1. ⚙️ المقدمة

- التعلم العميق أصبح أساسًا لتطبيقات الذكاء الاصطناعي الحديثة.
- تنفيذ الشبكات العصبية كان معقدًا في البداية، مما أدى إلى ظهور **أطر العمل (Frameworks)** لتبسيطها مثل TensorFlow و Keras.

2. TensorFlow 🔬

التعريف

هو إطار عمل مفتوح المصدر من تطوير **Google Brain Team** سنة **2015**، مخصص للتعلم الآلي والعميق.
الاسم يعني "تدفّق المصفوفات متعددة الأبعاد (Tensors)."

التاريخ

- 2007–2010: مشروع داخلي باسم DistBelief.
- 2015: إطلاق TensorFlow 1.0.
- 2019: إطلاق TensorFlow 2.0 بواجهة أسهل ودمج Keras رسميًا.

المزايا

- يدعم CPU و GPU و TPU.
- قابل للنشر على الهاتف، الويب، والسحابة.
- مرن وقوي، ويملك مجتمع دعم ضخم.

العيوب

- معقّد في النسخ الأولى.
- صعب للمبتدئين.
- كثرة التحديثات تغيّر الصيغة أحيانًا.

3. Keras 🧩

التعريف

مكتبة عالية المستوى لبناء الشبكات العصبية بسهولة، طوّرها **François Chollet** سنة **2015**.

التطور

- بدأت تعمل فوق Theano و CNTK.
- أصبحت الأكثر استخدامًا في التعليم والبحث.
- سنة 2019 دُمجت رسميًا داخل TensorFlow (tf.keras).

المزايا

- سهلة وبسيطة في الاستخدام.
- مناسبة للتجارب التعليمية والبحثية.
- قابلة للتخصيص وتتكامل مع TensorFlow.

العيوب

- مرونة أقل في النماذج المعقدة.
- تعتمد على TensorFlow كخلفية تشغيلية.

4. ⚙️ العلاقة بينهما

- **TensorFlow** = المحرك الذي ينفّذ العمليات الحسابية.
- **Keras** = الواجهة التي تسهّل بناء النماذج.

📘 باختصار:

Keras تحدد "ماذا نريد أن نفعل"،
و TensorFlow ينفّذ "كيف يتم ذلك".

5. 🧠 المكونات الأساسية

1. الطبقات (Layers).
2. دالة الخسارة (Loss).
3. المحسّن (Optimizer).
4. المقاييس (Metrics).

5. التدريب والتقييم والتنبؤ (Fit – Evaluate – Predict).

6. 💻 مثال عملي بسيط

```
import tensorflow as tf
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense
import numpy as np

model = Sequential([
Dense(32, activation='relu', input_shape=(10,)),
Dense(16, activation='relu'),
Dense(1, activation='sigmoid')
])

model.compile(optimizer='adam', loss='binary_crossentropy',
metrics=['accuracy'])

X = np.random.random((100, 10))
y = np.random.randint(2, size=(100, 1))
model.fit(X, y, epochs=10, batch_size=8)
```

➡️ يوضح المثال سهولة بناء شبكة عصبية باستخدام Keras.

7. 📊 المقارنة العامة

الجانب	TensorFlow	Keras
مستوى التحكم	منخفض (تفصيلي)	عالٍ (مبسّط)
السهولة	صعب نسبيًا	سهل وواضح

الجانب	TensorFlow	Keras
الأداء	قوي في المشاريع الكبيرة	مناسب للتجريب
الاستخدام	إنتاج وبحوث متقدمة	تعليم وتجارب
المطور	Google Brain	François Chollet (Google)

8. 🚀 أهم الاستخدامات

- الرؤية الحاسوبية.(CNN)
- معالجة اللغة الطبيعية.(RNN, LSTM)
- النماذج المولدة.(GAN)
- النماذج العملاقة مثل GPT و.BERT

9. 🧭 الخلاصة

- **TensorFlow**هو الإطار القوي للحوسبة العميقة.
- **Keras**هو الواجهة السهلة والمناسبة للتعليم.
- دمجهما معًا (tf.keras) يجمع بين **القوة والبساطة**.
- للاستخدام الأكاديميKeras :، وللإنتاج الصناعي.TensorFlow :