We'll be starting shortly!

To help us run the workshop smoothly, please kindly:

- Submit all questions using the Q&A function
- If you have an urgent request, please use the "Raise Hand" function

Thank you!

Exclusive Trainings by Zenika and Trent!

https://bit.ly/CodeMasterClass

Shopee CODE LEAGUE 2020

Introduction to TensorFlow

Harry POMMIER
Data Science Consultant @Zenika France, Nantes

zenika
<led by passion>

TRENT
GLOBAL COLLEGE

Trent Global College

- Imda Funded Tech Bootcamp

  http://bit.ly/codingdip

Supported By:

Our Graduates

Sam Tan
Data Analyst
SISTIC Singapore

Lee Tong Lin
Junior Software Developer
Love, Bonito

Jimmy Hsu
IT Engineer
Singtel

Lynette Lim
Front-End Developer
Sparkline Pte Ltd

TRENT GLOBAL COLLEGE

Zenika

zenika

<led by passion>

- TECHNOLOGICAL AND ORGANIZATIONAL INNOVATION. BUILDING TOGETHER A SIMPLER, MORE OPEN, FREER AND ACCESSIBLE WORLD

- Zenika is specialized in software architecture and Agile methods with a threefold-expertise in consulting, realization and training.

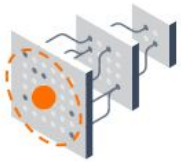
www.zenika.sg

www.zenika.com

What is TensorFlow?

TensorFlow

- Open source platform for machine learning (ML)
- Flexible ecosystem of tools, libraries and resources
- Build and deployment of ML powered applications
- Develop by Google Brain



Easy model building

Robust ML production anywhere

Powerful experimentation for research

https://www.tensorflow.org

TRENT GLOBAL COLLEGE

zenika
<led by passion>

Shopee CODE LEAGUE 2020

Some TensorFlow Use Cases

- Computer Vision
  - **Image Classification**
  - Object Detection
  - Activity Recognition
  - Image Reconstruction, Colorization

- Natural Language Processing (NLP)
  - Text Classification
  - Machine Translation
  - Sentiment Analysis
  - Speech Recognition

Prerequisite Concepts

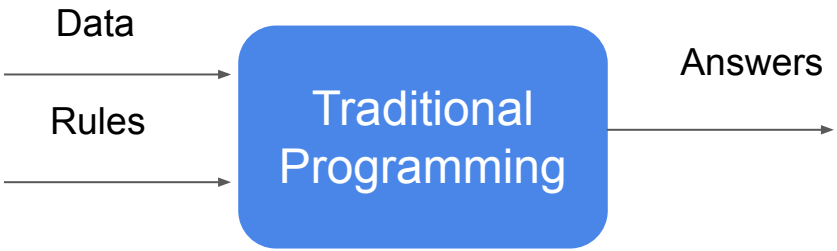
1. Machine Learning: a New Paradigm
2. Tensors
3. Computational Graphs

zenika
<led by passion>

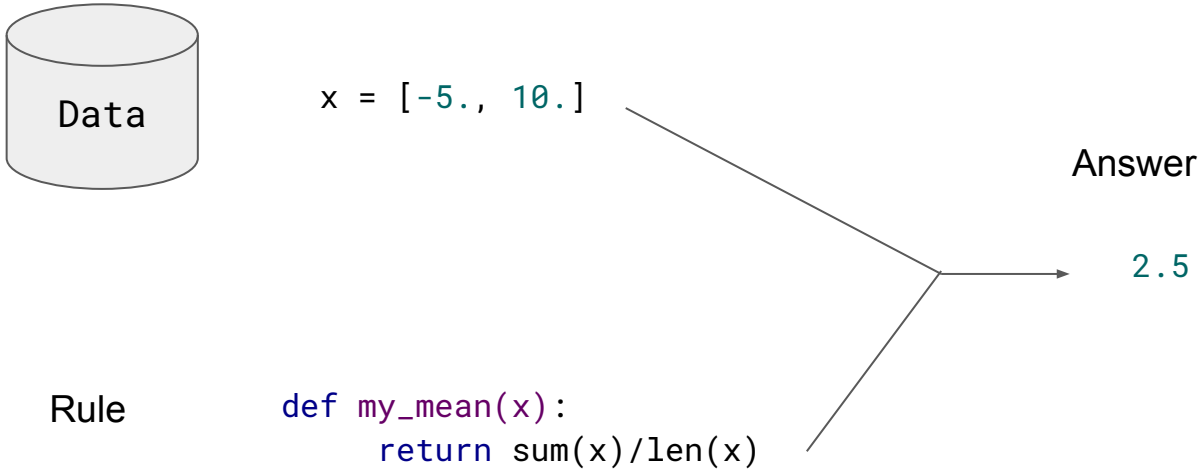
TRENT
GLOBAL COLLEGE

Shopee
CODE
LEAGUE
2020

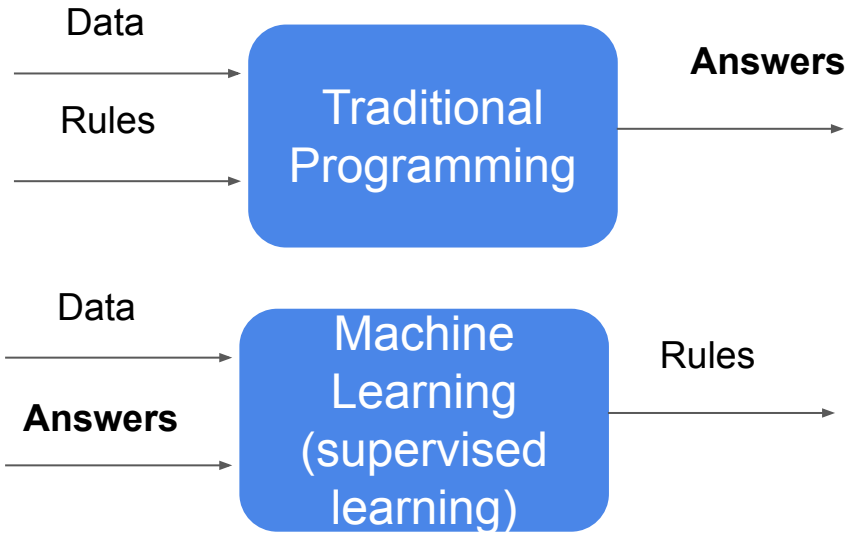
Prerequisite 1/3: Machine Learning Paradigm



Traditional Programming



A New Paradigm



zenika
<led by passion>

TRENT GLOBAL COLLEGE

Shopee CODE LEAGUE 2020

Why we need ML ?

You want to develop an **activity recognition** app for watersports (*e.g.* for a connected watch).
With **traditional programming:**

```
if depth == 0:
    activity =
"swimming"
else:
    activity =
"unknown"
```

```
if depth == 0:
    if speed > 4:
        activity =
"rowing"
    else:
        activity =
"swimming"
else:
    activity =
"unknown"
```

```
if depth == 0:
    if speed > 4:
        activity =
"rowing"
    else:
        activity =
"swimming"
elif depth < 0:
    activity =
"diving"
else:
    activity =
"unknown"
```

Surfing ?

Water polo ?

zenika
<led by passion>

TRENT GLOBAL COLLEGE

Shopee CODE LEAGUE 2020

You want to develop an **activity recognition** app for watersports (*e.g.* for a connected watch).
With **Machine Learning:**

```
[[ 7.1,  1.4],
 [-7.3, -1.8],
 [-7. , -6.9],
 [ 6.7, -3.2],
 [ 0.3,  6.5]]

label =
"swimming"
```

```
[[ 6.8, -2.6],
 [-8.1, -3.5],
 [-5.2, -1. ],
 [-1.6, -5.4],
 [-8.9, -9. ]]

label =
"rowing"
```

```
[[-1.5, 2.7],
 [ 7.1, -10. ],
 [-2.2, 1.4],
 [ 2.2, 9.8],
 [-2.6, 5.8]]

label =
"diving"
```

```
[[-1.2, -4.],
 [ 9.9, 3.1],
 [-0.8, 0.1],
 [-0.8, -6.6],
 [-6.3, 1.1]]

label =
"water polo"
```

```
[[-1.5, -8.7],
 [-3.8, -9.9],
 [-9.9, -5.8],
 [ 8.6, 7.6],
 [-7., -0.3]]

label =
"surfing"
```

zenika
<led by passion>

TRENT GLOBAL COLLEGE

Shopee CODE LEAGUE 2020

You want to develop an **activity recognition** app for watersports (*e.g.* for a connected watch).
With **Machine Learning:**

```
[[ 7.1,  1.4],
 [-7.3, -1.8],
 [-7. , -6.9],
 [ 6.7, -3.2],
 [ 0.3,  6.5]]

label =
"swimming"
```

```
[[ 6.8, -2.6],
 [-8.1, -3.5],
 [-5.2, -1. ],
 [-1.6, -5.4],
 [-8.9, -9. ]]

label =
"rowing"
```

```
[[-1.5, 2.7],
 [ 7.1, -10. ],
 [-2.2, 1.4],
 [ 2.2, 9.8],
 [-2.6, 5.8]]

label =
"diving"
```

```
[[-1.2, -4.],
 [ 9.9, 3.1],
 [-0.8, 0.1],
 [-0.8, -6.6],
 [-6.3, 1.1]]

label =
"water polo"
```

```
[[-1.5, -8.7],
 [-3.8, -9.9],
 [-9.9, -5.8],
 [ 8.6, 7.6],
 [-7., -0.3]]

label =
"surfing"
```

Stored in
Tensors

Dataset
Features + Labels
(Data + Answers)

a **feature** is an individual measurable property or characteristic
e.g. speed, depth, GPS position, time, pixel value

zenika
<led by passion>

TRENT GLOBAL COLLEGE

Shopee CODE LEAGUE 2020

Prerequisite 2/3: Tensors

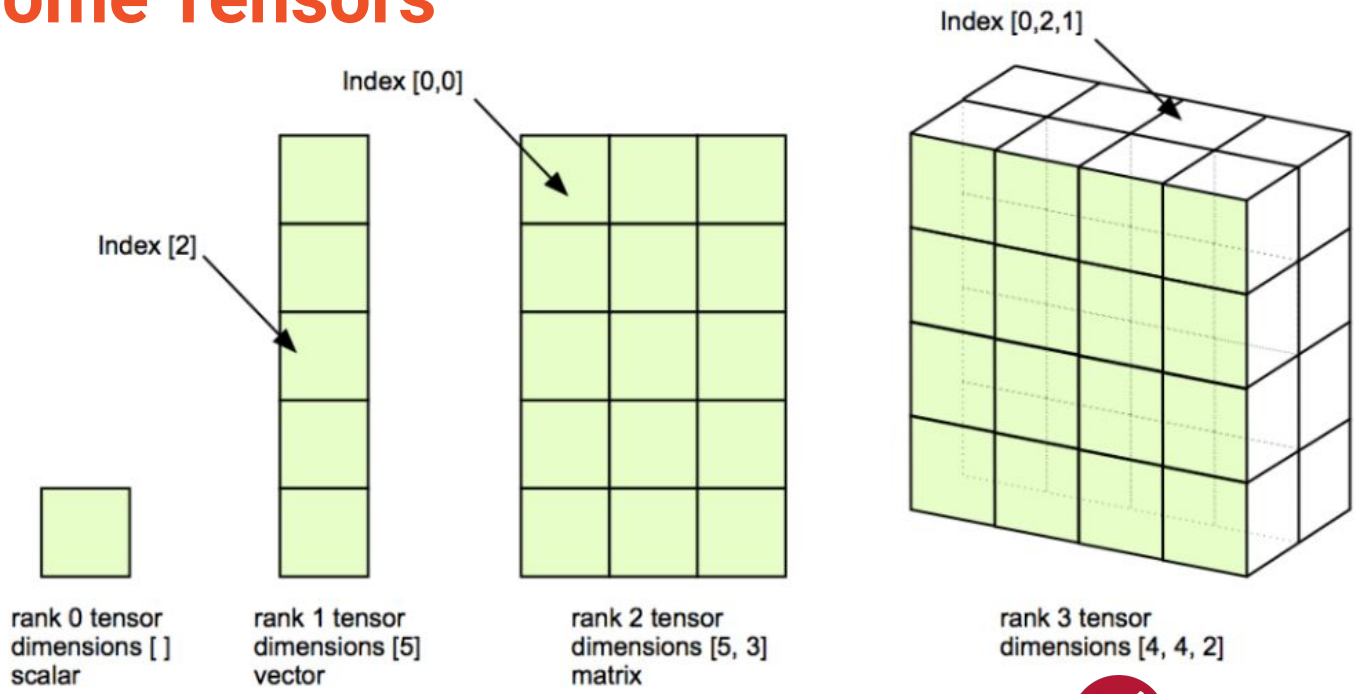
- Tensors are multidimensional arrays with a uniform type (float, string, boolean...)
- Can be seen as a generalization of scalars, vectors and matrices in N-dimentions, the dimension of a tensor is called the *rank*
  - scalar: rank = 0
  - vector: rank = 1
  - matrix: rank = 2
  - tensor: rank > 3
- Useful to represent data

zenika
<led by passion>

TRENT GLOBAL COLLEGE

Shopee CODE LEAGUE 2020

Some Tensors



rank 4
rank 5
...

shape

zenika
<led by passion>

TRENT
GLOBAL COLLEGE

Shopee CODE LEAGUE 2020

Some Tensors

e.g. Grayscale image

48 x 28 pixels

```
array([[255, 255, 128, ..., 128, 255, 255],
       [255, 255,  13, ...,  64, 255, 255],
       [255, 255,  13, ...,  64, 255, 255],
       ...,
       [255, 255, 255, ..., 255, 255, 255],
       [255, 255, 255, ..., 255, 255, 255],
       [255, 255, 255, ..., 255, 255, 255]], dtype=uint8)
```

rank 2 tensor, shape (48, 28)

zenika
<led by passion>

TRENT GLOBAL COLLEGE

Shopee CODE LEAGUE 2020

Some Tensors

e.g. RGB image

48 x 28 pixels

B
```
array([[236, 236, 128, ..., 128, 236, 236],
       [255, 255,  10, ...,  64, 255, 236],
```
G
```
array([[236, 236, 128, ..., 128, 236, 236],
       [255, 255,  11, ...,  64, 255, 236],
```
R
```
array([[255, 255, 128, ..., 128, 255, 255],
       [255, 255,  13, ...,  64, 255, 255],
       [255, 255,  13, ...,  64, 255, 255],
       ...,
       [255, 255, 255, ..., 255, 255, 255],
       [255, 255, 255, ..., 255, 255, 255],
       [255, 255, 255, ..., 255, 255, 255]], dtype=uint8)
```

rank 3 tensor, shape (48, 28, 3)

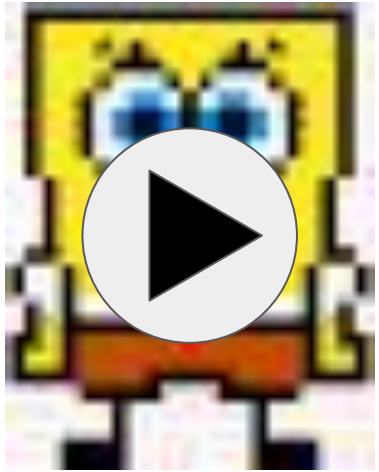
zenika
<led by passion>

TRENT GLOBAL COLLEGE

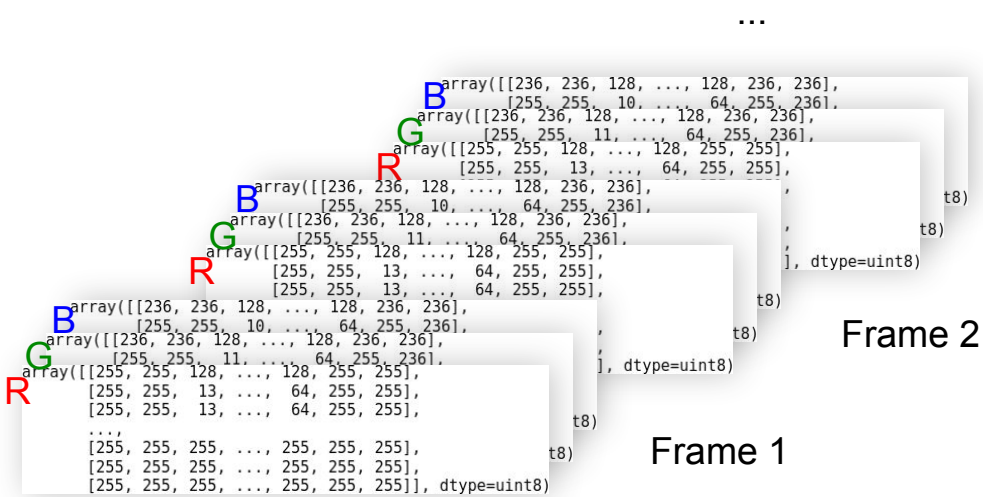
Shopee CODE LEAGUE 2020

Some Tensors

e.g. RGB video



48 x 28 pixels, 200 frames



rank 4 tensor, shape (200, 48, 28, 3)

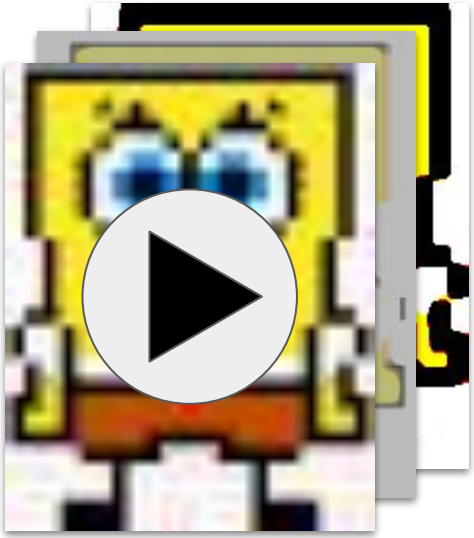
zenika
<led by passion>

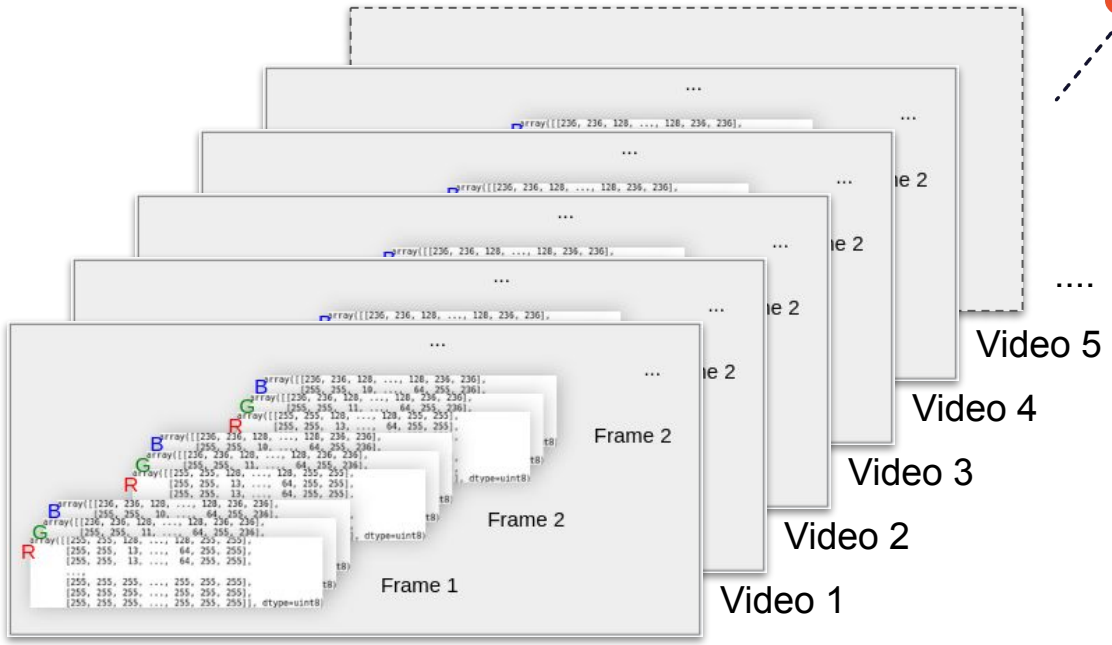
TRENT GLOBAL COLLEGE

Shopee CODE LEAGUE 2020

Some Tensors

e.g. Dataset of 10,000 videos





rank 5 tensor, shape (10000, 200, 48, 28, 3)

48 x 28 pixels, 200 frames, 10,000 examples

zenika
<led by passion>

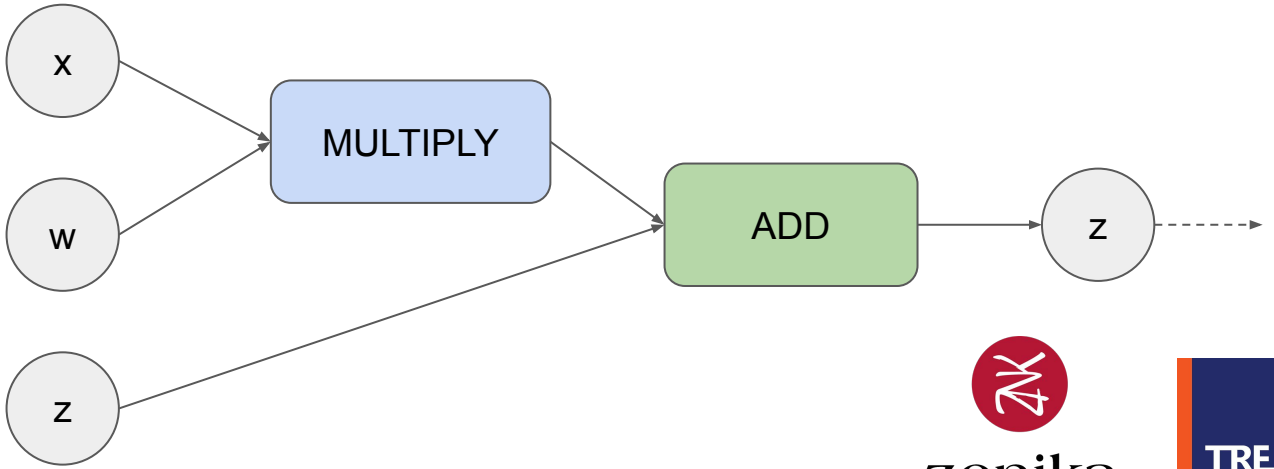
TRENT GLOBAL COLLEGE

Shopee CODE LEAGUE 2020

Prerequisite 3/3: Computational Graphs

- TensorFlow uses a **dataflow graph** to represent your computation in terms of the dependencies between individual operations

  Example: z = w*x + b



zenika
<led by passion>

TRENT
GLOBAL COLLEGE

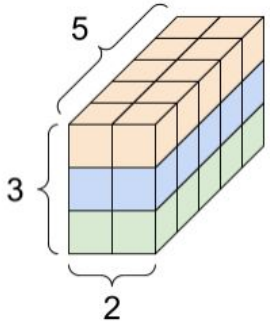
Shopee CODE LEAGUE 2020

Why Computational Graphs?

- **Parallelism**: easy for the system to identify operations that can execute in parallel.
- **Portability**: graph is defined in high level language, independent representation of the code in your model.
- **Compilation**: graph is compiled with TensorFlow's XLA (Accelerated Linear Algebra) compiler to generate faster code (*e.g.* fusing together adjacent operations, ...)
- **Distributed execution**: it is possible for TensorFlow to partition your program across multiple devices (CPUs, GPUs, and TPUs) attached to different machines.

TensorFlow Logo

Tensor



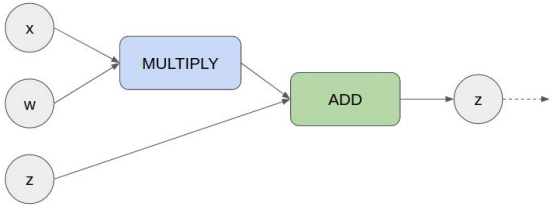

```
tf.Tensor(
[[[ 0  1  2  3  4]
  [ 5  6  7  8  9]]

 [[10 11 12 13 14]
  [15 16 17 18 19]]

 [[20 21 22 23 24]
  [25 26 27 28 29]]], shape=(3, 2, 5), dtype=int32)
```

TensorFlow

Flow



zenika
<led by passion>

TRENT
GLOBAL COLLEGE

Shopee CODE LEAGUE 2020

Simple Computation with TensorFlow 1

TF 1.x
(before 10/2019)

```
import tensorflow as tf
print("TensorFlow version: {}".format(tf.__version__))
```

```
TensorFlow version: 1.15.0
```

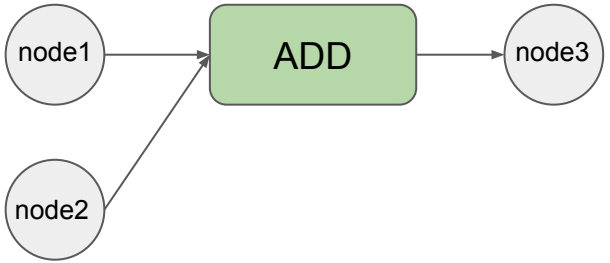
Construction phase

```
node1 = tf.constant([1, 2, 5])
node2 = tf.constant([0, 10, -5])
node3 = tf.add(node1, node2)
init = tf.global_variables_initializer()
```

Execution phase

```
with tf.Session() as sess:
    init.run()
    print(node3.eval())
```

```
[ 1 12  0]
```



zenika
<led by passion>

TRENT
GLOBAL COLLEGE

Shopee CODE LEAGUE 2020

Simple Computation with TensorFlow 2

TF 2.x
(since 10/2019)

```
import tensorflow as tf
print("TensorFlow version: {}".format(tf.__version__))
```

```
TensorFlow version: 2.2.0
```

```
node1 = tf.constant([1, 2, 5])
node2 = tf.constant([0, 10, -5])
node3 = tf.add(node1, node2)

node3.numpy() #Eager execution by default
```

```
array([ 1, 12,  0], dtype=int32)
```

- Makes it easy to get started with TensorFlow and debug models
- Reduces boilerplate code

zenika
<led by passion>

TRENT
GLOBAL COLLEGE

Shopee CODE LEAGUE 2020

First Neural Network with TensorFlow

Image Classification

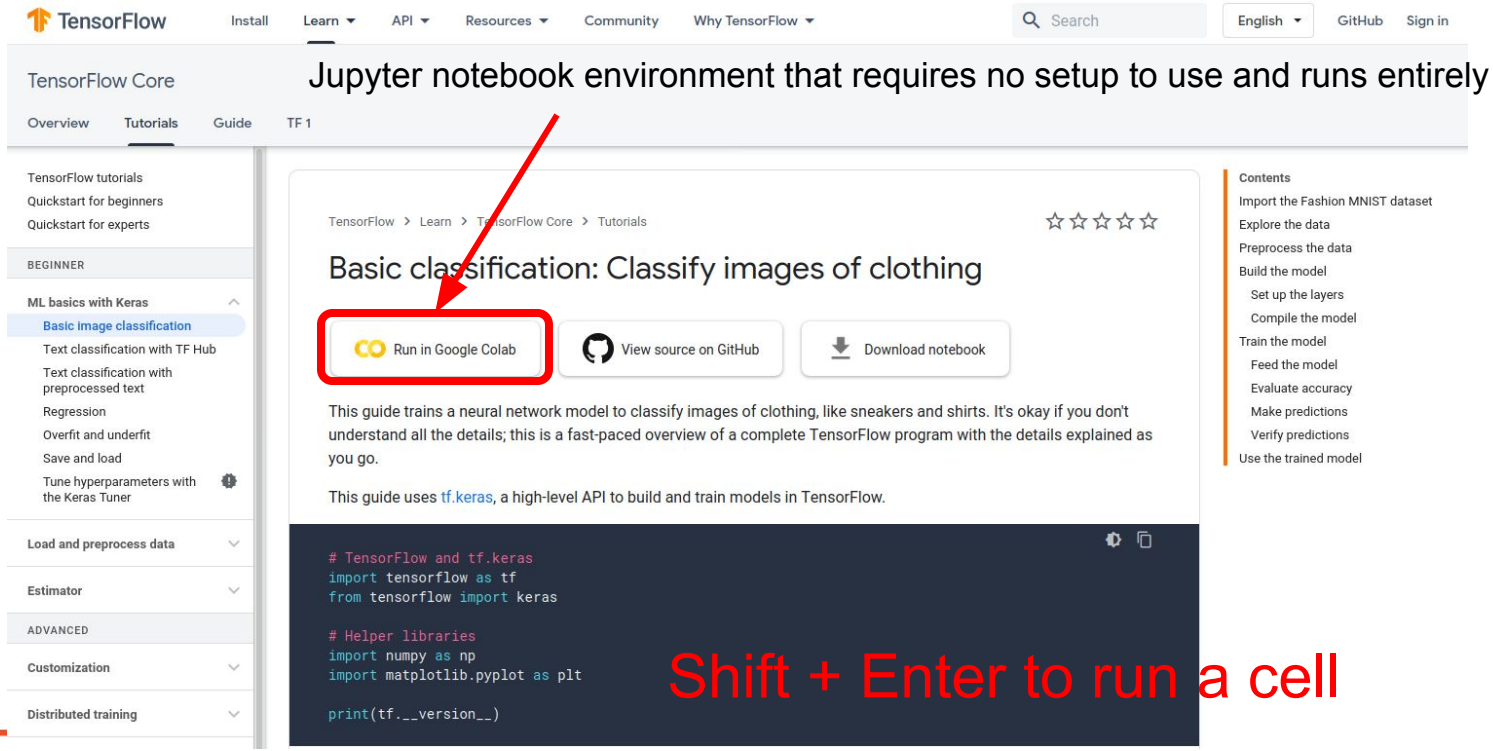
- https://www.tensorflow.org/tutorials/keras/classification
- **Keras**: high-level API for building and training deep learning models. It's used for fast prototyping.

zenika
<led by passion>

TRENT
GLOBAL COLLEGE

Shopee CODE LEAGUE 2020

Run in Google Colab https://www.tensorflow.org/tutorials/keras/classification



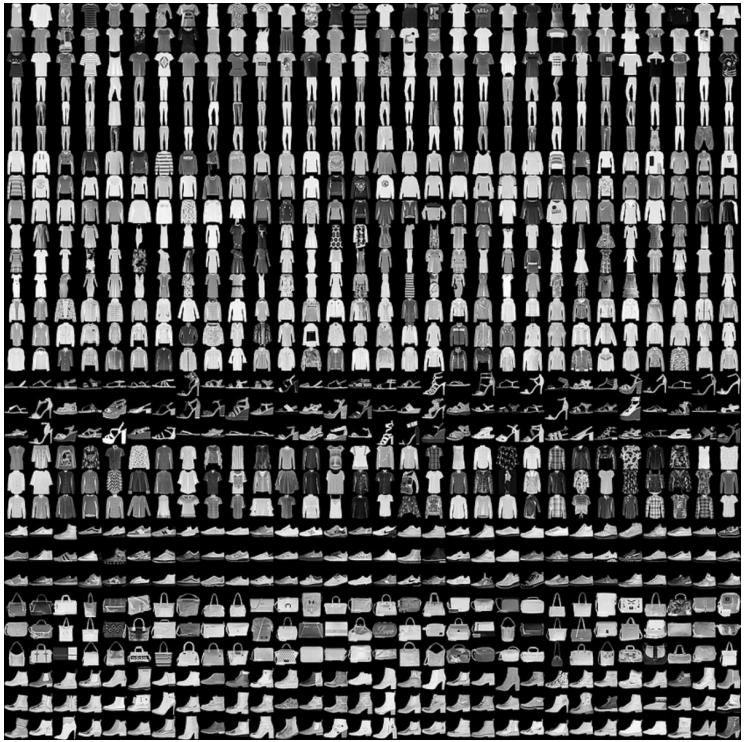
Jupyter notebook environment that requires no setup to use and runs entirely in the cloud

Shift + Enter to run a cell

Fashion MNIST

- 70 k grayscale images
- 28 x 28 pixels
- 10 categories/classes

Label	Class
0	T-shirt/top
1	Trouser
2	Pullover
3	Dress
4	Coat
5	Sandal
6	Shirt
7	Sneaker
8	Bag
9	Ankle boot

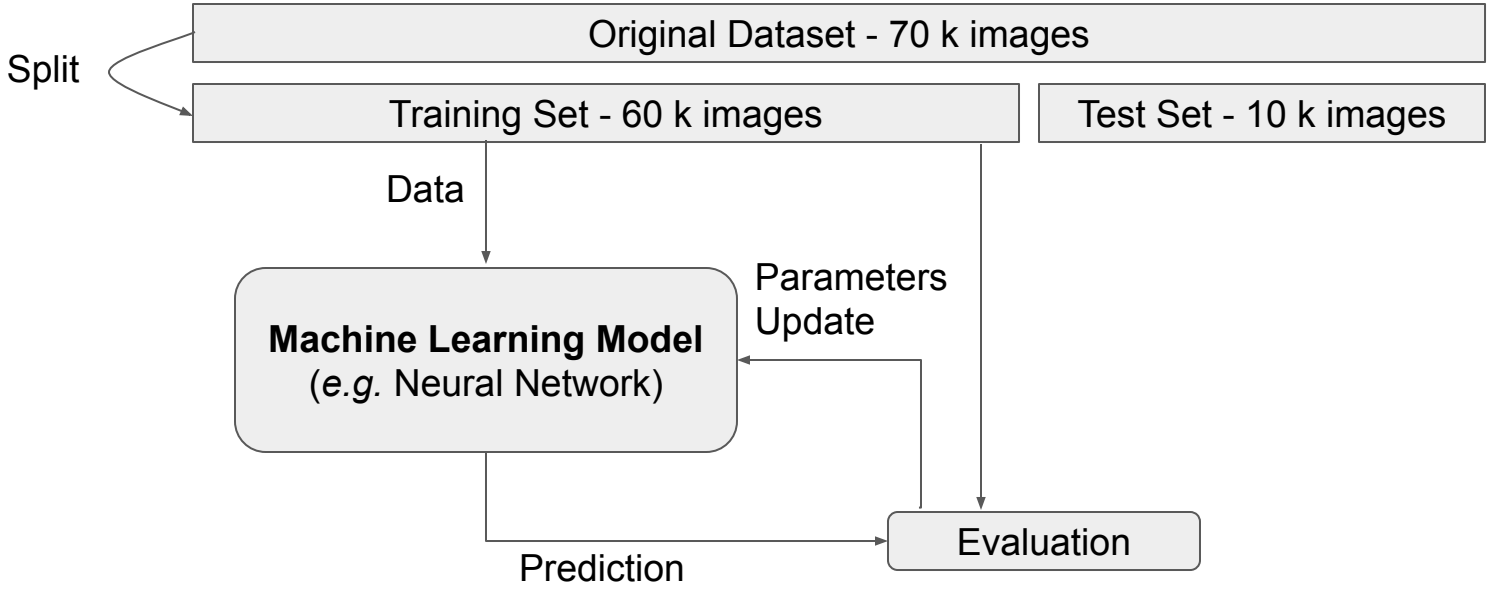


zenika
<led by passion>

TRENT
GLOBAL COLLEGE

Shopee CODE LEAGUE 2020

Basic Model Training

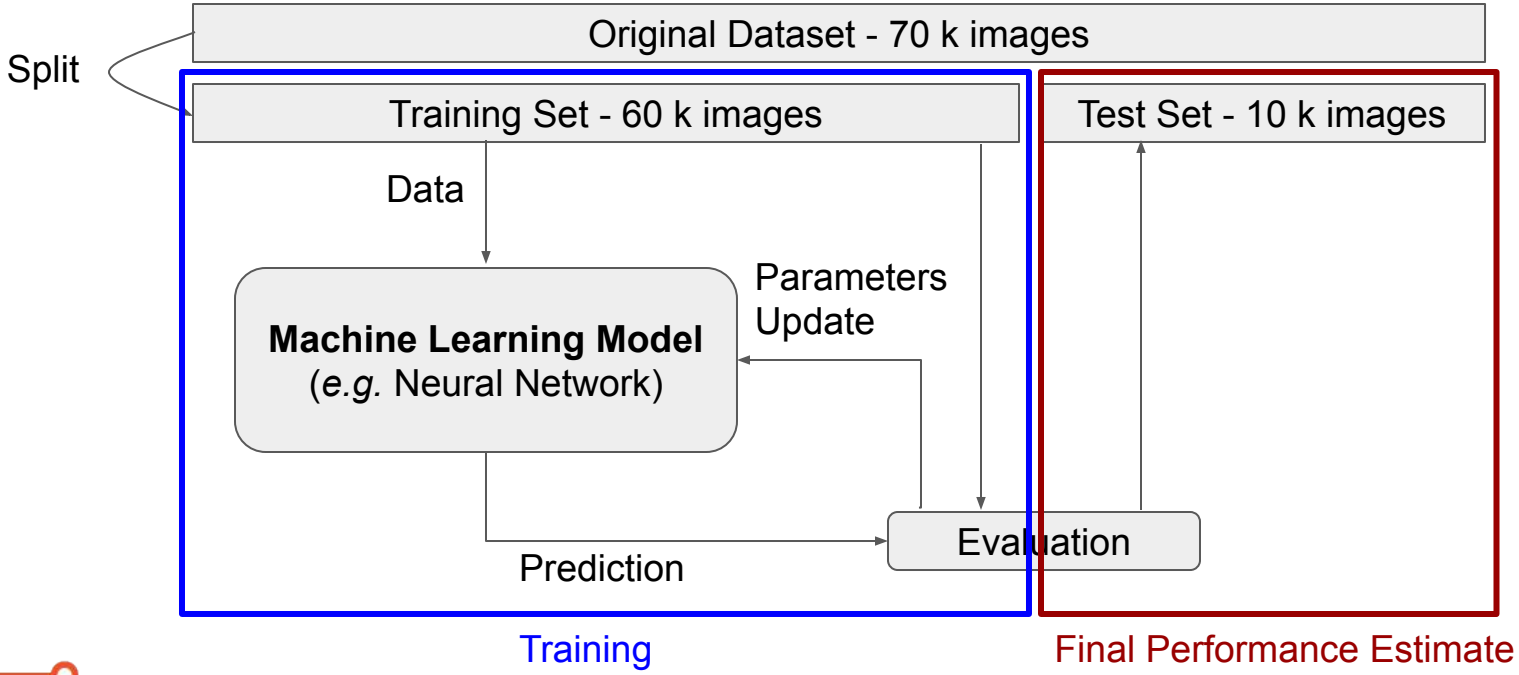


TRENT GLOBAL COLLEGE

zenika
<led by passion>

Shopee CODE LEAGUE 2020

Basic Model Training



TRENT GLOBAL COLLEGE

zenika
<led by passion>

Shopee CODE LEAGUE 2020

Libraries Importation

```
# TensorFlow and tf.keras
import tensorflow as tf
from tensorflow import keras

# Helper libraries
import numpy as np
import matplotlib.pyplot as plt

print(tf.__version__)
```

```
2.2.0
```

Keras: a high-level API to build and train models in TensorFlow

NumPy: Scientific Computing with Python

Pyplot: a plotting framework (e.g. visualize images, curves, histograms...)

zenika
<led by passion>

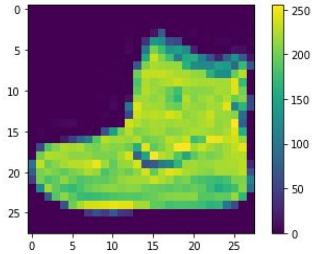
TRENT GLOBAL COLLEGE

Shopee CODE LEAGUE 2020

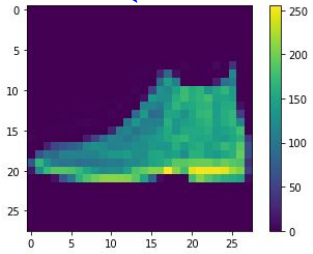
Dataset Importation

```
fashion_mnist = keras.datasets.fashion_mnist

(train_images, train_labels), (test_images, test_labels) = fashion_mnist.load_data()
```



09



09

Split

Original Dataset - 70 k images

Training Set - 60 k images

Test Set - 10 k images

TRENT GLOBAL COLLEGE

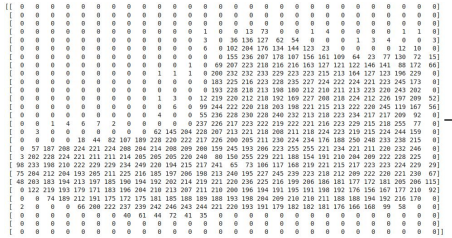
zenika
<led by passion>

Shopee CODE LEAGUE 2020

ML Model Construction

```
model = keras.Sequential([
    keras.layers.Flatten(input_shape=(28, 28)),
    keras.layers.Dense(128, activation='relu'),
    keras.layers.Dense(10)
])
```

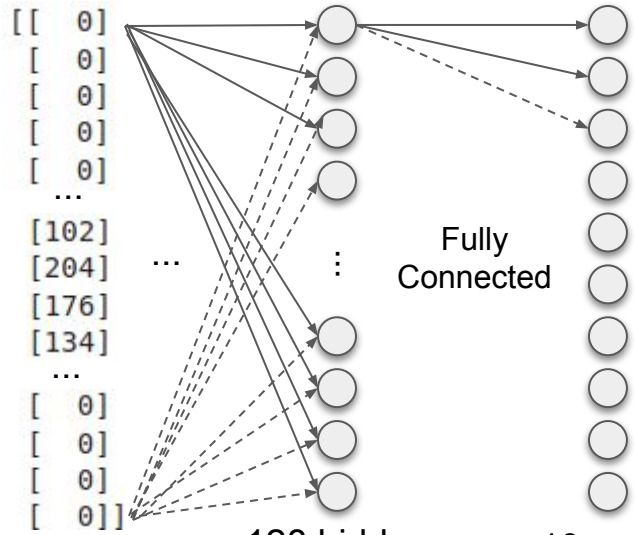
A Sequential model is appropriate for **a plain stack of layers** where each layer has **exactly one input tensor and one output tensor**.



(28, 28)

Flatten

(784, 1)



Fully Connected

128 hidden units

10 output units

Label	Class
0	T-shirt/top
1	Trouser
2	Pullover
3	Dress
4	Coat
5	Sandal
6	Shirt
7	Sneaker
8	Bag
9	Ankle boot

TRENT GLOBAL COLLEGE

zenika
<led by passion>

Shopee CODE LEAGUE 2020

Model Compilation

Optimization algorithm (update parameters according to the **loss**)

A **loss** function measures how well the output of a model for a given input matches the target output. The goal is to minimize this difference during training.

```
model.compile(optimizer='adam',
              loss=tf.keras.losses.SparseCategoricalCrossentropy(from_logits=True),
              metrics=['accuracy'])
```

Accuracy is the number of correctly predicted data points out of all the data points

TRENT GLOBAL COLLEGE

zenika
<led by passion>

Shopee CODE LEAGUE 2020

Model Training

An **epoch** is a complete learning pass through the entire dataset

```
model.fit(train_images, train_labels, epochs=10)
```

```
1875/1875 [==============================] - 3s 2ms/step - loss: 0.3138 - accuracy: 0.8854
Epoch 5/10
1875/1875 [==============================] - 3s 2ms/step - loss: 0.2948 - accuracy: 0.8914
Epoch 6/10
1875/1875 [==============================] - 3s 2ms/step - loss: 0.2803 - accuracy: 0.8958
Epoch 7/10
1875/1875 [==============================] - 3s 2ms/step - loss: 0.2678 - accuracy: 0.9006
Epoch 8/10
1875/1875 [==============================] - 3s 2ms/step - loss: 0.2590 - accuracy: 0.9043
Epoch 9/10
1875/1875 [==============================] - 3s 2ms/step - loss: 0.2464 - accuracy: 0.9078
Epoch 10/10
1875/1875 [==============================] - 3s 2ms/step - loss: 0.2402 - accuracy: 0.9099

<tensorflow.python.keras.callbacks.History at 0x7fe472eff4a8>
```

TRENT GLOBAL COLLEGE

zenika
<led by passion>

Shopee CODE LEAGUE 2020

Evaluate Accuracy

```
test_loss, test_acc = model.evaluate(test_images,  test_labels, verbose=2)

print('\nTest accuracy:', test_acc)
```

```
313/313 - 1s - loss: 0.3382 - accuracy: 0.8840

Test accuracy: 0.8840000033378601
```

88% of images are correctly classified

zenika
<led by passion>

TRENT
GLOBAL COLLEGE

Shopee
CODE
LEAGUE
2020

Make predictions

```
probability_model = tf.keras.Sequential([model,
                                         tf.keras.layers.Softmax()])
```

The **Softmax** layer normalizes the output into a probability distribution.

Model output **without** Softmax:

```
array([-11.4, -15.4, -15.1, -14.5, -18.8, -1.4, -13.2, 3.39, -16.1, 7.92], dtype=float32)
```

Model output **with** Softmax:

```
array([4.05e-09, 7.34e-11, 9.56e-11, 1.89e-10, 2.36e-12, 8.91e-05, 6.72e-10, 0.0107, 3.56e-11, 0.989], dtype=float32)
```

zenika
<led by passion>

TRENT GLOBAL COLLEGE

Shopee CODE LEAGUE 2020

Make predictions

```
probability_model = tf.keras.Sequential([model,
                                         tf.keras.layers.Softmax()])
```

```
predictions = probability_model.predict(test_images)
```

```
array([1.2275562e-07, 8.9534730e-08, 7.6224147e-09, 1.9726743e-08,
       6.1515345e-07, 4.6329503e-03, 4.1164008e-06, 1.5043605e-02,
       2.8817641e-07, 9.8031825e-01], dtype=float32)
```

The **Softmax** layer normalizes the output into a probability distribution.

zenika
<led by passion>

Make predictions

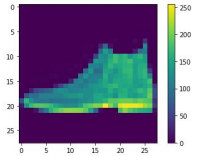
```
probability_model = tf.keras.Sequential([model,
                                         tf.keras.layers.Softmax()])
```

```
predictions = probability_model.predict(test_images)
```

```
array([1.2275562e-07, 8.9534730e-08, 7.6224147e-09, 1.9726743e-08,
       6.1515345e-07, 4.6329503e-03, 4.1164008e-06, 1.5043605e-02,
       2.8817641e-07, 9.8031825e-01], dtype=float32)
```

```
np.argmax(predictions[0])
```

```
9
```



Ankel boot !

The **Softmax** layer normalizes the output into a probability distribution.

TRENT GLOBAL COLLEGE

zenika
<led by passion>

Shopee CODE LEAGUE 2020

Make predictions

```
probability_model = tf.keras.Sequential([model,
                                         tf.keras.layers.Softmax()])
```

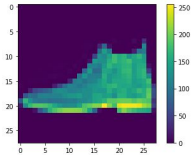
The **Softmax** layer normalizes the output into a probability distribution.

```
predictions = probability_model.predict(test_images)
```

```
array([1.2275562e-07, 8.9534730e-08, 7.6224147e-09, 1.9726743e-08,
       6.1515345e-07, 4.6329503e-03, 4.1164008e-06, 1.5043605e-02,
       2.8817641e-07, 9.8031825e-01], dtype=float32)
```

```
np.argmax(predictions[0])
```

```
9
```



Ankel boot !

Make prediction with
your own (28, 28) image:

probability_model.predict(**my_image**)

TRENT GLOBAL COLLEGE

zenika
<led by passion>

Shopee CODE LEAGUE 2020

Key Take Aways

- Tensors are multidimensional arrays, useful to represent data
- With Machine Learning, we generally try to **deduce rules from data and answers**
- Data (features) and answers (labels) constitute a **dataset**
- Model training: we split the dataset into a **training set** and a **testing set**
- Model performance during training is measured with the **loss**
- Model training is **iterative**

zenika
<led by passion>

TRENT GLOBAL COLLEGE

Shopee CODE LEAGUE 2020

Next ?

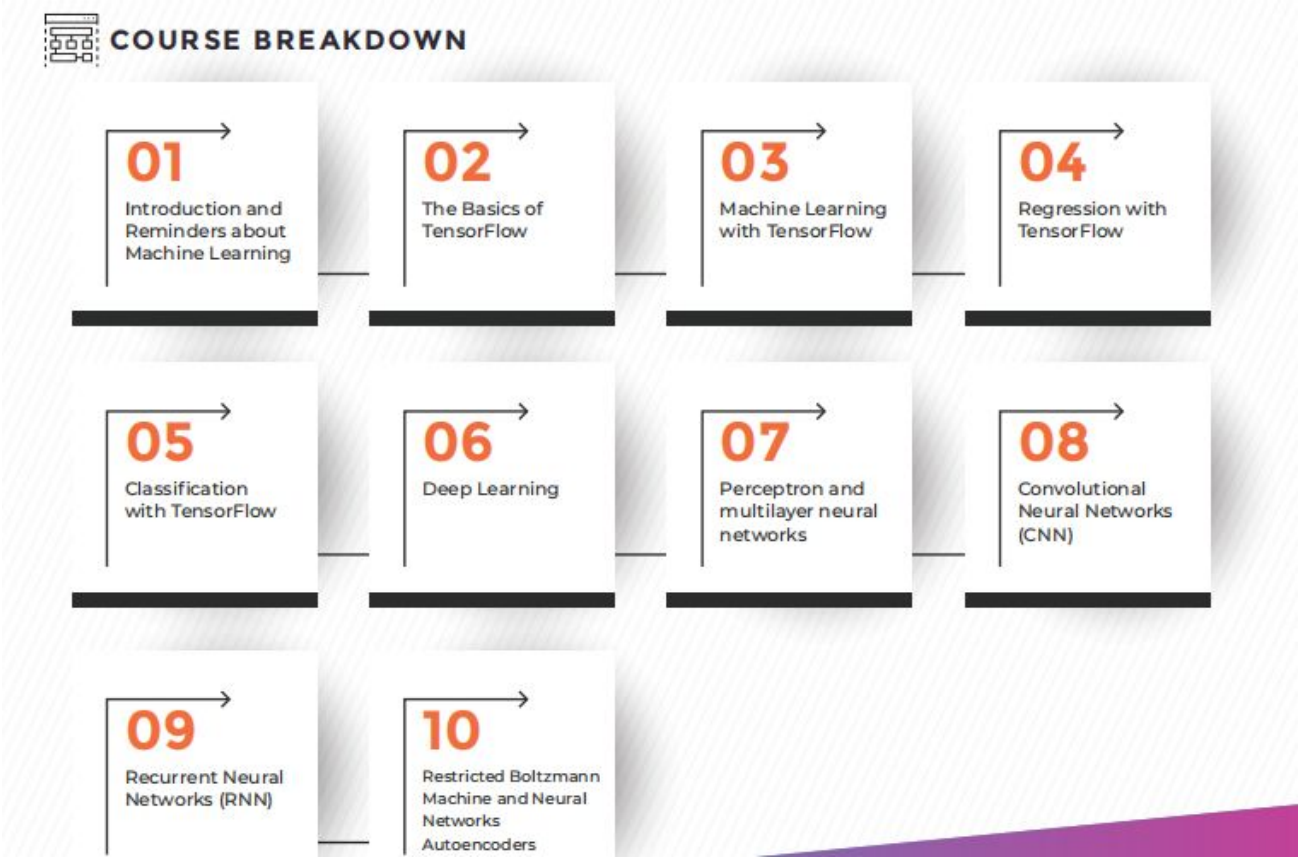
- TensorFlow Documentation, https://www.tensorflow.org/learn
- Courses/Training
- MOOC
- Books, https://www.deeplearningbook.org
- Blogs, https://towardsdatascience.com/
- Kaggle Competitions, https://www.kaggle.com
- Code your own ML project !

zenika
<led by passion>

TRENT
GLOBAL COLLEGE

Shopee
CODE
LEAGUE
2020

COURSE BREAKDOWN



https://bit.ly/CodeMasterClass

Thanks for attending !

harry.pommier@zenika.com

https://www.linkedin.com/in/harry-pommier-phd-a3911145

https://medium.com/@harry.pommier

Neural Style Transfer with TensorFlow, Harry POMMIER

zenika
<led by passion>

TRENT GLOBAL COLLEGE

Shopee CODE LEAGUE 2020

Your Feedback Matters!

https://techatshopee.formstack.com/forms/shopeecodeleague_workshopfeedbackform

Tensorflow Class

https://bit.ly/CodeMasterClass

zenika
<led by passion>

TRENT
GLOBAL COLLEGE

Shopee CODE LEAGUE 2020