



**Data Science
Academy**

www.datascienceacademy.com.br

Deep Learning Frameworks

Por Que Usaremos TensorFlow 2?



Listamos abaixo as principais razões pelas quais usaremos o TensorFlow 2.0 (embora em algumas operações o TensorFlow 1.x ainda seja necessário):

- Os dados de treinamento são lidos usando pipelines de entrada que são criados usando `tf.data`, o que facilita muito o trabalho.
- Criamos, treinamos e validamos o modelo com Keras ou API Estimators. Não precisamos mais programar direto com código TensorFlow e isso reduz o tempo total de desenvolvimento.
- O TensorFlow Hub no ecossistema do TensorFlow contém um grande número de modelos pré-treinados. Usando a interface padrão, você pode importar qualquer um dos modelos do TensorFlow Hub e treiná-lo do zero ou ajustá-lo para seus dados usando a técnica de Transfer Learning (que estudaremos na Formação IA).
- Executamos e depuramos com Eager Execution e usamos `function` para obter os benefícios dos grafos (que ainda estão disponíveis na versão TF 1).
- Usamos estratégias de distribuição para treinamento distribuído. Para grandes tarefas de treinamento de Machine Learning, a API facilita a distribuição e o treinamento de modelos em diferentes configurações de hardware sem alterar a definição do modelo. Você pode distribuir sua carga de treinamento para vários aceleradores de hardware, como CPUs, GPUs e TPUs.
- Embora essa API suporte uma variedade de configurações de cluster, são fornecidos modelos para implantar o treinamento em clusters Kubernetes em ambientes locais ou na nuvem.
- O TensorFlow padronizará o `SavedModel` como um formato de intercâmbio para o TensorFlow Serving, o TensorFlow Lite, o TensorFlow.js, o TensorFlow Hub e muito mais. Crie o modelo usando uma API e use-o em todas as outras.



- Depois de treinar e salvar seu modelo, você pode executá-lo diretamente em seu aplicativo ou servi-lo usando uma das bibliotecas de implantação:
 - TensorFlow Serving: Uma biblioteca TensorFlow que permite que os modelos sejam veiculados em HTTP / REST ou em buffers de protocolo gRPC.
 - TensorFlow Lite: Uma solução leve do TensorFlow para dispositivos móveis e embarcados que oferece a capacidade de implantar modelos no Android, iOS e sistemas embarcados, como TPU Raspberry Pi e Edge.
 - TensorFlow.js: Permite a implementação de modelos em ambientes JavaScript, como em um navegador web ou no lado do servidor, através do Node.js. O TensorFlow.js também suporta a definição de modelos em JavaScript e o treinamento diretamente no navegador web, usando uma API do tipo Keras.