

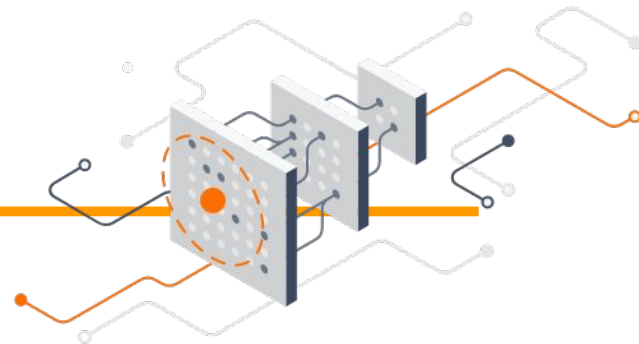


TensorFlow

Scripting no Processamento de Linguagem Natural

Mestrado Integrado em Engenharia Informática

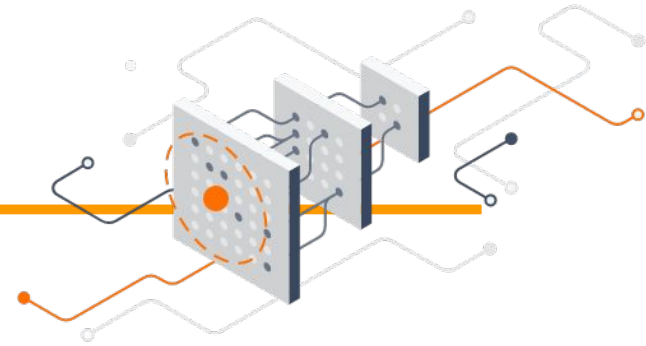
O que é?



- **Biblioteca open-source**
- **Utilizada para machine e deep learning**
- **Criada originalmente pela equipa Google Brains em 2015**
- **Sucessor de DistBelief**



O que significa?



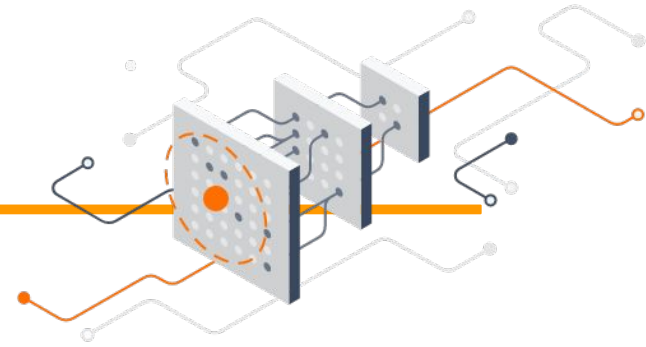
O nome **Tensorflow** vem do uso do componente principal, o tensor (uma matriz de n-dimensões)

Scalar Vector Matrix Tensor

1

$$\begin{bmatrix} 1 \\ 2 \end{bmatrix}$$
$$\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}$$
$$\begin{bmatrix} \begin{bmatrix} 1 & 2 \end{bmatrix} & \begin{bmatrix} 3 & 2 \end{bmatrix} \\ \begin{bmatrix} 1 & 7 \end{bmatrix} & \begin{bmatrix} 5 & 4 \end{bmatrix} \end{bmatrix}$$

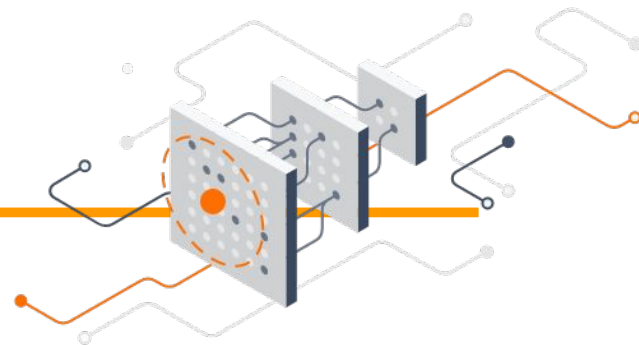
Features



- **Eficaz com arrays multidimensionais**
- **Suporte de redes neuronais complexas**
- **Escalabilidade de computação**
- **Capacidade de debugging e criação de modelos rápidos**



Keras

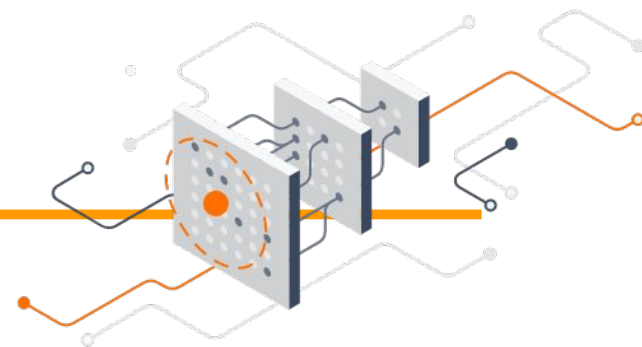
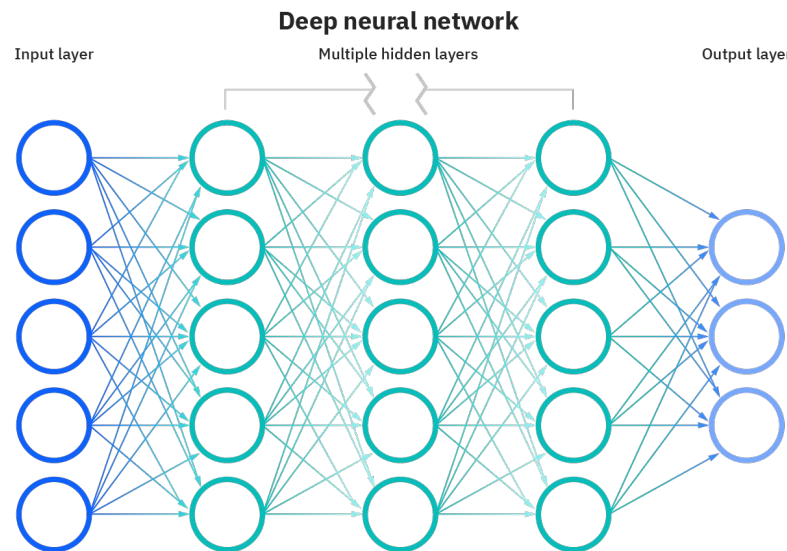
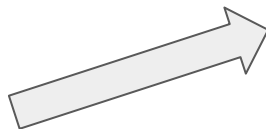
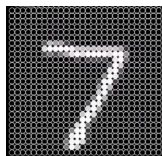


Dá uso da biblioteca Keras (`tf.keras`) para a criação de redes neuronais. Já esteve disponível para outras bibliotecas como Microsoft Cognitive Toolkit, mas agora é exclusivo ao Tensorflow.

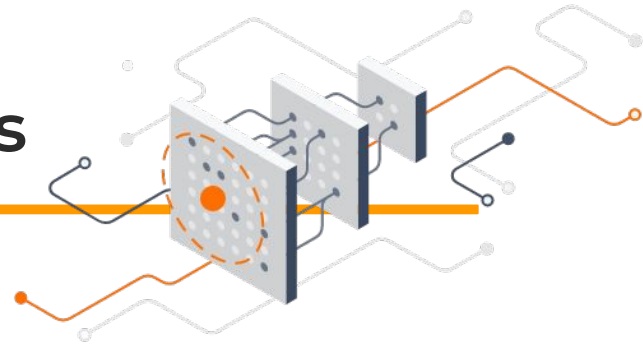
Como usar?

Construção e uso de um modelo:

1. Importação das dependências
2. Construção da rede neuronal
3. Compilação e treino
4. Avaliação

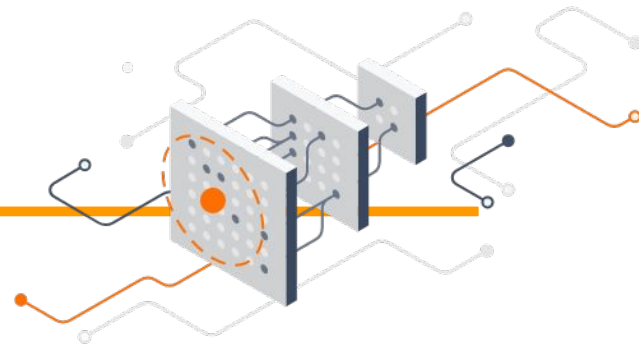


Importação das dependências



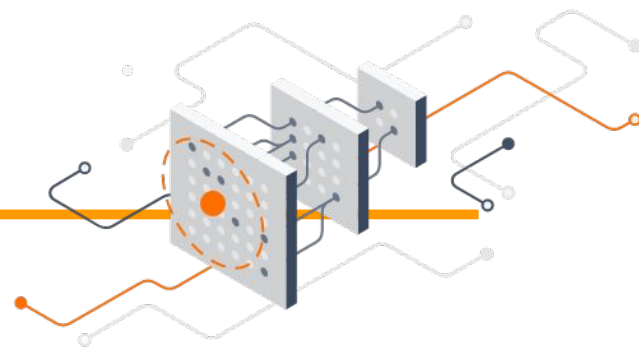
```
from tensorflow.keras.models import Sequential, load_model  
from tensorflow.keras.layers import Dense  
from sklearn.metrics import accuracy_score
```

Construção da rede neuronal



```
model = Sequential()  
model.add(Dense(units=32, activation='relu', input_dim=len(X_train.columns)))  
model.add(Dense(units=64, activation='relu'))  
model.add(Dense(units=1, activation='sigmoid'))
```


Compilação e treino



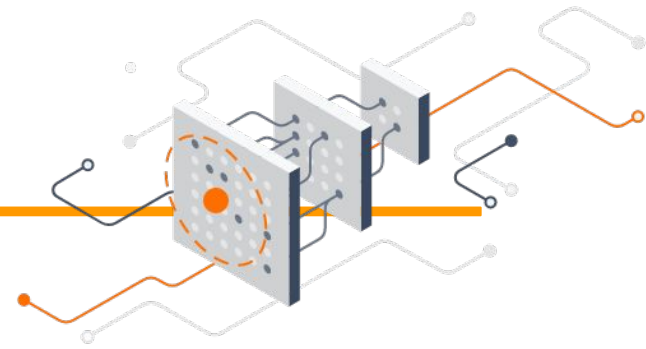
▶ ▶ M↓

```
model.compile(loss='binary_crossentropy', optimizer='sgd', metrics='accuracy')
```

▶ ▶ M↓

```
model.fit(X_train, y_train, epochs=200, batch_size=32)
```

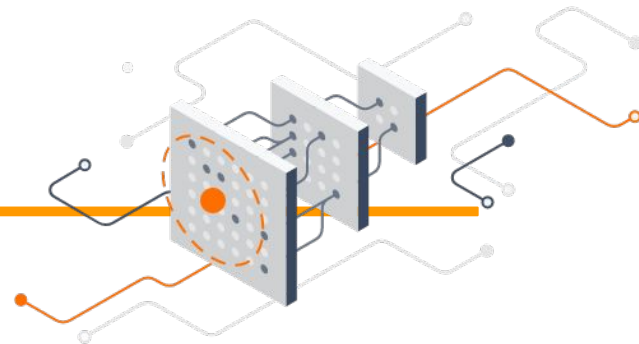
Avaliação



Após avaliar os resultados de teste, podemos fazer as mudanças necessárias ao modelo de modo a aumentar a precisão.

```
▶ ▶ ML  
y_pr = model.predict(X_test)  
y_pr = [0 if val < 0.5 else 1 for val in y_pr]  
  
▶ ▶ ML  
accuracy_score(y_test,y_pr)  
  
0.7998580553584103
```

Outras implementações



- `tf.graph_util`
- `tf.image`
- `tf.io`
- `tf.keras`
- `tf.linalg`
- `tf.lite`
- `tf.lookup`
- `tf.math`
- `tf.mixed_precision`
- `tf.mlir`
- `tf.nest`
- `tf.nn`
- `tf.profiler`
- `tf.quantization`
- `tf.queue`
- `tf.ragged`

...

Dispõe de vários métodos de encoding de vários tipos de ficheiros e datasets.

Por exemplo, **`tensorflow.audio`**, **`tensorflow.image`** e **`tensorflow.io`** são usados para dar encoding a vários tipos de ficheiros e até fazer operações sobre estes, como o aumento do contraste em imagens.

Outras implementações

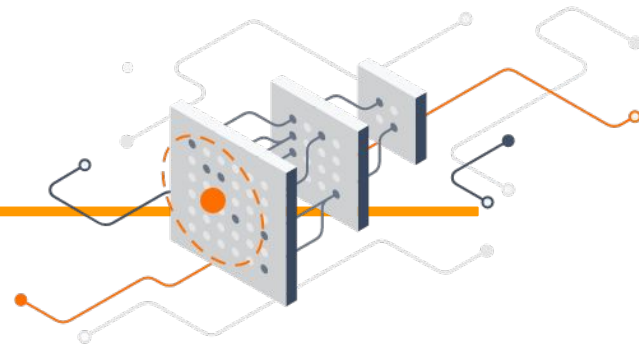


Image decoding and encoding

- `tf.io.decode_bmp`
- `tf.io.decode_gif`
- `tf.io.decode_image`
- `tf.io.decode_jpeg`
- `tf.io.decode_and_crop_jpeg`
- `tf.io.decode_png`
- `tf.io.encode_jpeg`
- `tf.io.encode_png`

Cropping

- `tf.image.central_crop`
- `tf.image.crop_and_resize`
- `tf.image.crop_to_bounding_box`
- `tf.io.decode_and_crop_jpeg`
- `tf.image.extract_glimpse`
- `tf.image.random_crop`
- `tf.image.resize_with_crop_or_pad`



TensorFlow

Scripting no Processamento de Linguagem Natural

Mestrado Integrado em Engenharia Informática