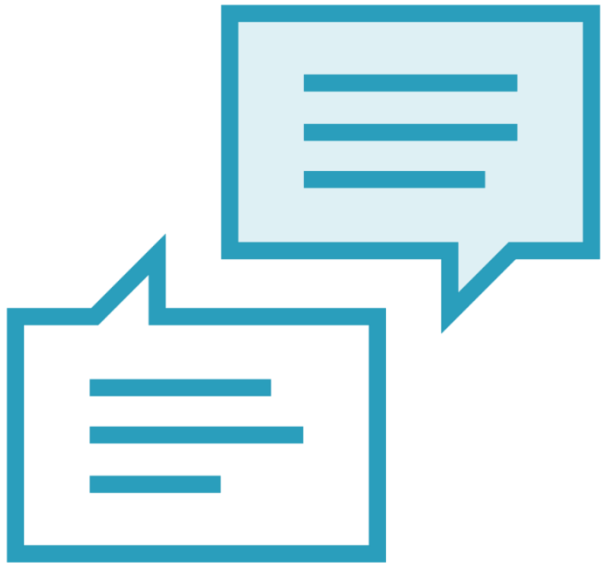


TensorFlow.js Ecosystem

TFJS data

TFJS viz

Course Demo



Online communication platforms

Risk of rude, offensive, toxic comments

Detect toxic comments violating guidelines

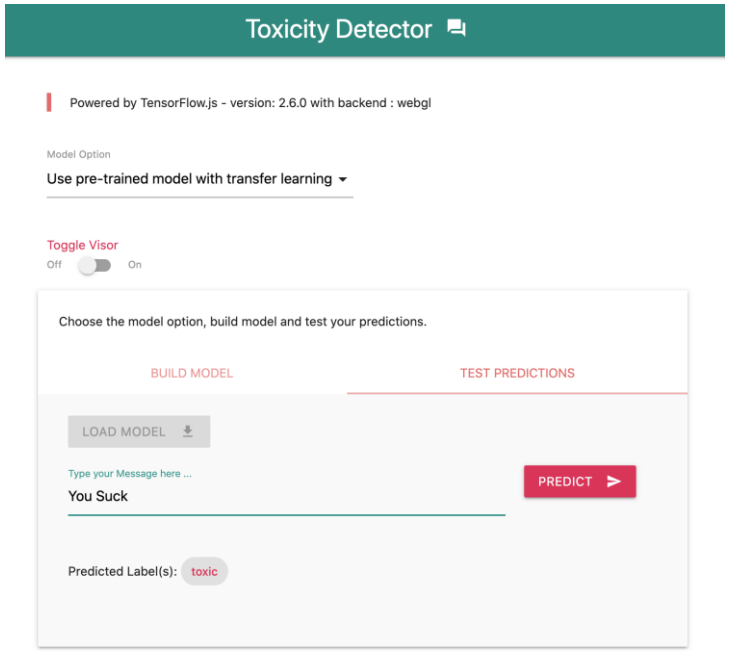
NLP based detection

Server-side detection

Client-side detection

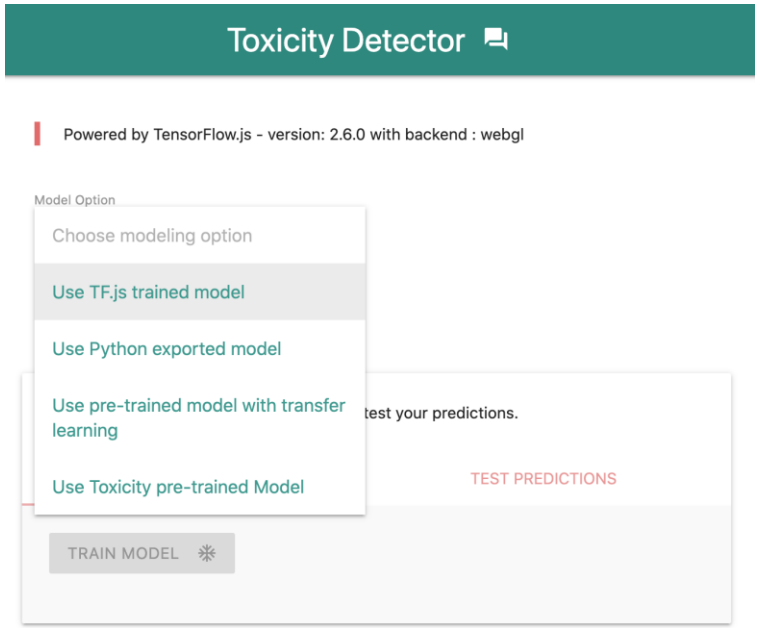
- Feedback before publishing comments
- Eliminate potential privacy concerns
- Reduce server-side cost for detection

Course Demo



TensorFlow.js powered “Toxicity Detector”

Course Demo

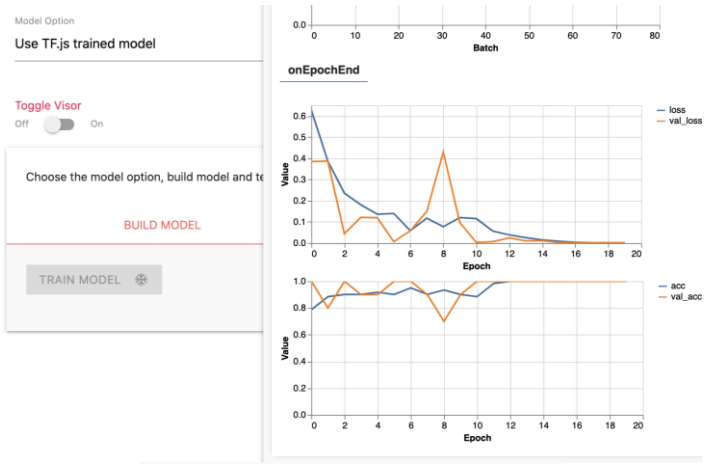


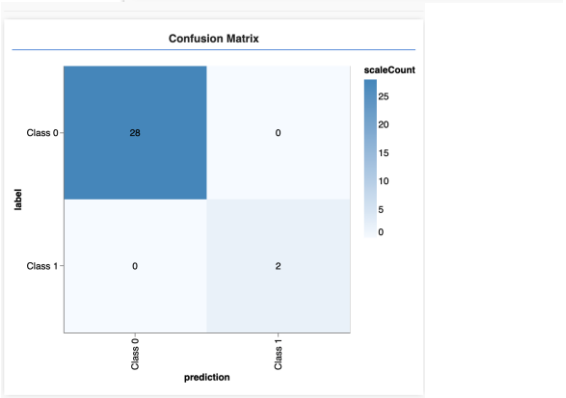
TensorFlow.js powered "Toxicity Detector"

4 flavors

- Build, train, and evaluate from scratch

Course Demo



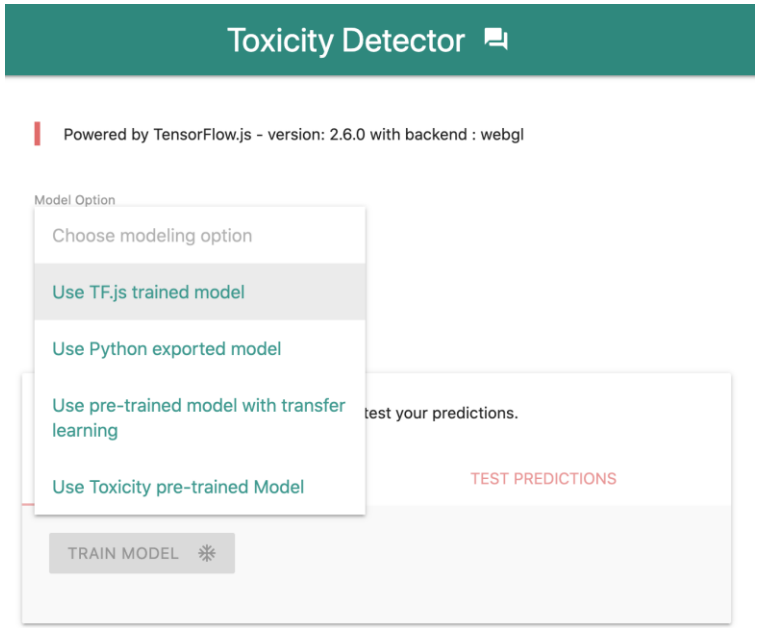


TensorFlow.js powered "Toxicity Detector"

4 flavors

- Build, train, and evaluate from scratch

Course Demo



TensorFlow.js powered "Toxicity Detector"

4 flavors

- Build, train, and evaluate from scratch
- Export from Python
- Transfer learning using Universal Sentence Encoder (USE)
- Pre-built toxicity model

Client-side and server-side

Use template for any other use case

Course Structure

- Setting up TensorFlow.js environment
- Understanding TensorFlow.js core concepts
- Preparing data for ML model
- Building, training and evaluating ML model
- Saving and loading ML model
- Predicting using trained ML model
- Using pre-trained models with TensorFlow.js
- What's next?

Prerequisites

JavaScript basics

Machine learning basics

Up Next:
Setting up TensorFlow.js Environment