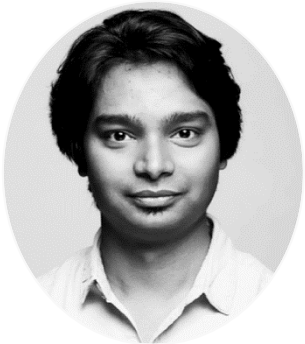


# Predicting Using Trained Machine Learning Model

---



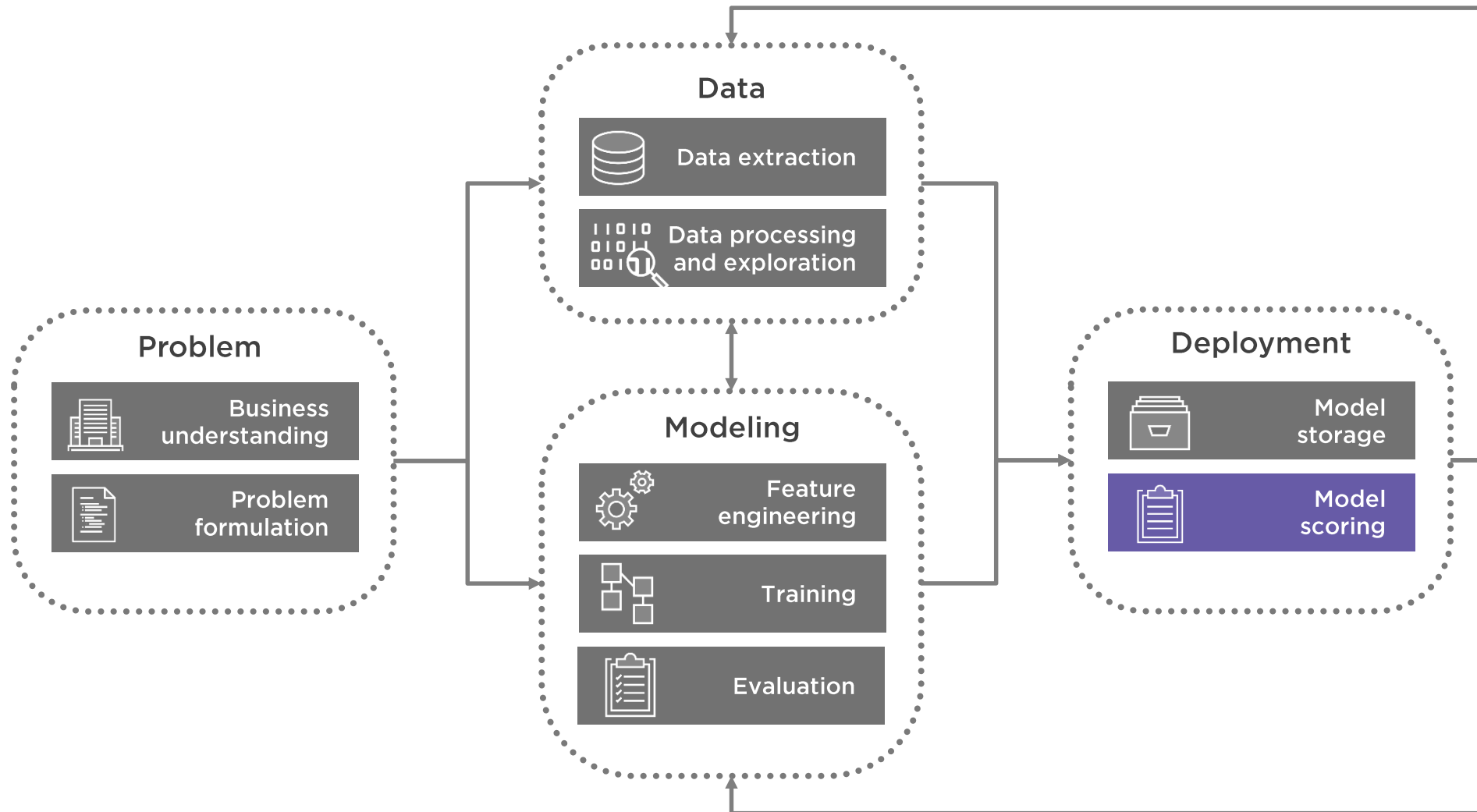
**Abhishek Kumar**

DATA SCIENTIST | AUTHOR | SPEAKER

@meabhishekkumar



# Machine Learning Workflow



# Overview



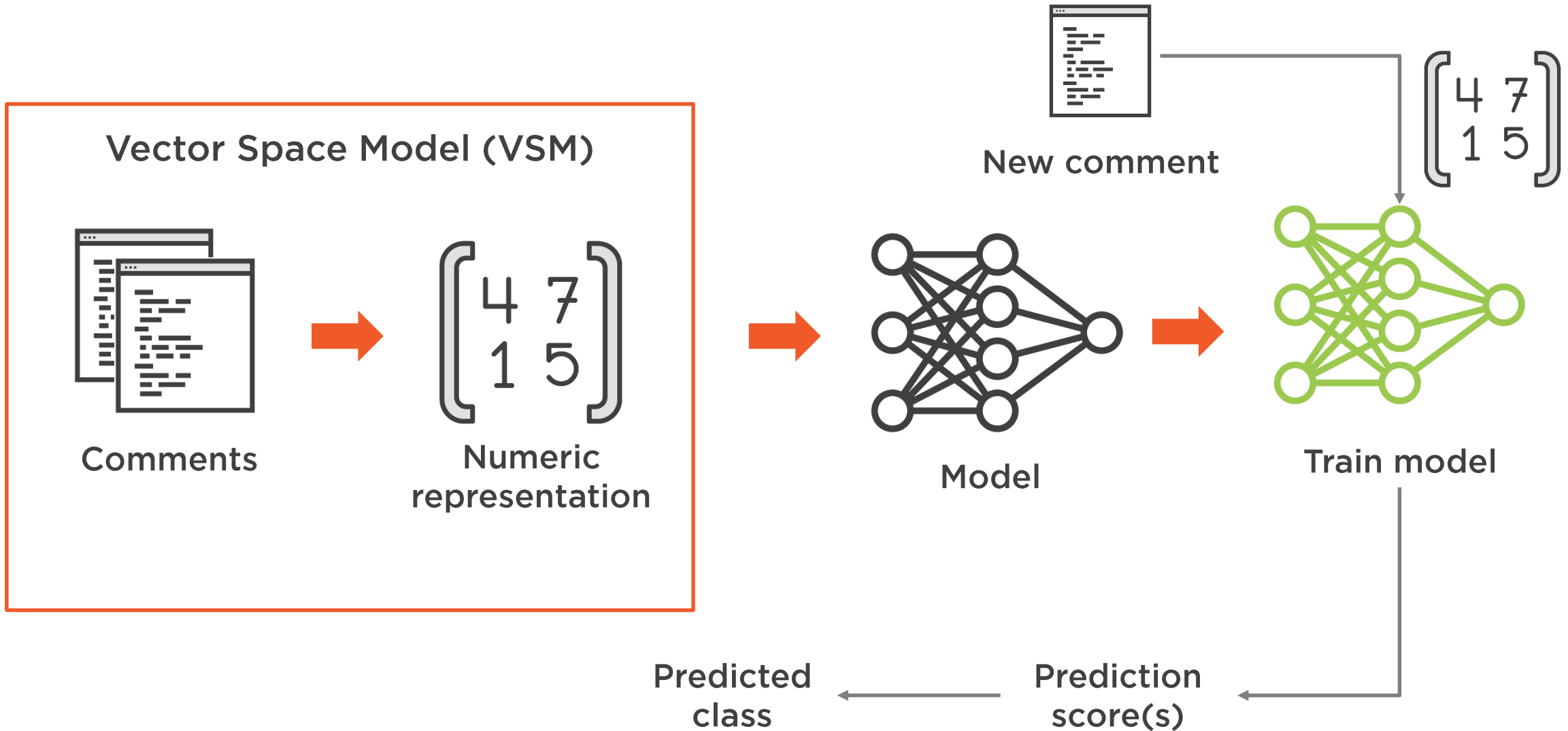
Model prediction

Wire prepare, train, load and predict with UI

Use exported model from Python



# Model Scoring



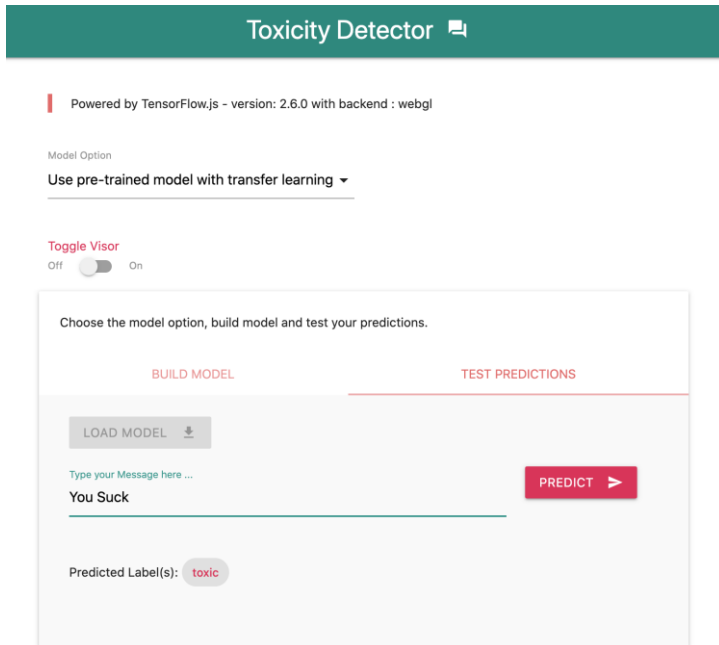
# Demo



Predicting using trained TensorFlow.js model



# Materialize Front-end Framework



**Responsive front-end framework**

**Based on material design**

**Easy to use**

**Can be integrated**

- CDN script tag
- NPM install



# Demo



## Setting up Materialize UI



# Demo

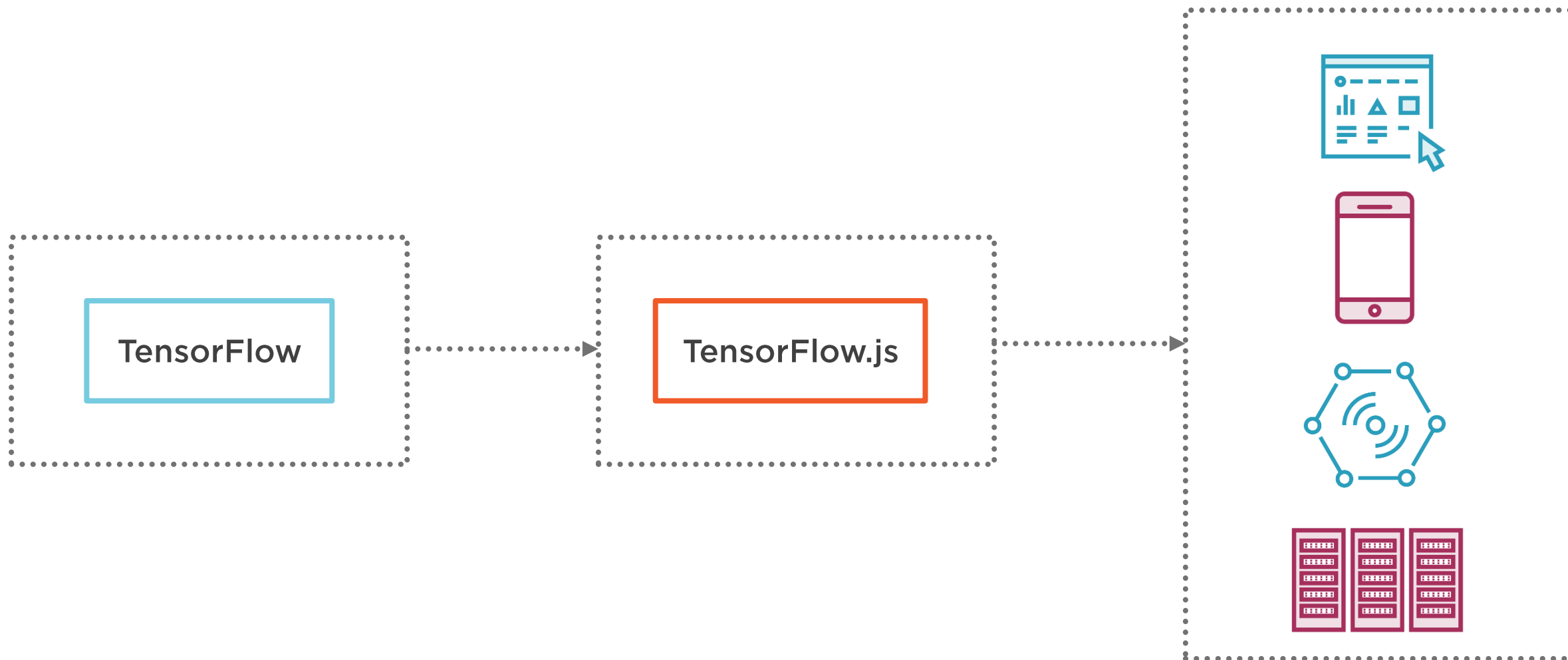


## Integrating steps with UI

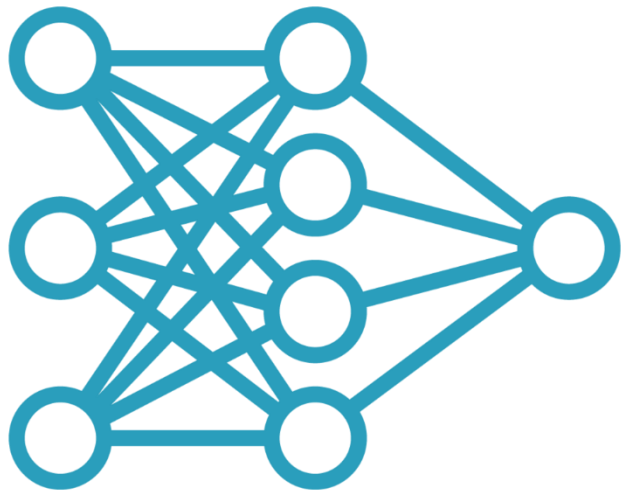




# TensorFlow.js Converter



# Scenarios for TensorFlow.js Exported Model



Existing investment in Python based ecosystem

Use best of the both worlds

Run trained model on the client side

Retrain in TensorFlow.js



# Demo



## Predicting using Python exported model



# Summary



## Model scoring

- Predict method to get score
- Apply threshold to get class

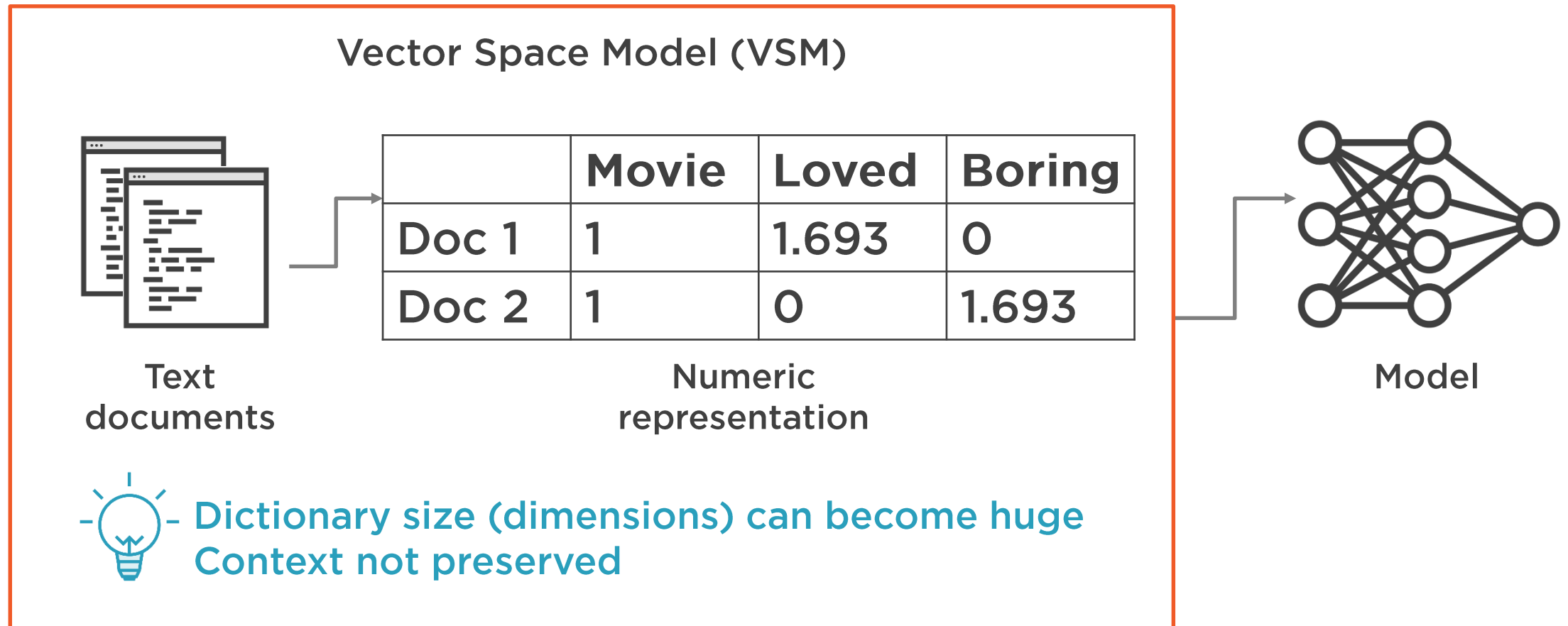
## UI

- Materialize front-end framework
- Integrate all steps

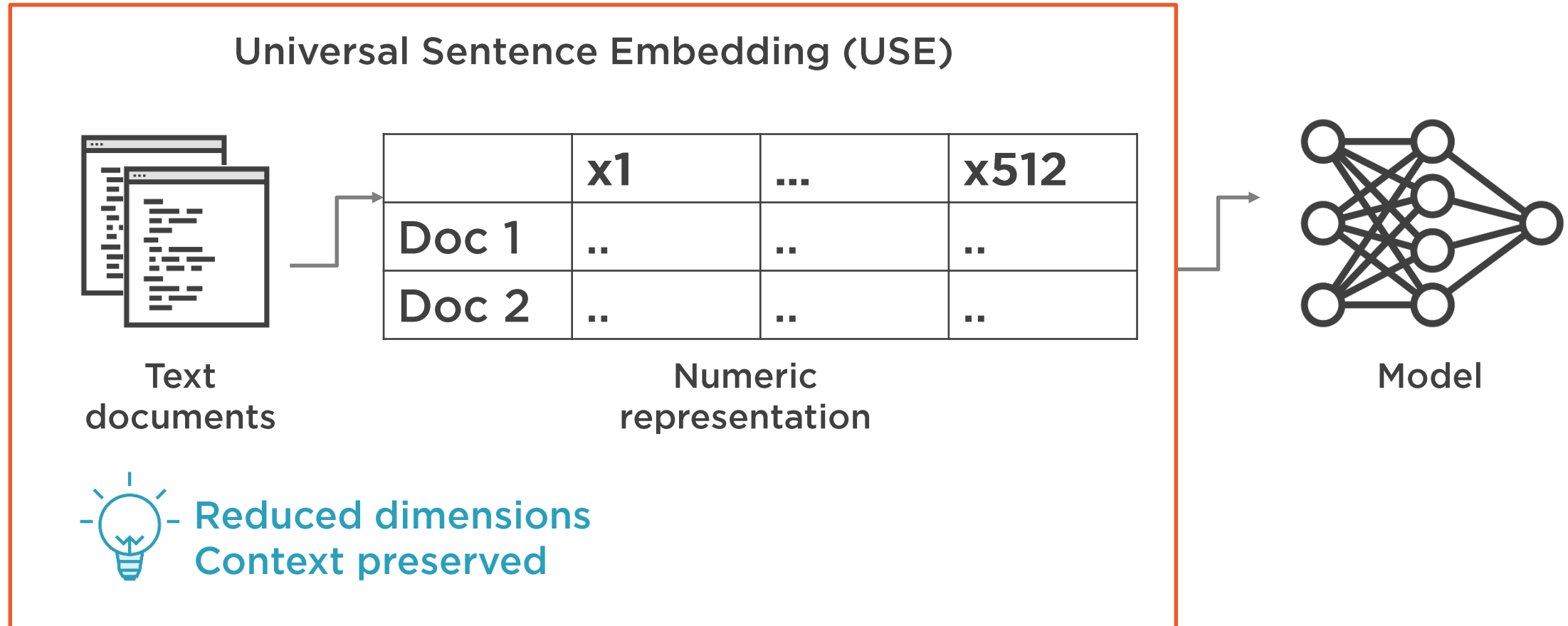
## Use model from Python

- Use TensorFlow.js converter
- Load model

# Generating Features from Text



# Generating Features from Text



Up Next:

Using Pre-trained Models with TensorFlow.js

---

