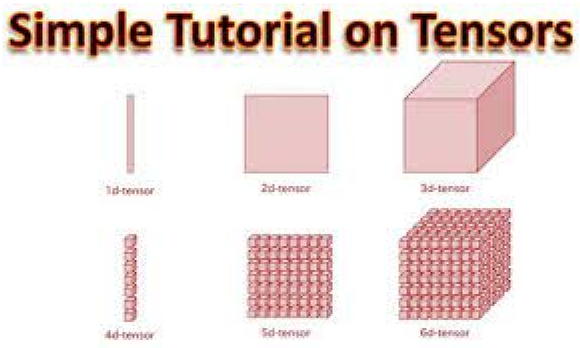
A Dive into TensorFlow: Tensors, Execution Modes, Gradient Descent, and Neural Networks

Introduction

Discover the core concepts of TensorFlow, the go-to machine learning library. This concise guide unravels tensors, execution modes (eager and lazy), the optimization powerhouse – gradient descent, and the essentials of neural networks.

Tensors and TensorFlow

TensorFlow revolves around tensors – versatile mathematical entities representing data. From scalars to matrices, tensors are the fundamental building blocks for expressing and manipulating data in this open-source machine learning library.



```
import tensorflow as tf
```

```
# Create tensors in TensorFlow
```

```
a = tf.constant(5)
b = tf.Variable(10)
c = a + b
```

Eager and Lazy Execution

- Eager Execution: TensorFlow's default mode, executing operations immediately as called. Ideal for debugging and experimentation.
- Lazy Execution: Implemented using `@tf.function`. Involves tracing and compiling operations into a graph for optimization, often enhancing performance.

```
# Eager Execution
def eager_fn(a, b):
    c = a * b + 1
    d = a * b * 3
    print(c)
    print(d)


# Lazy Execution
@tf.function
def lazy_fn(a, b):
    c = a * b + 1
    d = a * b * 3
    print(c)
    print(d)


# Eager Execution
def eager_fn(a, b):
    c = a * b + 1
    d = a * b * 3
    print(c)
    print(d)


# Lazy Execution
@tf.function
def lazy_fn(a, b):
    c = a * b + 1
    d = a * b * 3
    print(c)
    print(d)
```

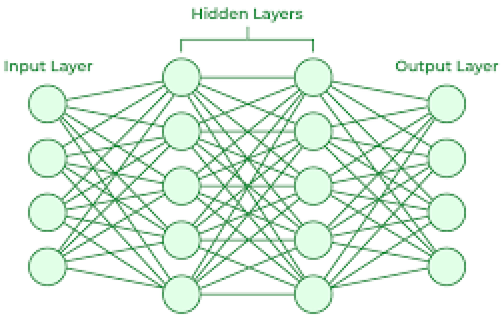
Gradient Descent

A pivotal optimization algorithm, gradient descent minimizes the loss function by iteratively updating model parameters in the opposite direction of the gradient.

```
x = tf.constant(3.0)
```

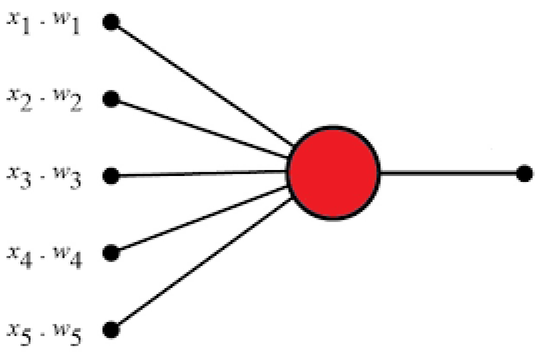
```
y = tf.constant(6.0)

w = tf.Variable(20.0)



def loss_compute():

    with tf.GradientTape() as tape:

        loss = tf.math.abs(w * x - y)

    dx = tape.gradient(loss, w)

    w.assign(w - dx)
```

Neural Networks



Explore the fundamentals:

- Neural Networks (NN): Algorithms designed for pattern recognition, inspired by human brain functioning.
- Artificial Neural Network (ANN): Comprising interconnected nodes in layers, adjusting weights during training to optimize predictions.
- Perceptron: The simplest neural network unit, processing inputs and producing an output.



Concluding this succinct journey, armed with the basics, delve deeper into the fascinating world of TensorFlow, machine learning, and the potential to address real-world challenges.

Check_Out_Detailed_Blog:-https://medium.com/@srivastavayushmaan1347/demystifying-tensorflow-a-comprehensive-guide-to-tensors-lazy-and-eager-execution-gradient-e30798062be5