

Ready to Use Deep Learning Models

Gabriela de Queiroz

Sr. Engineering & Data Science Manager, IBM
Founder, R-Ladies & AI Inclusive

 [@gdequeiroz](https://twitter.com/gdequeiroz) | k-roz.com

IBM Developer

slides: bit.ly/max-dsgo

IBM

Gabriela de Queiroz

- Founder of R-Ladies
- Founder of AI Inclusive (ai-inclusive.org)
- Member of R Foundation
- Sr. Engineering & Data Science Manager, IBM



Data Scientist + Developer Advocate + Open Source Developer +
Manager + Community Builder + Mentor

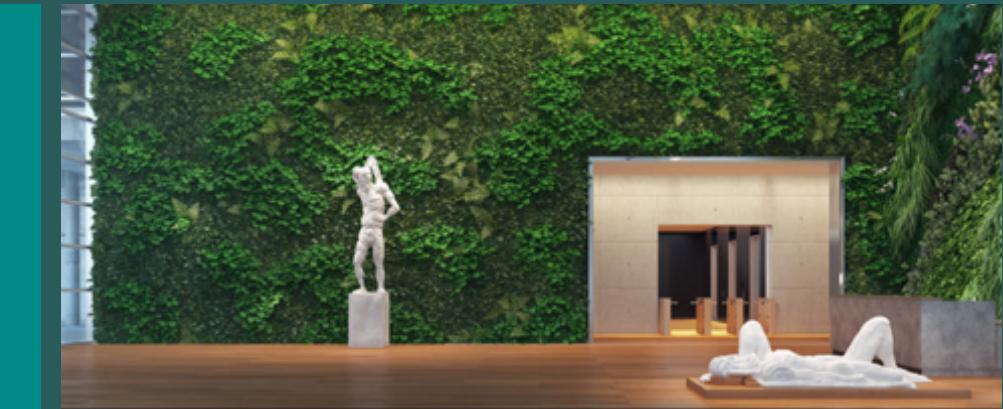
Center for Open Source Data and AI Technologies (CODAIT)

30+ open source developers!

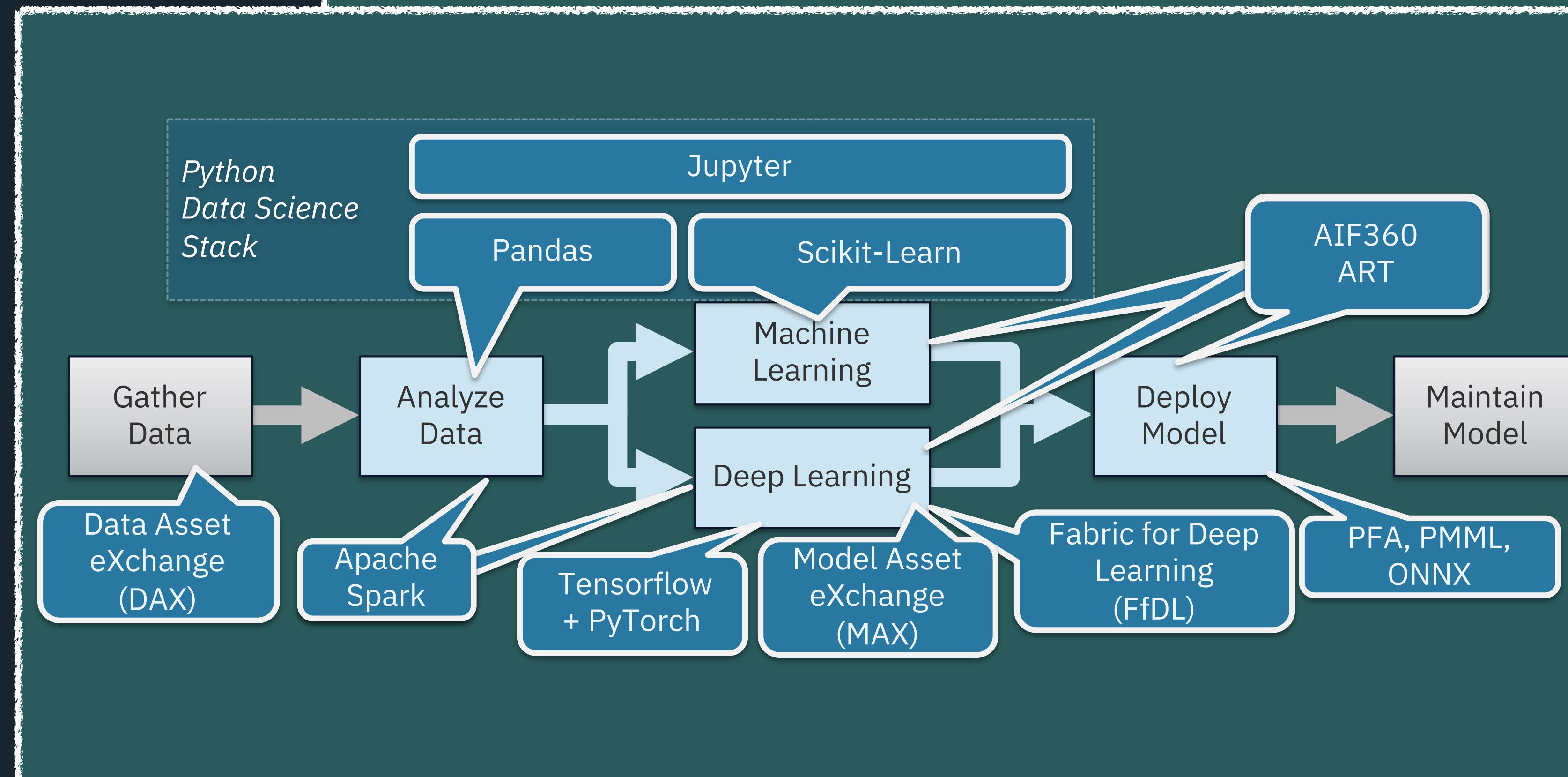
**Build tools to make
AI accessible to all**

CODAIT

codait.org



Watson West Building
505 Howard St.
San Francisco, California



About 262,000,000 results (0.55 seconds)

Technical Skills: Computer Science

- Python Coding. Python is the most common coding language I typically see required in data science roles, along with Java, Perl, or C/C++. ...
- Hadoop Platform. ...
- SQL Database/Coding. ...
- Apache Spark. ...
- Machine Learning and AI. ...
- Data Visualization. ...
- Unstructured data.



9 Must-have skills you need to become a Data Scientist, updated

<https://www.kdnuggets.com/2018/05/simplilearn-9-must-have-skills-data...>

A data scientist needs strong math skills, particularly in multivariable calculus and linear algebra.

- Identifying Algorithms.
- Creating and Maintaining Algorithms.
- Information Retrieval Data Sets.
- Linear Algebra.
- Machine Learning Models.
- Machine Learning Techniques.
- Multivariable Calculus.
- Statistics.

[More items...](#)

Important Job Skills for Data Scientists - The Balance Careers

<https://www.thebalancecareers.com/list-of-data-scientist-skills-2062381>



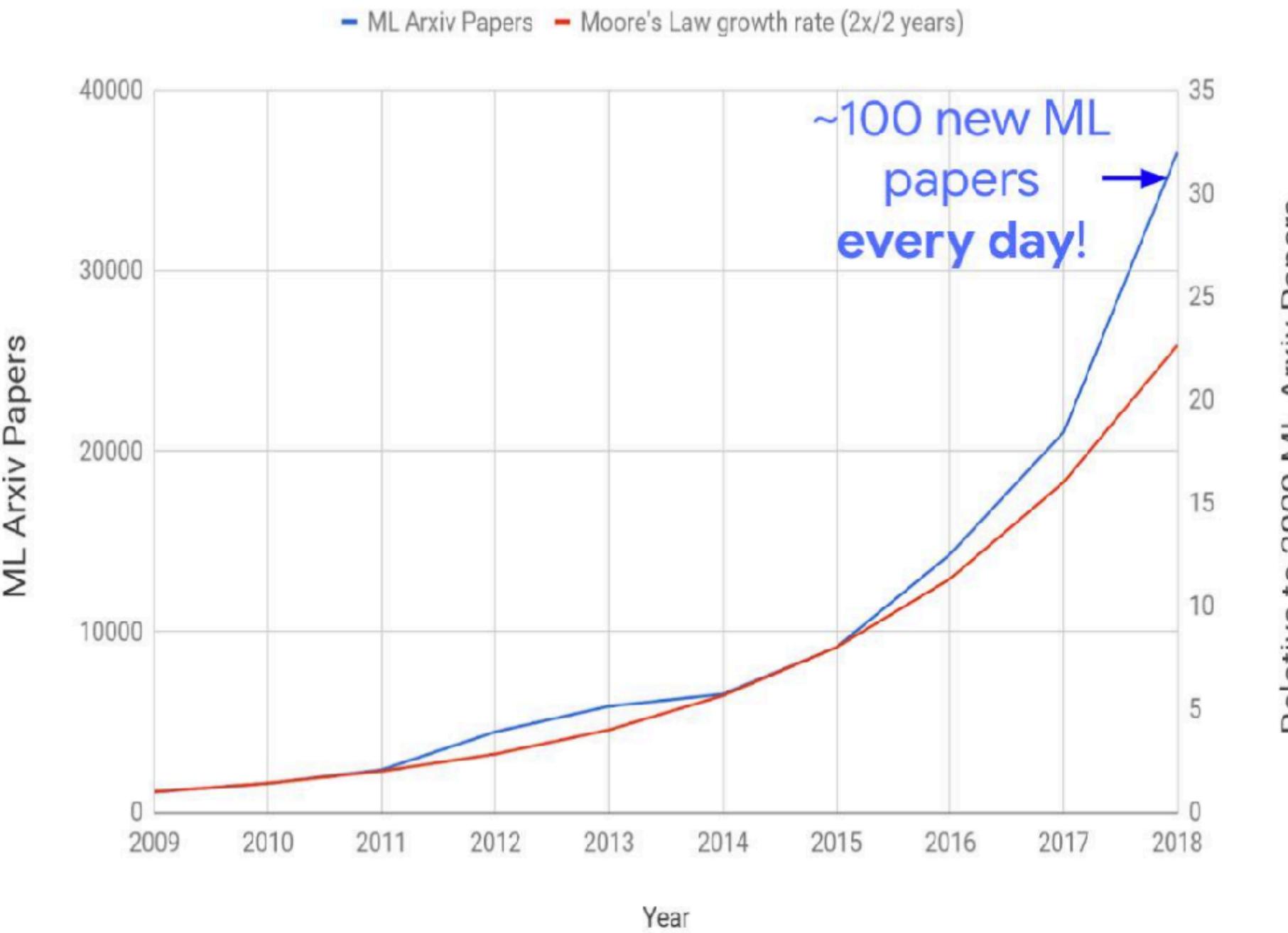
Jeff Dean ✅
@JeffDean

Arxiv ML papers/day

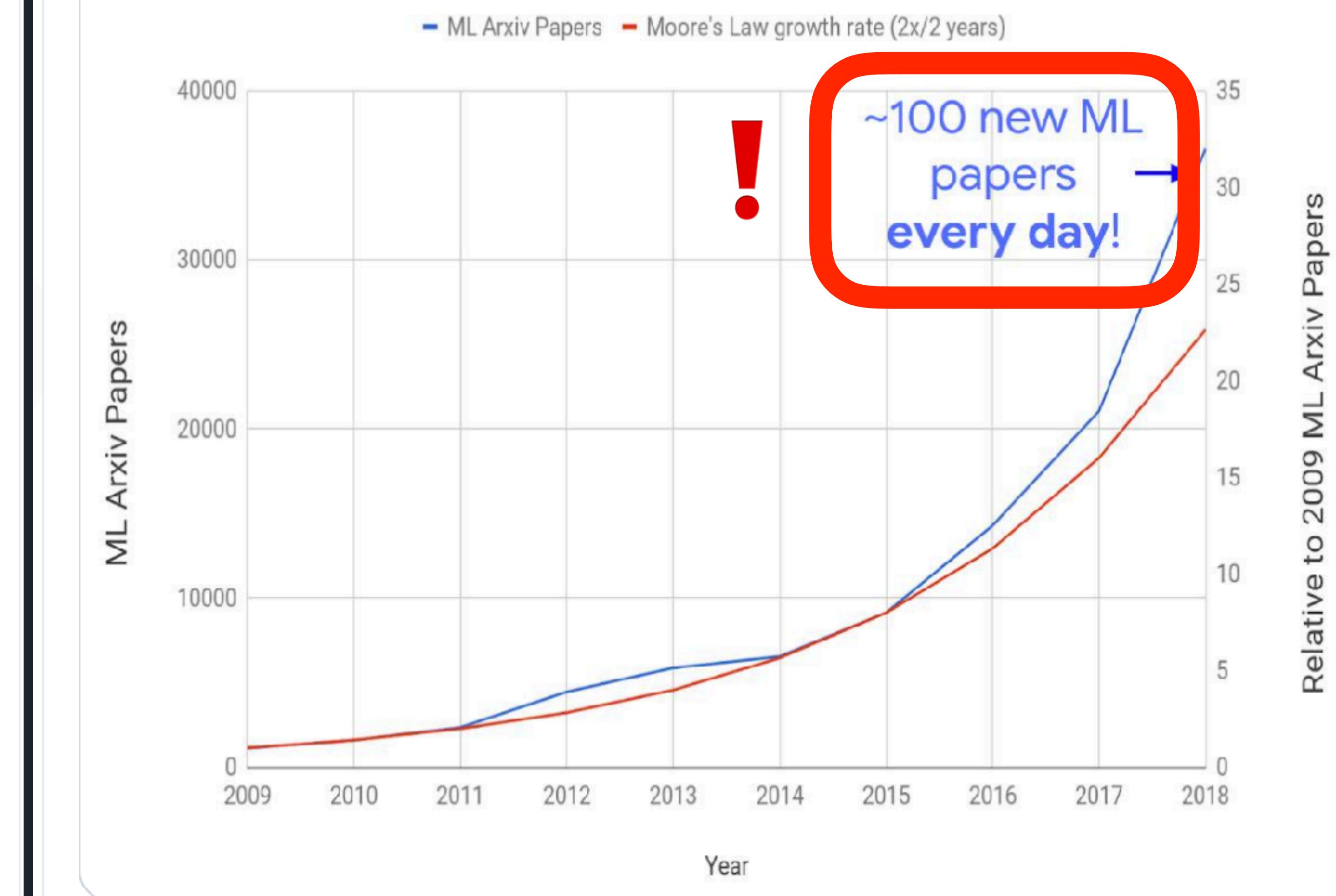
I have some bad news for you, [@deliprao](#). I collect data on the "ML subset" of [@arxiv_org](#) and its growth.

Absolute numbers are approx ~3000 new papers/month at end of 2018 (100/day!) & likely higher now, well past the bioRxiv 2500/month that you feared!

Machine Learning Arxiv Papers per Year



Machine Learning Arxiv Papers per Year



<https://twitter.com/JeffDean/status/1135114657344237568?s=20>



Articles

About 4,500,000 results (0.05 sec)

> 4 million results!

deep learning courses



All

News

Videos

Images

Shopping

More

Settings

Tools

About 183,000,000 results (0.73 seconds)

> 183 million results!



Model Asset eXchange

Place for developers/data scientists to find and use
free and **open source** deep learning models

ibm.biz/model-exchange

Model Asset eXchange

Free, deployable, and trainable code.

A place for developers to find and use free and open source deep learning models.

[View all models >](#)

[Try the tutorial >](#)

[Join the community >](#)

Featured

Deployable

Trainable

Deployable | Text Classification

Toxic Comment Classifier

Detect 6 types of toxicity in user comments

[View model »](#)

Artificial intelligence Docker +

Deployable | Object Detection In Images

Object Detector

Localize and identify multiple objects in a single image.

[View model »](#)

Artificial intelligence Deep learning +

Deployable, Trainable | Text Classification

Text Sentiment Classifier

Detect the sentiment captured in short pieces of text

[View model »](#)

Artificial intelligence Deep learning +

Deployable | Audio Classification

Audio Classifier

Identify sounds in short audio clips.

[View model »](#)

Artificial intelligence Audio Classification +

<p>Deployable, Trainable</p> <h3>Question Answering</h3> <p>Answer questions on a given corpus of text.</p> <p>View model »</p> <p>(Artificial Intelligence) (Deep learning) +</p>	<p>Deployable ...</p> <h3>Text Summarizer</h3> <p>Generate a summarized description of a body of text.</p> <p>View model »</p> <p>(Artificial Intelligence) (Deep learning) +</p>	<p>Deployable Text Classification</p> <h3>Toxic Comment Classifier</h3> <p>Detect 6 types of toxicity in user comments.</p> <p>View model »</p> <p>(Artificial Intelligence) (Docker) +</p>	<p>Deployable ...</p> <h3>Chinese Phonetic Similarity Estimator</h3> <p>Estimate the phonetic distance between Chinese words and get similar sounding candidate words.</p> <p>View model »</p> <p>(Artificial Intelligence) (Docker) +</p>	<p>Deployable ...</p> <h3>Image Colorizer</h3> <p>Adds color to black and white images.</p> <p>View model »</p> <p>(Artificial Intelligence) (Deep learning) +</p>	<p>Deployable Audio Modeling</p> <h3>Audio Sample Generator</h3> <p>Generate short audio clips of speech commands and lo-fi instrumental samples.</p> <p>View model »</p> <p>(Artificial Intelligence) (Audio Modeling) +</p>	<p>Deployable, Trainable ...</p> <h3>Image Segmenter</h3> <p>Identify objects in an image, additionally assigning each pixel of the image to a particular object.</p> <p>View model »</p> <p>(Artificial Intelligence) (Deep learning) +</p>	<p>Deployable Image Classification</p> <h3>Breast Cancer Mitosis Detector</h3> <p>Detect whether a mitosis exists in an image of breast cancer tumor cells.</p> <p>View model »</p> <p>(Artificial Intelligence) (Deep learning) +</p>
<p>Deployable ...</p> <h3>Image Resolution Enhancer</h3> <p>Upscale an image by a factor of 4, while generating photo-realistic details.</p> <p>View model »</p> <p>(Artificial Intelligence) (Deep learning) +</p>	<p>Deployable, Trainable ...</p> <h3>Text Sentiment Classifier</h3> <p>Detect the sentiment captured in short pieces of text.</p> <p>View model »</p> <p>(Artificial Intelligence) (Deep learning) +</p>	<p>Deployable ...</p> <h3>Nucleus Segmenter</h3> <p>Identify nuclei in a microscopy image and assign each pixel of the image to a particular nucleus.</p> <p>View model »</p> <p>(Artificial Intelligence) (Deep learning) +</p>	<p>Deployable Audio Modeling</p> <h3>Speech to Text Converter</h3> <p>Converts spoken words into text form.</p> <p>View model »</p> <p>(Artificial Intelligence) (Audio Modeling) +</p>	<p>Deployable Image Classification</p> <h3>Image Classifier - Inception ResNet v2</h3> <p>Identify objects in images using a third-generation deep residual network.</p> <p>View model »</p> <p>(Artificial Intelligence) (Deep learning) +</p>	<p>Deployable ...</p> <h3>Image Caption Generator</h3> <p>Generate captions that describe the contents of images.</p> <p>View model »</p> <p>(Artificial Intelligence) (Deep learning) +</p>	<p>Trainable ...</p> <h3>Name Generator</h3> <p>Generate names based on a dataset of existing baby names.</p> <p>View model »</p> <p>(Artificial Intelligence) (Deep learning) +</p>	<p>Deployable Language Modeling</p> <h3>News Text Generator</h3> <p>Generate English-language text similar to the news articles in the One Billion Words data set.</p> <p>View model »</p> <p>(Artificial Intelligence) (Deep learning) +</p>
<p>Deployable ...</p> <h3>Human Pose Estimator</h3> <p>Detect humans in an image and estimate the pose for each person.</p> <p>View model »</p> <p>(Artificial Intelligence) (Deep learning) +</p>	<p>Deployable, Trainable ...</p> <h3>Named Entity Tagger</h3> <p>Locate and tag named entities in text.</p> <p>View model »</p> <p>(Artificial Intelligence) (Deep learning) +</p>	<p>Deployable ...</p> <h3>Fast Neural Style Transfer</h3> <p>Generate a new image that mixes the content of a source image with the style of another image.</p> <p>View model »</p> <p>(Artificial Intelligence) (Deep learning) +</p>	<p>Deployable Audio Feature Extraction</p> <h3>Audio Embedding Generator</h3> <p>Generate embedding vectors from audio files.</p> <p>View model »</p> <p>(Artificial Intelligence) (Audio Feature Extraction) +</p>	<p>Deployable, Trainable ...</p> <h3>Object Detector</h3> <p>Localize and identify multiple objects in a single image.</p> <p>View model »</p> <p>(Artificial Intelligence) (Deep learning) +</p>	<p>Deployable, Trainable ...</p> <h3>Image Classifier - ResNet50</h3> <p>Identify objects in images using a first-generation deep residual network.</p> <p>View model »</p> <p>(Artificial Intelligence) (Deep learning) +</p>	<p>Deployable Language Modeling</p> <h3>Review Text Generator</h3> <p>Generate English-language text similar to the text in the Yelp® review data set.</p> <p>View model »</p> <p>(Artificial Intelligence) (Deep learning) +</p>	<p>Deployable Image Classification</p> <h3>Scene Classifier</h3> <p>Classify images according to the place/location labels in the Places365 data set.</p> <p>View model »</p> <p>(Artificial Intelligence) (Deep learning) +</p>

30 ready to use deep learning models

Model Asset eXchange (MAX)

- Wide variety of domains (text, audio, image, etc)
- Multiple deep learning frameworks (TensorFlow, PyTorch, Keras)
- **Trainable** and **Deployable** versions

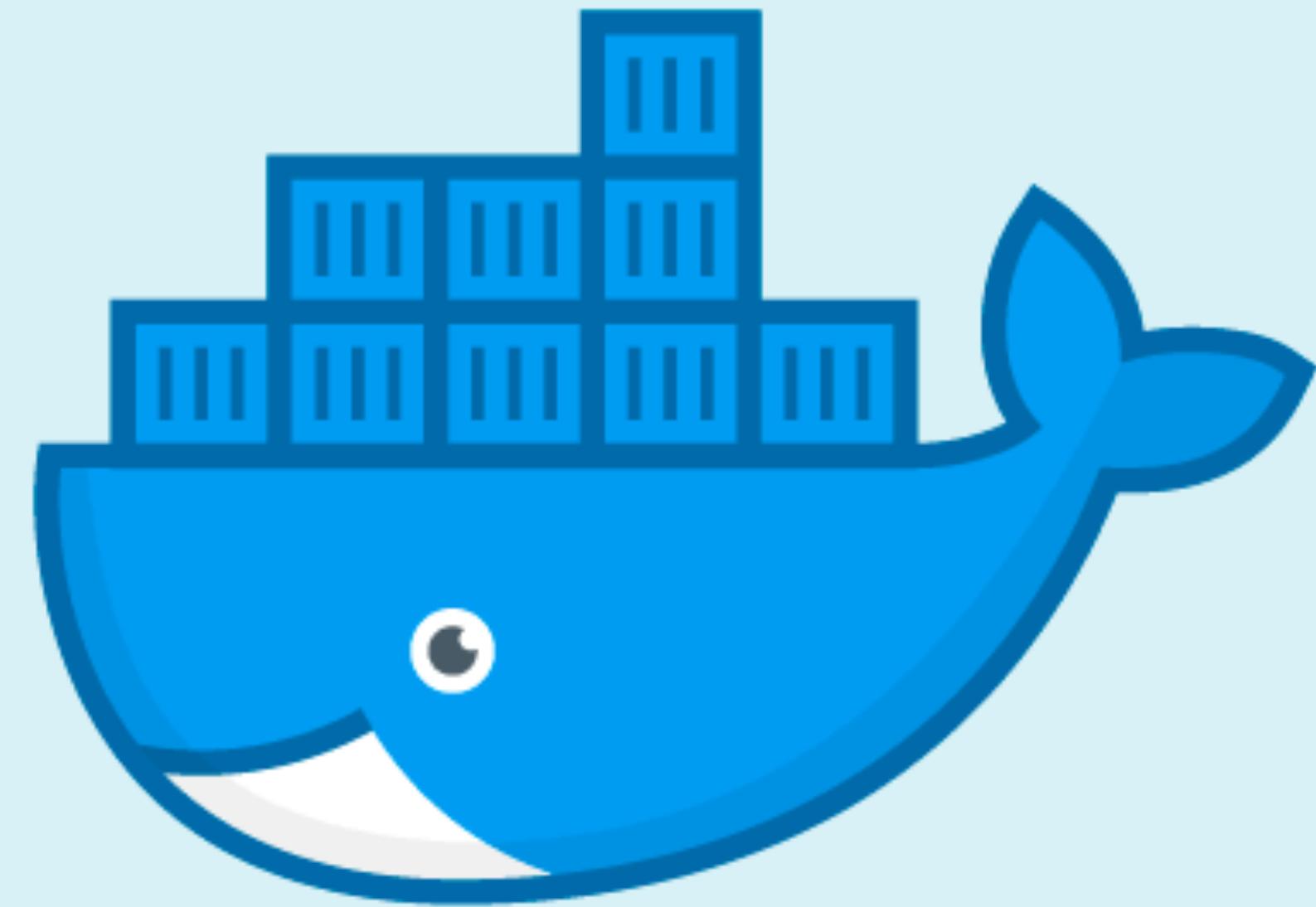
The screenshot shows the homepage of the Model Asset eXchange (MAX) website. The header features the title "Model Asset eXchange" in large white font, with "Free, deployable, and trainable code." and "A place for developers to find and use free and open source deep learning models." Below the header are three buttons: "View all models >", "Try the tutorial >", and "Join the community >". A navigation bar at the bottom includes tabs for "Featured" (which is selected), "Deployable", and "Trainable". The main content area displays four model cards:

- Toxic Comment Classifier** (Deployable | Text Classification): Detects 6 types of toxicity in user comments. Includes a "View model >" button and tags: Artificial intelligence, Docker, +.
- Text Sentiment Classifier** (Deployable, Trainable | Text Classification): Detects sentiment in short pieces of text. Includes a "View model >" button and tags: Artificial intelligence, Deep learning, +.
- Object Detector** (Deployable | Object Detection In Images): Localizes and identifies multiple objects in a single image. Includes a "View model >" button and tags: Artificial intelligence, Deep learning, +.
- Audio Classifier** (Deployable | Audio Classification): Identifies sounds in short audio clips. Includes a "View model >" button and tags: Artificial intelligence, Audio Classification, +.

ibm.biz/model-exchange

What do I need to get started?

START
HERE.

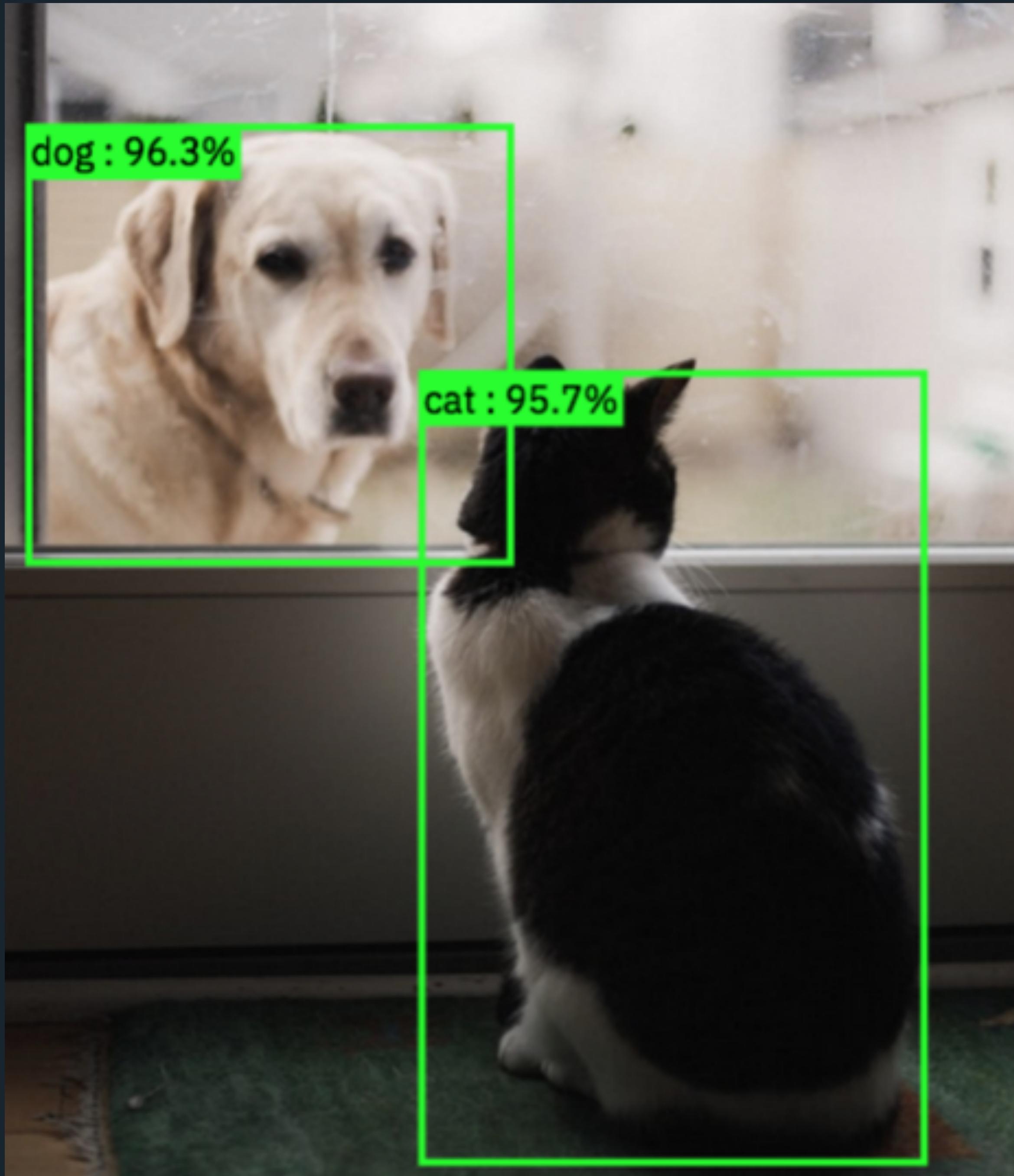


docker

<https://www.docker.com>

Ways of accessing the models





OBJECT DETECTOR

Localize and identify multiple objects in a single image

Model Asset eXchange

Free, deployable, and trainable code.

A place for developers to find and use free and open source deep learning models.

[View all models >](#)

[Try the tutorial >](#)

[Join the community >](#)

Featured

Deployable

Trainable

Deployable | Text Classification

Toxic Comment Classifier

Detect 6 types of toxicity in user comments

[View model »](#)

Artificial intelligence Docker +

Deployable | Object Detection In Images

Object Detector

Localize and identify multiple objects in a single image.

[View model »](#)

Artificial intelligence Deep learning +

Deployable, Trainable

Object Detection In Images

Object Detector

Localize and identify multiple objects in a single image.

[Get this model](#)

[Try the API](#)

[Try the web app](#)

[Try in a Node-RED flow](#)

[Try in CodePen](#)

By IBM Developer Staff | Updated September 21, 2018 - Published March 20, 2018

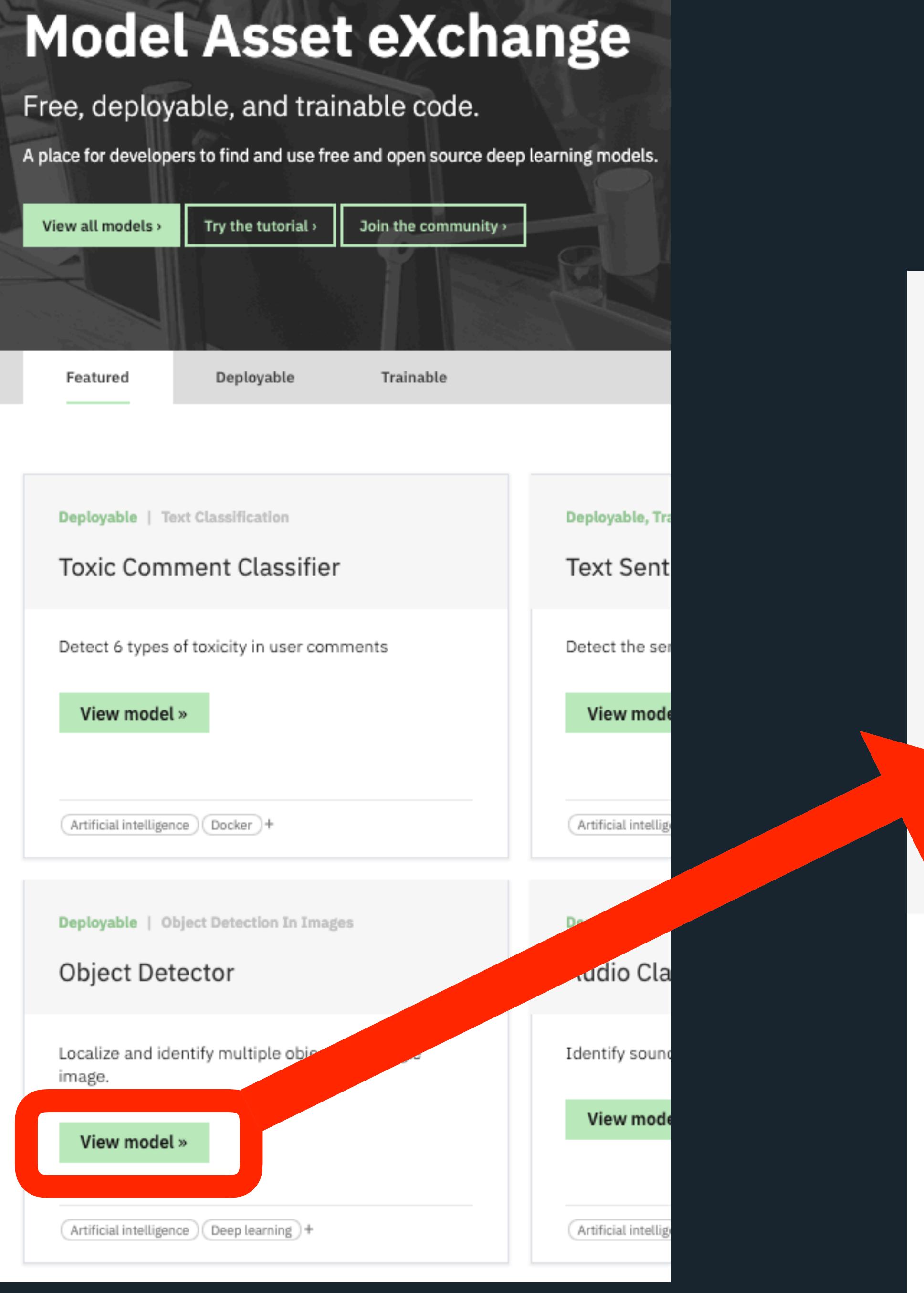
Artificial intelligence

Deep learning

Visual recognition

Image Classification

Object Detection in Images



ibm.biz/model-exchange

Object Detector

Localize and identify multiple objects in a single image.

[Get this model](#)

[Try the API](#)

[Try the web app](#)

[Try in a Node-RED flow](#)

By IBM Developer Staff | Updated September 21, 2018 - Published March 20, 2018

Artificial intelligence Deep learning Visual recognition Image Classification Object Detection in Images

Overview

This model recognizes the objects present in an image from the 80 different high-level classes. The model consists of a deep convolutional net base model for image feature extraction followed by additional convolutional layers specialized for the task of object detection, that was trained on the COCO Dataset. The input to the model is an image, and the output is a list of estimated class probabilities for the objects detected in the image.

The model is based on the [SSD Mobilenet V1 object detection model for TensorFlow](#).

Model Metadata

Domain	Application	Industry	Framework	Training Data
Vision	Object Detection	General	TensorFlow	COCO Dataset

Code

Issues 1

Pull requests 1

Actions

Projects 0

Wiki

Security

Insights

Settings

Localize and identify multiple objects in a single image. <https://github.com/IBM/MAX-Object-Detector>

Edit

docker-image

machine-learning

machine-learning-models

coco-dataset

tensorflow-model

Manage topics

56 commits

3 branches

0 packages

3 releases

16 contributors

Apache-2.0

Branch: master

New pull request

Create new file

Upload files

Find File

Clone or download

SSaishruthi and MLnick Update to training readme (#67) ...

Latest commit da95aed 2 days ago

api

Add license headers (#55)

2 months ago

core

code cleanup (#64)

last month

docs

[ImgBot] Optimize images (#59)

2 months ago

protos

code cleanup (#64)

last month

samples

WML Training (#61)

4 days ago

tests

WML Training (#61)

4 days ago

training

Update to training readme (#67)

2 days ago

utils

code cleanup (#64)

last month

Deployment options

- Deploy from Docker Hub
- Deploy on Red Hat OpenShift
- Deploy on Kubernetes
- Run Locally

Access the API via Swagger

Deployable | Object Detection In Images

Object Detector

Localize and identify multiple objects in a single image.

Get this model Try the API Try the web app Try in a Node-RED flow

By IBM Developer Staff | Updated September 21, 2018 - Published March 20, 2018

Artificial intelligence Deep learning Visual recognition Image Classification Object Detection in Images

Overview

This model recognizes the objects present in an image from the 80 different high-level classes in the [COCO Dataset](#). The model consists of a deep convolutional net base model for image feature extraction, with additional convolutional layers specialized for the task of object detection, that was trained on a large dataset. The input to the model is an image, and the output is a list of estimated class probabilities for each object detected in the image. The model is based on the [SSD Mobilenet V1 object detection model](#) for TensorFlow.

Model Metadata

Domain	Application	Industry	Framework	Training Data
Vision	Object Detection	General	TensorFlow	COCO Dataset

MAX Object Detector 1.2.0

Base URL: /
<http://max-object-detector.max.us-south.containers.appdomain.cloud/swagger.json>

Localize and identify multiple objects in a single image.

model Model information and inference operations

GET /model/labels Return the list of labels that can be predicted by the model

GET /model/metadata Return the metadata associated with the model

POST /model/predict Make a prediction given input data

POST /model/predict Make a prediction given input data

Parameters

Name	Description
image * required	An image file (encoded as PNG or JPG/JPEG) File (formData)
threshold number (query)	Probability threshold for including a detected object in the response in the range [0, 1] (default: 0.7). Lowering the threshold includes objects the model is less certain about.

Responses

Response content type application/json

Curl

```
curl -X POST 'http://max-object-detector.max.us-south.containers.appdomain.cloud/model/predict?threshold=0.7' -H 'accept: application/json' -H 'Content-Type: multipart/form-data' -F 'image=@kid-dog.jpeg;type=image/jpeg'
```

Request URL

<http://max-object-detector.max.us-south.containers.appdomain.cloud/model/predict?threshold=0.7>

Server response

Code	Details
200	Response body

```
{
  "status": "ok",
  "predictions": [
    {
      "label_id": "1",
      "label": "person",
      "probability": 0.8891462087631225,
      "detection_box": [
        0.04557418132421375,
        0.07023211918296814,
        0.8919000625610352,
        0.6151688098907471
      ]
    },
    {
      "label_id": "18",
      "label": "dog",
      "probability": 0.8122376799583435,
      "detection_box": [
        0.27157526326179504,
        0.5359986295700073,
        0.4154074960418701,
        0.7918230295181271
      ]
    }
  ]
}
```

Access the API via Python

Deployable | Object Detection In Images

Object Detector

Localize and identify multiple objects in a single image.

Get this model

Try the API

Try the web app

Try in a Node-RED flow

Try in CodePen

```
# Model
url = 'http://max-object-detector.max.us-south.containers.appdomain.cloud/'
model_endpoint = 'model/predict'
complete_url = url + model_endpoint

# Upload an image to the MAX model's rest API
path_to_input_image = 'baby-bear.jpg'

with open(path_to_input_image, 'rb') as file:
    file_form = {'image': (path_to_input_image, file, 'image/jpeg')}
    # Post the image to the rest API using the requests library
    r = requests.post(url=complete_url, files=file_form)
    # Return the JSON
    response = r.json()
```

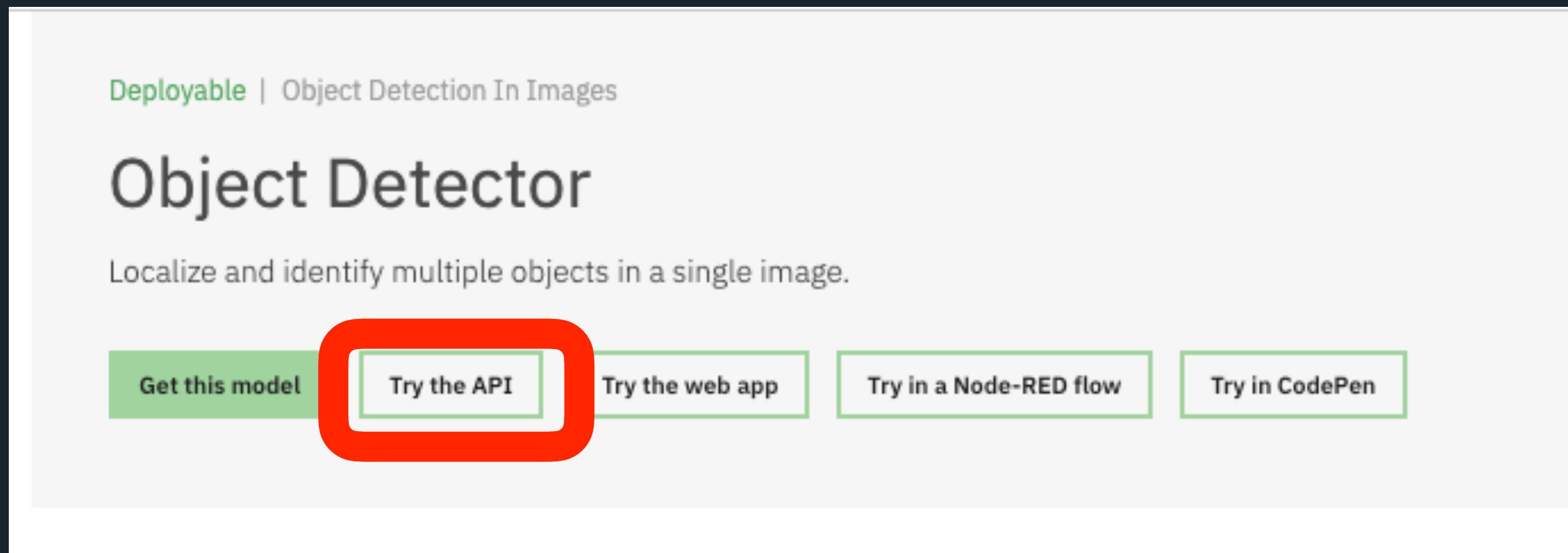
Try yourself here:
<http://ibm.biz/max-notebook>

response

```
{'status': 'ok',
'predictions': [ {'label_id': '88',
'label': 'teddy bear',
'probability': 0.9896332025527954,
'detection_box': [ 0.27832502126693726,
0.5611844062805176,
0.643224835395813,
0.8432191610336304 ] },
{'label_id': '1',
'label': 'person',
'probability': 0.9879012107849121,
'detection_box': [ 0.24251867830753326,
0.26926860213279724,
0.655893087387085,
0.5768759250640869 ] } ] }
```

Access the API Via R

slides: bit.ly/max-dsgo



```
library(dplyr)
library(httr)

# Endpoint
endpoint <- 'http://max-object-detector.max.us-south.containers.appdomain.cloud/'
# endpoint <- 'http://localhost:5000' # if running docker locally or docker hub

object_detector <- function(path_to_img, endpoint) {
  model_endpoint <- paste0(endpoint, 'model/predict') # Model endpoint
  # POST
  response <- httr::POST(model_endpoint,
                           body = list(image = upload_file(path_to_img,
                                                             type = "image/jpeg")),
                           encode = c("multipart"))
  ) %>% content()
  response$predictions
}

object_detector("imgs/baby-bear.jpg", endpoint)
```

Access the API Via Web App

Deployable | Object Detection In Images

Object Detector

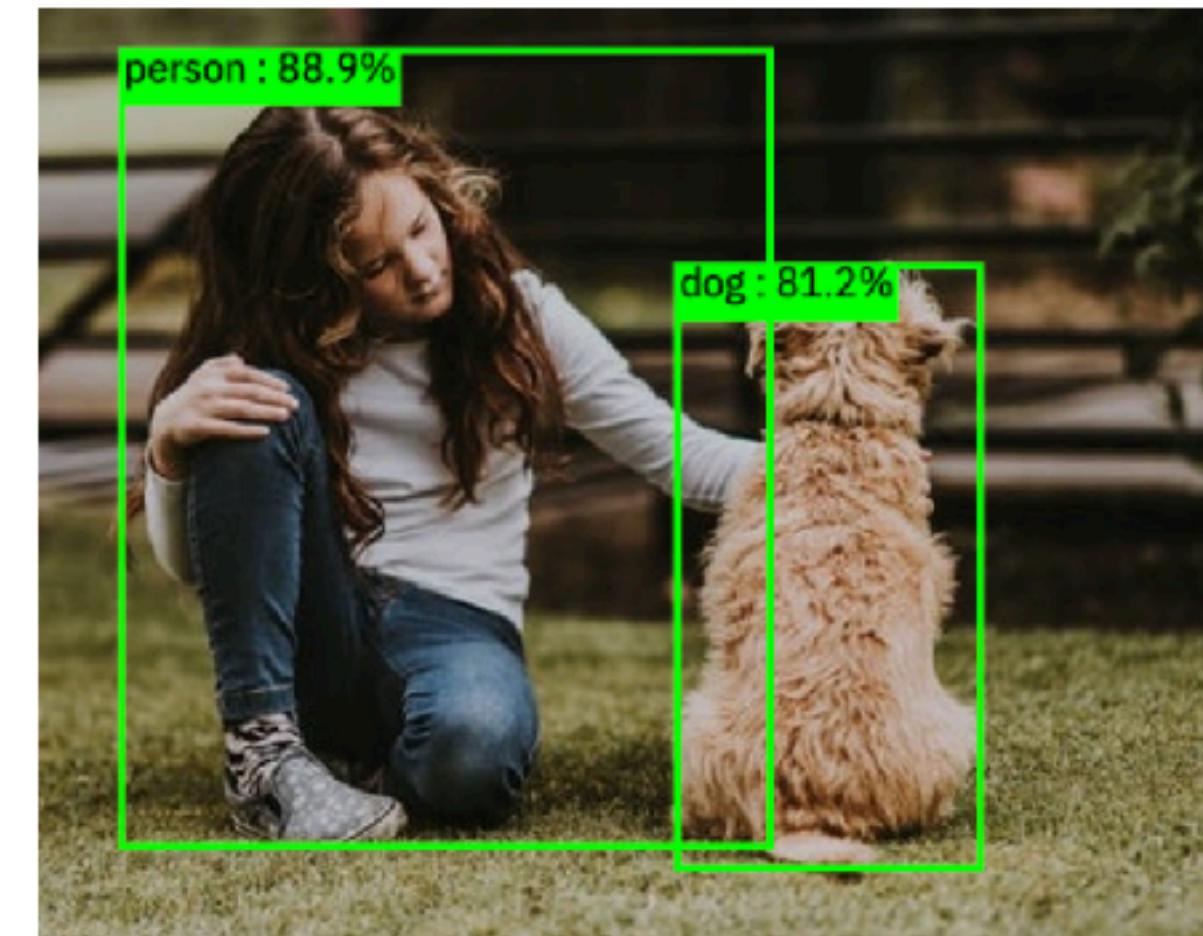
Localize and identify multiple objects in a single image.

[Get this model](#) [Try the API](#) [Try the web app](#) [Try in a Node-RED flow](#) [Try in CodePen](#)

By IBM Developer Staff | Updated September 21, 2018 - Published Ma

Artificial intelligence Deep learning Visual recognition Image Classifica

MAX Object Detector



Overview

This model recognizes the objects present in an image from the [COCO Dataset](#). The model consists of a deep convolutional neural network with additional convolutional layers specialized for the task of object detection. The input to the model is an image, and the output is a list of detected objects in the image. The model is based on the [SSD Mobilenet](#) architecture.

Model Metadata

Domain	Application	Industry	Framework
Vision	Object Detection	General	TensorFlow

Try yourself here:
ibm.biz/object-detector-webapp

Upload an image

Choose File No file chosen

Submit

Filter detected objects ⓘ

Probability Threshold: 70%

Labels Found ⓘ

person dog

Access the API Via Node-RED flow

Deployable | Object Detection In Images

Object Detector

Localize and identify multiple objects in a single image.

Get this model

Try the API

Try the web app

Try in a Node-RED flow

Try in CodePen

By IBM Developer Staff | Updated September 21, 2018 - Published March 20, 2018

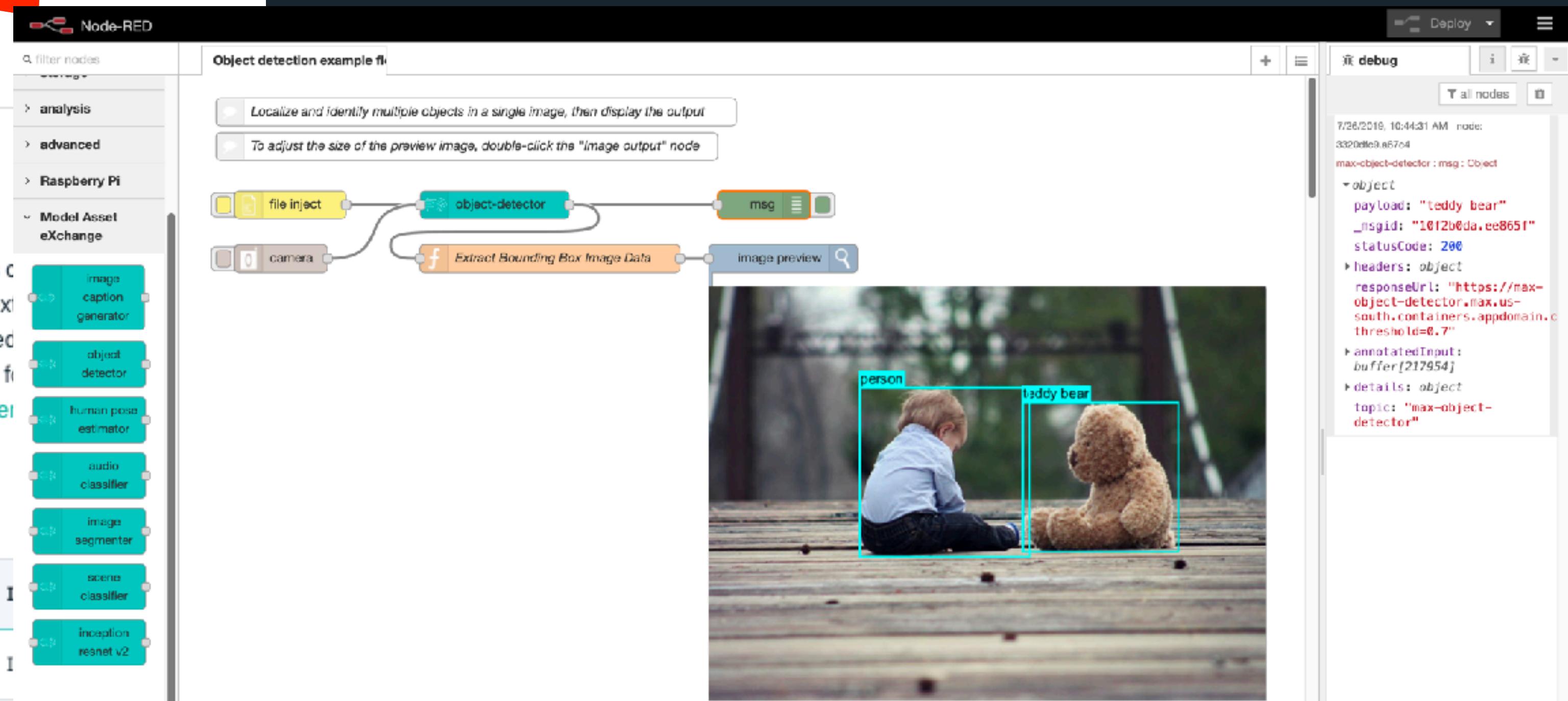
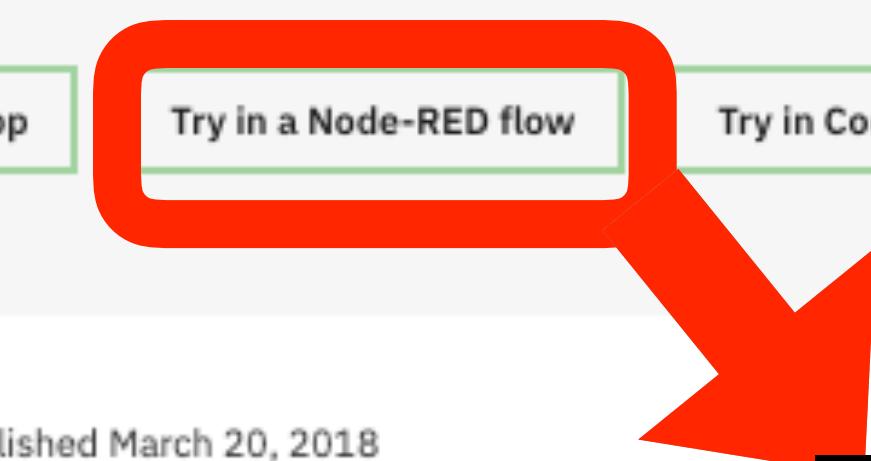
Artificial intelligence Deep learning Visual recognition Image Classification Object Detection in Images

Overview

This model recognizes the objects present in an image from the 80 different high-level classes contained in the [COCO Dataset](#). The model consists of a deep convolutional net base model for image feature extraction, with additional convolutional layers specialized for the task of object detection, that was trained on the COCO dataset. The input to the model is an image, and the output is a list of estimated class probabilities for each object detected in the image. The model is based on the [SSD Mobilenet V1 object detection model for TensorFlow](#).

Model Metadata

Domain	Application	Industry	Framework	Training Data	Tags
Vision	Object Detection	General	TensorFlow	COCO Dataset	I



Access the API Via CodePen

Deployable | Object Detection In Images

Object Detector

Localize and identify multiple objects in a single image.

[Get this model](#) [Try the API](#) [Try the web app](#) [Try in a Node-RED flow](#) [Try in CodePen](#)

By IBM Developer Staff | Updated September 21, 2018 - Published March 20, 2018

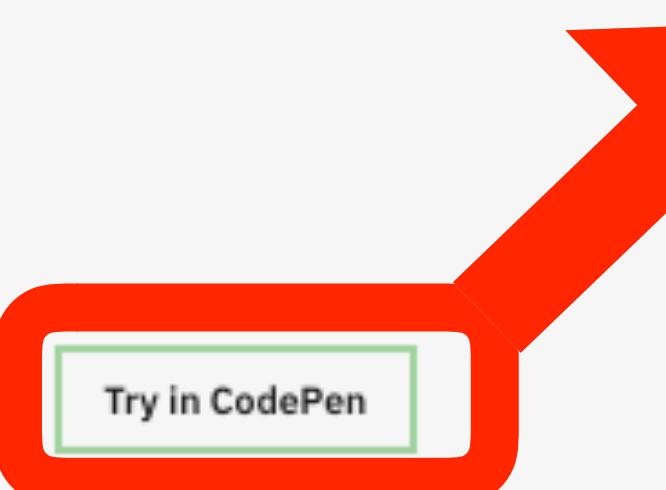
Artificial intelligence Deep learning Visual recognition Image Classification Object Detection in Images

Overview

This model recognizes the objects present in an image from the 80 different high-level classes of objects in the [COCO Dataset](#). The model consists of a deep convolutional net base model for image feature extraction, together with additional convolutional layers specialized for the task of object detection, that was trained on the COCO data set. The input to the model is an image, and the output is a list of estimated class probabilities for the objects detected in the image. The model is based on the [SSD Mobilenet V1 object detection model for TensorFlow](#).

Model Metadata

Domain	Application	Industry	Framework	Training Data	Input Data Format
Vision	Object Detection	General	TensorFlow	COCO Dataset	Image (RGB/HWC)



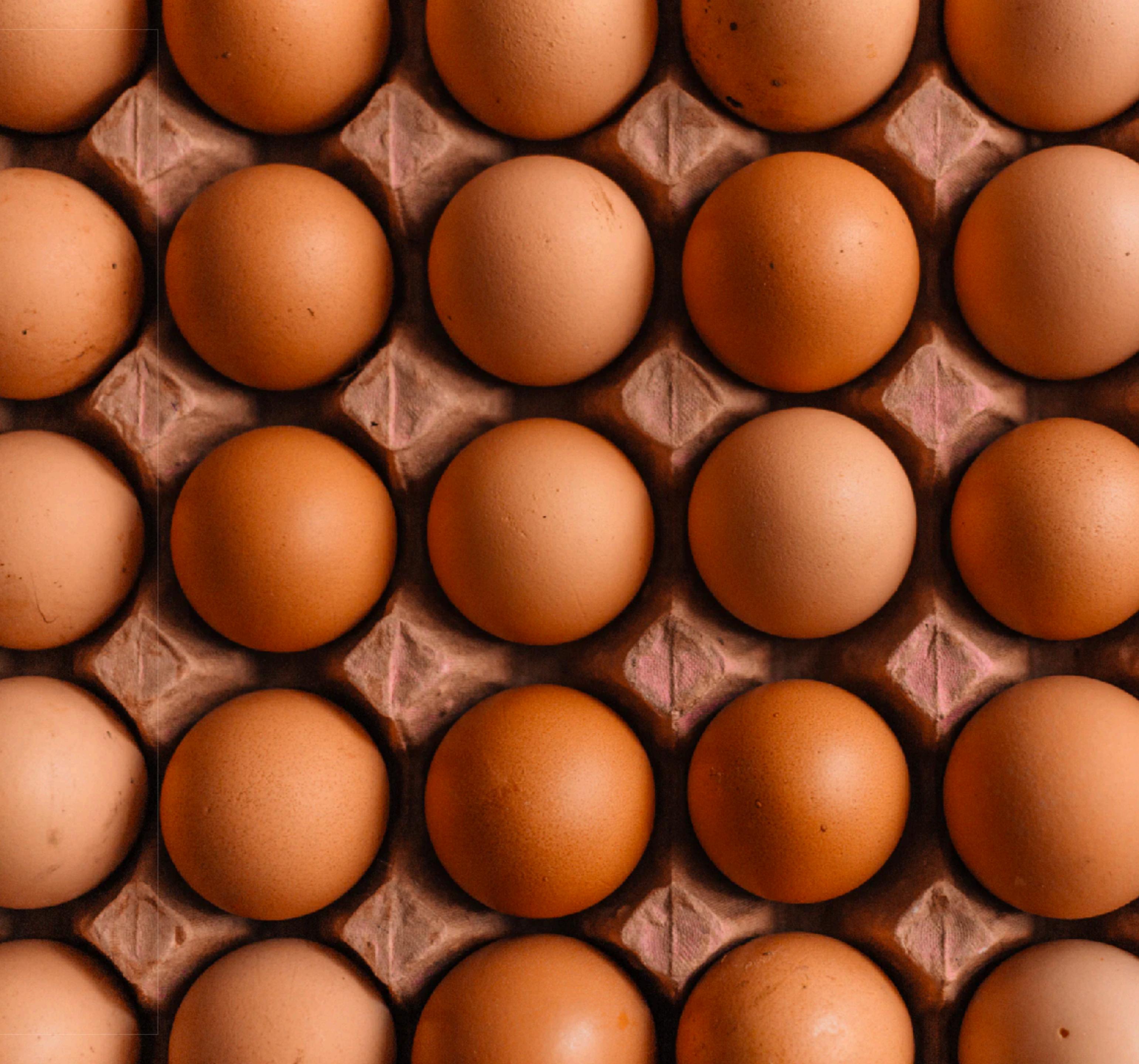
The screenshot shows two views of the MAX image models API. The top view is a CodePen interface titled 'MAX image models A PEN BY IBM CODAIT'. It displays three tabs: HTML, CSS, and JS. The JS tab contains the following code:

```
1 let webcamStream
2 let video
3 let prediction
4 let inputImage
5 let predictionLabels
6
7 const constraints = {
8   video: true,
9   audio: false
10 }
11
12 const modelEndpoint = {
13   objectDetector: 'https://max-object-detector.max.us-south.containers.appdomain.cloud/model/pr'
```

The bottom view is a web-based interface titled 'MAX Image Models'. It shows a dropdown for 'Model' set to 'Object Detector' and an 'Endpoint' field containing the URL 'https://max-object-detector.max.us-south.containers.appdomain.cloud/model/predict'. Below this is a file input field labeled 'Select an image...' with a placeholder image of a person petting a dog. To the right, a JSON response is displayed:

```
{
  "status": "ok",
  "predictions": [
    {
      "label_id": "1",
      "label": "person",
      "probability": 0.8891462087631226,
      "detection_box": [
        0.045574188232421875,
        0.07823271918296814,
        0.8919000625610352,
        0.6151688098907471
      ]
    },
    {
      "label_id": "18",
      "label": "dog",
      "probability": 0.8122376799583435,
      "detection_box": [
        0.27457526326179504,
        0.5369986295700073,
        0.9154078968418781,
        0.7918238295181274
      ]
    }
  ]
}
```

All this in a
standardized way



Licenses

Component	License
Model GitHub Repository	Apache 2.0
Model Weights	Apache 2.0
Model Code (3rd party)	Apache 2.0
Test Assets	CC0

Overview

This model recognizes the objects present in an image from the 80 different high-level classes of objects in the [COCO Dataset](#). The model consists of a deep convolutional net base model for image feature extraction, together with additional convolutional layers specialized for the task of object detection, that was trained on the COCO data set. The input to the model is an image, and the output is a list of estimated class probabilities for the objects detected in the image. The model is based on the [SSD Mobilenet V1 object detection model for TensorFlow](#).

References

- *J. Huang, V. Rathod, C. Sun, M. Zhu, A. Korattikara, A. Fathi, I. Fischer, Z. Wojna, Y. Song, S. Guadarrama, K. Murphy, “Speed/accuracy trade-offs for modern convolutional object detectors”, CVPR 2017*
- *Tsung-Yi Lin, M. Maire, S. Belongie, L. Bourdev, R. Girshick, J. Hays, P. Perona, D. Ramanan, C. Lawrence Zitnick, P. Dollár, “Microsoft COCO: Common Objects in Context”, arXiv 2015*
- *W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C. Fu, A. C. Berg, “SSD: Single Shot MultiBox Detector”, CoRR (abs/1512.02325), 2016*
- *A.G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, H. Adam, “MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications”, arXiv 2017*
- [TensorFlow Object Detection GitHub Repo](#)

Model Metadata

Domain	Application	Industry	Framework	Training Data	Input Data Format
Vision	Object Detection	General	TensorFlow	COCO Dataset	Image (RGB/HWC)

BEHIND THE SCENES

Find* a state-of-art open source deep learning model specific to domain

Validate license terms

Perform model health check & code clean up

Wrap models in MAX framework and provide REST API

Publish the deployable model as Docker images on Docker Hub

Use the MAX training framework to create an image for custom model training

Review and Continuous Integration

* or build from scratch

And if you are
feeling
adventurous...



You can train your model using your own data



Object Detector

Localize and identify multiple objects in a single image.

[Get this model](#) [Try the API](#) [Try the web app](#) [Try in a Node-RED flow](#) [Try in CodePen](#)

By IBM Developer Staff | Updated September 21, 2018 - Published March 20, 2018

Artificial intelligence Deep learning Visual recognition Image Classification Object Detection in Images

Overview

This model recognizes the objects present in an image from the 80 different high-level classes. The [Dataset](#) consists of a deep convolutional net base model for image feature extraction and additional convolutional layers specialized for the task of object detection, that was trained on the COCO dataset. The input to the model is an image, and the output is a list of estimated class probabilities for the objects in the image. The model is based on the [SSD Mobilenet V1 object detection model for TensorFlow](#).

Model Metadata

Domain	Application	Industry	Framework	Training Data
Vision	Object Detection	General	TensorFlow	COCO Dataset

IBM / MAX-Object-Detector

Code Issues 1 Pull requests 1 Actions Projects 0 Wiki Security Insights Settings

Localize and identify multiple objects in a single image. <https://github.com/IBM/MAX-Object-Detector>

docker-image machine-learning machine-learning-models coco-dataset tensorflow-model Manage topics

56 commits 3 branches 0 packages 3 releases 16 contributors Apache-2.0

Branch: master New pull request Create new file Upload files Find File Clone or download

SSaishruthi and MLnick Update to training readme (#67) ... ✓ Latest commit da95aed 2 days ago

api Add license headers (#55) 2 months ago

core code cleanup (#64) last month

docs [ImgBot] Optimize images (#59) 2 months ago

protos code cleanup (#64) last month

samples WML Training (#61) 4 days ago

tests WML Training (#61) 4 days ago

training Update to training readme (#67) 2 days ago

utils code cleanup (#64) last month



How to Train Object Detector Model Using Your Own Data

- [Collect Data for Training](#)
- [Train the Model](#)
- [Rebuild the Model-Serving Microservice](#)

Collect Data for Training

Collect RGB images encoded as jpeg or png containing objects that need to be detected. Make sure the training images have large variations in angle, resolution, lighting and background so that they generalize well with the test data. Use a reasonably large number of images per class to provide better results.

Train the Model

- [Install Local Prerequisites](#)
- [Run the Setup Script](#)
- [Prepare Data for Training](#)
- [Customize Training](#)
- [Train the Model Using Watson Machine Learning](#)



How do I get started?

bit.ly/max-tutorial

SERIES

Learning Path: An introduction to the Model Asset Exchange

Learn how to use state-of-the-art deep learning models in your applications or services

Objectives

Upon completion of this learning path, you will be able to:

- Find and explore deployable and trainable deep learning models on the exchange
- Deploy a model-serving microservice on Docker
- Deploy a model-serving microservice on the Red Hat OpenShift container platform
- Consume the microservice from a Node.js or Python web application
- Consume the microservice from a Node-RED flow
- [Consume the microservice in a web browser using JavaScript](#)
- Complete the sample [code patterns for the Model Asset Exchange](#)

CODE PATTERN | APR 22, 2019

Use your arms to make music

[Get the Code »](#)

(Artificial intelligence) (Cloud) +

CODE PATTERN | MAR 28, 2019

Create a web app to interact with machine learning generated image captions

[Get the Code »](#)

(Artificial intelligence) (Docker) +

CODE PATTERN | MAR 28, 2019

Create a web app to visually interact with objects detected using machine learning

[Get the Code »](#)

(Artificial intelligence) (Deep learning) +

CODE PATTERN | MAR 28, 2019

Deploy a deep learning-powered 'Magic cropping tool'

[Get the Code »](#)

(Artificial intelligence) (Data science) +

CODE PATTERN | JUL 13, 2018

Train and evaluate an audio classifier

[Get the Code »](#)

(Artificial intelligence) (Data science) +

Code Patterns

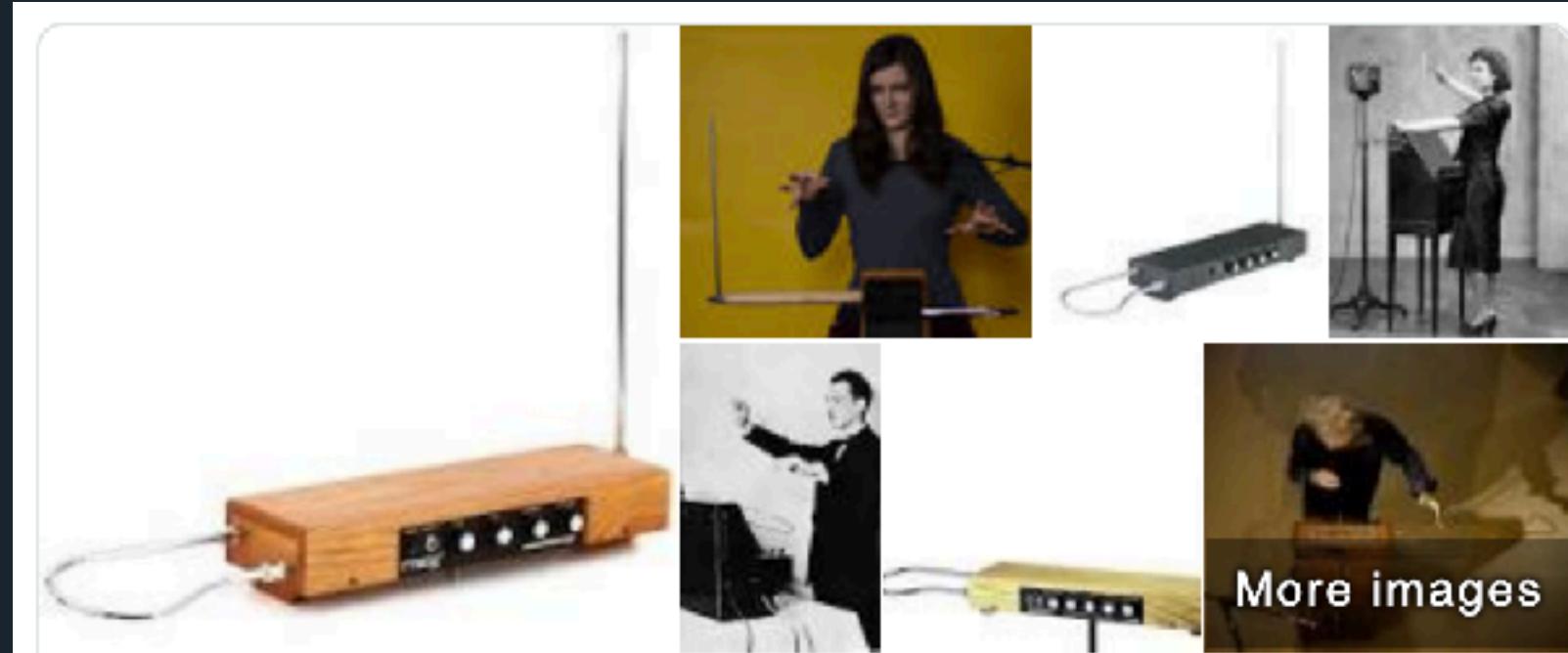
How to easily consume MAX models

ibm.biz/max-code-patterns

Use your arms to make music

Create music following your arm movements using

Model Asset eXchange (MAX) **human pose estimator** model and **TensorFlow**



Theremin
Musical instrument

The theremin is an electronic musical instrument controlled without physical contact by the thereminist. It is named after its inventor, Léon Theremin, who patented the device in 1928. [Wikipedia](#)

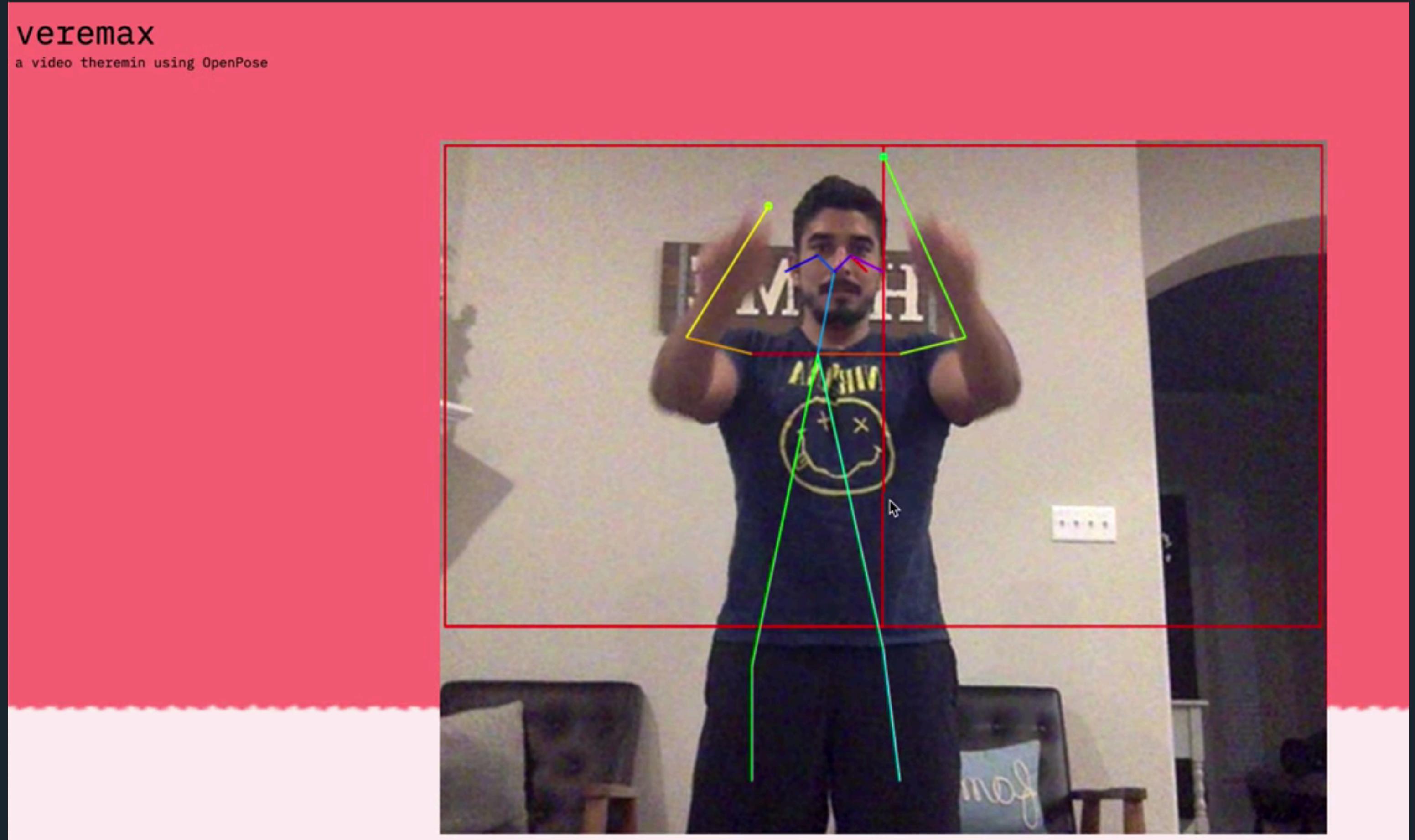
Instrument family: [Electronic Musical Instruments](#), [Musical Keyboards](#)

Invented: 1920

Related instrument: [Ondes Martenot](#), [Electro-Theremin](#)

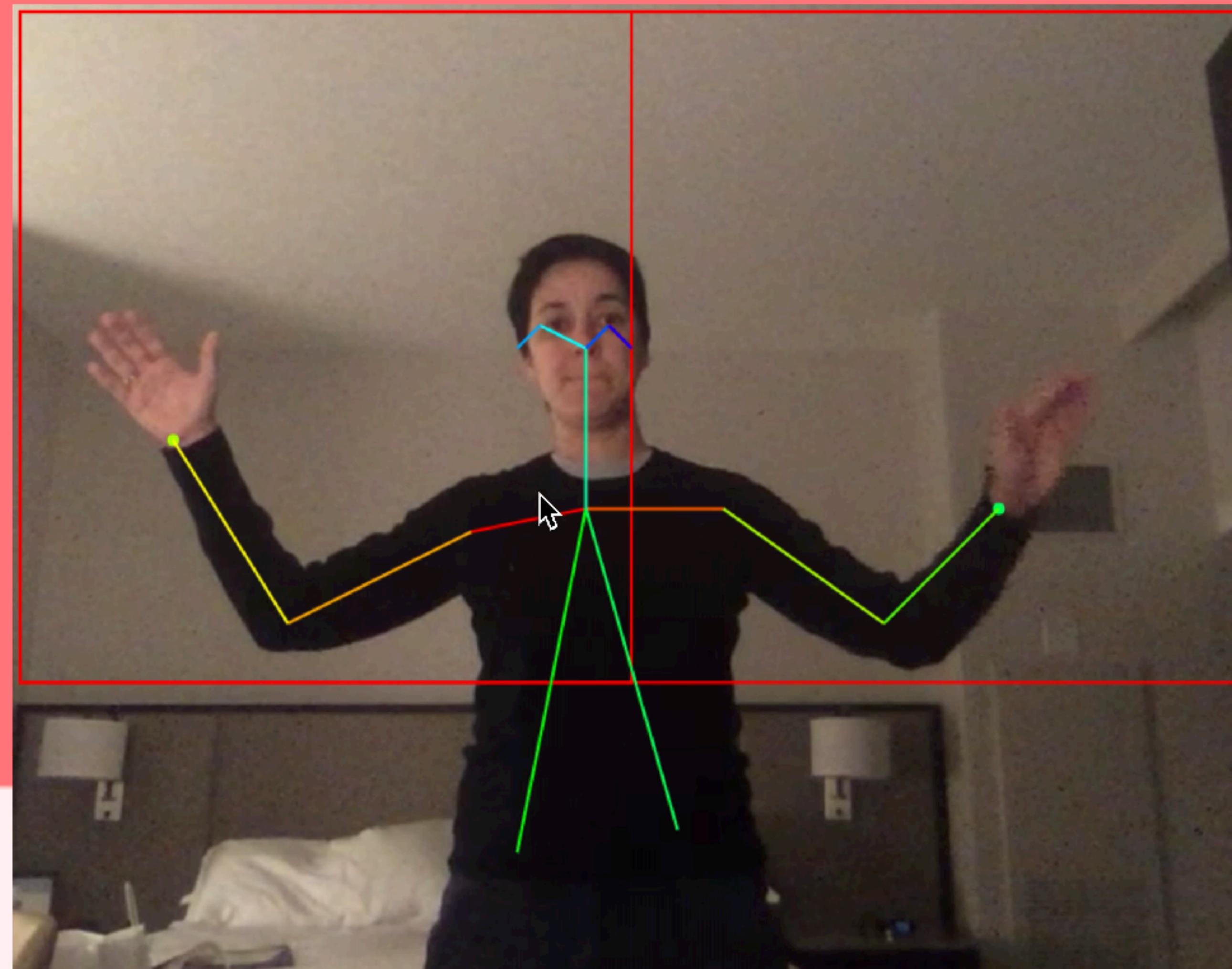
Inventor: [Léon Theremin](#)

Hornbostel-Sachs classification: 531.1; (Electrophone)



veremax

a video theremin using OpenPose





FACES
OF
OPEN
SOURCE

Photo by Peter Adams -
<http://www.facesofopensource.com/>

Thank you!

→ K-ROZ.COM
🐦 @GDEQUEIROZ



ai-inclusive.org

ibm.biz/model-exchange