

# Deep Learning exp-1

**1) Tensorflow:-**Tensorflow is a library in python that allows you to create dataflow graphs that describe how data moves through a graph.

## Tensorflow methods:-

### Creation:

tf.constant: Creates a constant tensor with the provided values.

tf.Variable: Creates a mutable tensor variable that can be updated during training.

tf.zeros: Creates a tensor filled with zeros.

tf.ones: Creates a tensor filled with ones.

tf.random.normal: Creates a tensor with random values from a normal distribution.

tf.random.uniform: Creates a tensor with random values from a uniform distribution.

### Operations:

tf.add: Element-wise addition of tensors.

tf.subtract: Element-wise subtraction of tensors.

tf.multiply: Element-wise multiplication of tensors.

tf.divide: Element-wise division of tensors.

tf.matmul: Matrix multiplication of tensors.

### Activation Functions:

tf.nn.relu: Rectified Linear Unit (ReLU) activation function.

tf.nn.sigmoid: Sigmoid activation function.

tf.nn.tanh: Hyperbolic tangent activation function.

tf.nn.softmax: Softmax activation function used for multi-class classification.

### Loss Functions:

tf.keras.losses: Module containing various loss functions (e.g., mean squared error, categorical cross-entropy).

### Optimizers:

tf.keras.optimizers: Module containing various optimization algorithms (e.g., Adam, SGD).

Model Building:

tf.keras.Sequential: A linear stack of layers for building neural network models.

tf.keras.Model: A customizable class for building complex models.

### **Model Training:**

tf.keras.Model.compile: Configures the model for training, specifying the optimizer, loss function, and metrics.

tf.keras.Model.fit: Trains the model on the given data for a specified number of epochs.

tf.keras.Model.evaluate: Evaluates the model's performance on a validation or test dataset.

tf.keras.Model.predict: Generates predictions for new data using the trained model.

### **Saving and Loading Models:**

tf.keras.models.save\_model: Saves the entire model to a file in the HDF5 format.

tf.keras.models.load\_model: Loads a saved model from a file.

### **Eager Execution:**

tf.config.run\_functions\_eagerly: A method to enable or disable eager execution.

**2) Keras**:-Keras is a powerful and easy-to-use free open source Python library for developing and evaluating deep learning models.

### **Keras methods:-**

#### **Model Building:**

Sequential: A linear stack of layers for building feedforward neural network models.

Model: A customizable class for building complex models with shared layers or multiple inputs/outputs.

#### **Layers:**

Dense: Fully connected layer.

Conv2D: 2D convolutional layer for image processing.

LSTM: Long Short-Term Memory layer for sequence data.

Embedding: Word embedding layer for natural language processing.

BatchNormalization: Layer that normalizes activations to stabilize training.

#### **Activation Functions:**

relu: Rectified Linear Unit (ReLU) activation function.

sigmoid: Sigmoid activation function.

softmax: Softmax activation function for multi-class classification.

### **Loss Functions:**

mean\_squared\_error: Mean Squared Error loss function.

categorical\_crossentropy: Categorical Cross-Entropy loss function for multi-class classification.

binary\_crossentropy: Binary Cross-Entropy loss function for binary classification.

### **Optimizers:**

SGD: Stochastic Gradient Descent optimizer.

Adam: Adaptive Moment Estimation optimizer.

RMSprop: Root Mean Square Propagation optimizer.

### **Model Training:**

compile: Configures the model for training, specifying the optimizer, loss function, and metrics.

fit: Trains the model on the given data for a specified number of epochs.

evaluate: Evaluates the model's performance on a validation or test dataset.

predict: Generates predictions for new data using the trained model.

### **Callbacks:**

EarlyStopping: Stops training early if a monitored metric has stopped improving.

ModelCheckpoint: Saves the model at certain intervals during training.

ReduceLROnPlateau: Reduces the learning rate when a metric has stopped improving.

### **Preprocessing:**

ImageDataGenerator: Generates augmented images for image data during training.

Tokenizer: Tokenizes text data and converts it into sequences for natural language processing.