# Functional API for Deep Learning



```
# defining the architecture of the model
model = Sequential()
model.add(InputLayer(input_shape=(input_neurons,)))
model.add(Dense(units=neuron_hidden_layer_1, activation='relu'))
model.add(Dense(units=neuron_hidden_layer_2, activation='relu'))
model.add(Dense(units=output_neurons, activation='sigmoid'))
```

```
# summary of the model
model.summary()
```

Layer (type)	Output Shape	Param #
dense_1 (Dense)	(None, 10)	120
dense_2 (Dense)	(None, 5)	55
dense_3 (Dense)	(None, 1)	6

Total params: 181 Trainable params: 181 Non-trainable params: 0 nalytics idhya



```
# defining the architecture of the model
model = Sequential()
model.add(InputLayer(input shape=(input neurons,)))
model.add(Dense(units=neuron hidden layer 1, activation='relu'))
model.add(Dense(units=neuron hidden layer 2, activation='relu'))
model.add(Dense(units=output neurons, activation='sigmoid'))
                                                                 nalytics
idhya
# summary of the model
model.summary()
                                                      Param #
                             Output Shape
Layer (type)
dense 1 (Dense)
                             (None, 10)
                                                      120
dense 2 (Dense)
                             (None, 5)
                                                      55
dense 3 (Dense)
                             (None, 1)
                                                       6
Total params: 181
Trainable params: 181
Non-trainable params: 0
```



```
# defining the architecture of the model
model = Sequential()
model.add(InputLayer(input shape=(input neurons,)))
model.add(Dense(units=neuron hidden layer 1, activation='relu'))
model.add(Dense(units=neuron hidden layer 2, activation='relu'))
model.add(Dense(units=output neurons, activation='sigmoid'))
                                                                 nalytics
idhya
# summary of the model
model.summary()
                                                      Param #
                            Output Shape
Layer (type)
dense 1 (Dense)
                                                      120
                             (None, 10)
dense 2 (Dense)
                             (None, 5)
                                                      55
dense 3 (Dense)
                             (None, 1)
                                                      6
Total params: 181
Trainable params: 181
Non-trainable params: 0
```

Input Layer



```
# defining the architecture of the model
model = Sequential()
                                                                                               Input Layer
model.add(InputLayer(input shape=(input neurons,)))
model.add(Dense(units=neuron hidden layer 1, activation='relu'))
model.add(Dense(units=neuron hidden layer 2, activation='relu'))
model.add(Dense(units=output neurons, activation='sigmoid'))
                                                               nalytics
                                                                                             HiddenLayer1
# summary of the model
model.summary()
                                                     Param #
                            Output Shape
Layer (type)
                                                                                             HiddenLayer2
dense 1 (Dense)
                                                     120
                            (None, 10)
dense 2 (Dense)
                            (None, 5)
                                                     55
dense 3 (Dense)
                            (None, 1)
                                                     6
Total params: 181
Trainable params: 181
Non-trainable params: 0
```



```
# defining the architecture of the model
model = Sequential()
                                                                                              Input Layer
model.add(InputLayer(input shape=(input neurons,)))
model.add(Dense(units=neuron hidden layer 1, activation='relu'))
model.add(Dense(units=neuron hidden layer 2, activation='relu'))
model.add(Dense(units=output neurons, activation='sigmoid'))
                                                               nalytics
                                                                                            HiddenLayer1
# summary of the model
model.summary()
                                                     Param #
                            Output Shape
Layer (type)
                                                                                            HiddenLayer2
dense 1 (Dense)
                                                     120
                            (None, 10)
dense 2 (Dense)
                            (None, 5)
                                                     55
dense 3 (Dense)
                            (None, 1)
                                                     6
                                                                                             OutputLayer
Total params: 181
Trainable params: 181
Non-trainable params: 0
```

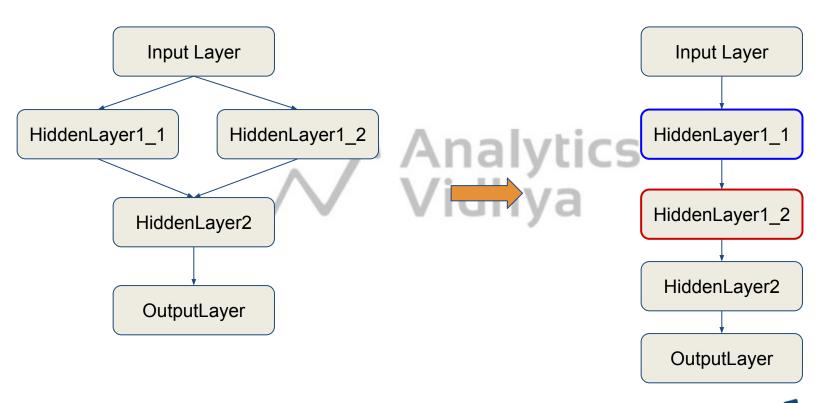


```
# defining the architecture of the model
model = Sequential()
                                                                                              Input Layer
model.add(InputLayer(input shape=(input neurons,)))
model.add(Dense(units=neuron hidden layer 1, activation='relu'))
model.add(Dense(units=neuron hidden layer 2, activation='relu'))
model.add(Dense(units=output neurons, activation='sigmoid'))
                                                               nalytic
                                                                                            HiddenLayer1
# summary of the model
model.summary()
                                                     Param #
                            Output Shape
Layer (type)
                                                                                            HiddenLayer2
dense 1 (Dense)
                                                     120
                            (None, 10)
dense 2 (Dense)
                            (None, 5)
                                                     55
dense 3 (Dense)
                            (None, 1)
                                                     6
                                                                                             OutputLayer
Total params: 181
Trainable params: 181
Non-trainable params: 0
```

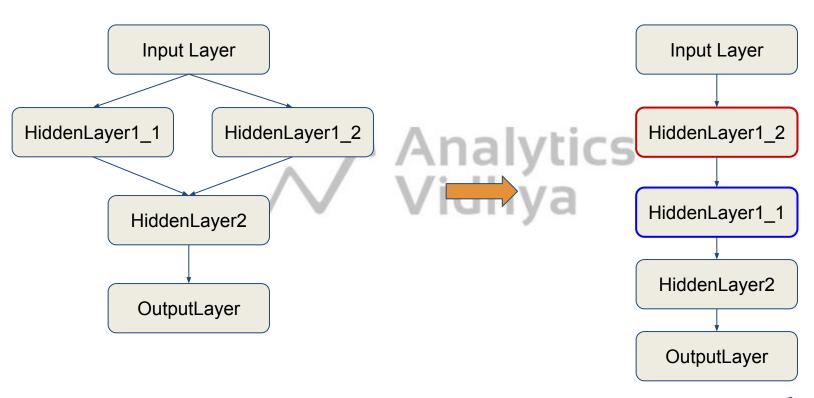


```
# defining the architecture of the model
model = Sequential()
                                                                                                Input Layer
model.add(InputLayer(input shape=(input neurons,)))
model.add(Dense(units=neuron hidden layer 1, activation='relu'))
model.add(Dense(units=neuron hidden layer 2, activation='relu'))
model.add(Dense(units=output neurons, activation='sigmoid'))
                                                                              HiddenLayer1 1
                                                                                                            HiddenLayer1 2
# summary of the model
model.summary()
Layer (type)
                            Output Shape
                                                      Param #
                                                                                              HiddenLayer2
dense 1 (Dense)
                            (None, 10)
                                                      120
dense 2 (Dense)
                            (None, 5)
                                                      55
dense 3 (Dense)
                            (None, 1)
                                                                                               OutputLayer
Total params: 181
Trainable params: 181
Non-trainable params: 0
```











#### Functional API

- 1. Provides more flexibility to define models
- 2. Can define multiple inputs, outputs models
- 3. Split and share the intermediate layers
- 4. Build State-of-the-Art model architectures or Custom architecture

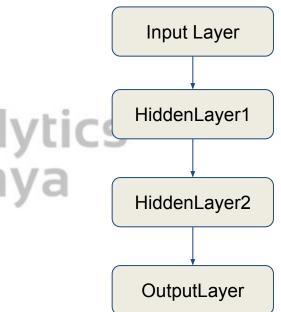


Input Layer Create Layers and Connect Analytics Vidhya HiddenLayer1 HiddenLayer2 OutputLayer



```
# defining the architecture of the model using Functional API
x = Input(shape = (input neurons,))
hidden1 = Dense(units=neuron hidden layer 1, activation='relu')(x)
hidden2 = Dense(units=neuron hidden layer 2, activation='relu')(hidden1)
output = Dense(units=output neurons, activation='sigmoid')(hidden2)
model functional = Model(x, output)
# summary of the model
model functional.summary()
                             Output Shape
                                                        Param #
input 2 (InputLayer)
                              (None, 11)
dense 4 (Dense)
                              (None, 10)
                                                        120
                                                        55
dense 5 (Dense)
                              (None, 5)
dense 6 (Dense)
                              (None, 1)
Total params: 181
Trainable params: 181
```

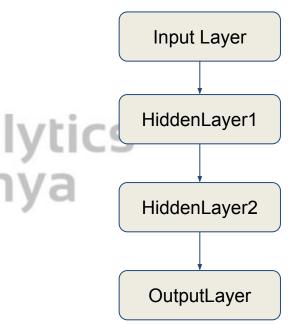
Non-trainable params: 0





Layer (type)	Output Shape	Param #
input_2 (InputLayer)	(None, 11)	0
dense_4 (Dense)	(None, 10)	120
dense_5 (Dense)	(None, 5)	55
dense_6 (Dense)	(None, 1)	6
Total params: 181		

Total params: 181
Trainable params: 181
Non-trainable params: 0





```
[ ] # defining the architecture of the model using Functional API
    x = Input(shape = (input_neurons,))
    hidden1 = Dense(units=neuron_hidden_layer_1, activation='relu')(x)

hidden2 = Dense(units=neuron_hidden_layer_2, activation='relu')(hidden1)

output = Dense(units=output_neurons, activation='sigmoid')(hidden2)

model_functional = Model(x, output)

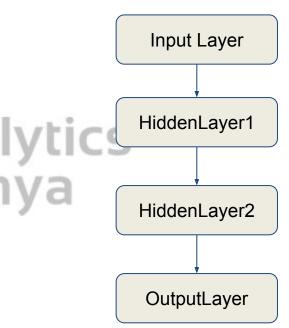
[ ] # summary of the model
    model_functional.summary()
Layer (type)

Output Shape

Param #
```

Layer (type)	Output Shape	Param #
input_2 (InputLayer)	(None, 11)	0
dense_4 (Dense)	(None, 10)	120
dense_5 (Dense)	(None, 5)	55
dense_6 (Dense)	(None, 1)	6
motal narama, 101		

Total params: 181
Trainable params: 181
Non-trainable params: 0





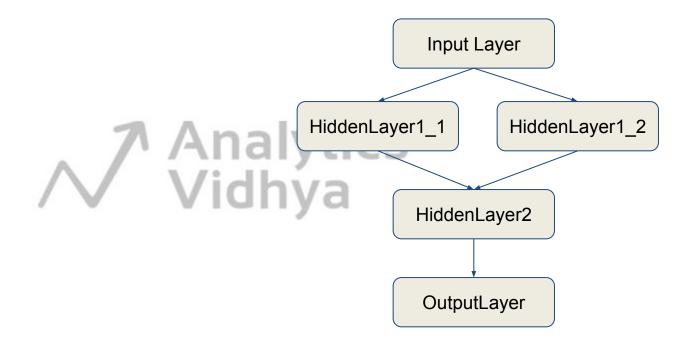
```
# defining the architecture of the model using Functional API
x = Input(shape = (input neurons,))
                                                                                                 Input Layer
hidden1 = Dense(units=neuron hidden layer 1, activation='relu')(x)
hidden2 = Dense(units=neuron hidden layer 2, activation='relu')(hidden1)
output = Dense(units=output neurons, activation='sigmoid')(hidden2)
model functional = Model(x, output)
                                                                                               HiddenLayer1
# summary of the model
model functional.summary()
                            Output Shape
                                                      Param #
                                                                                               HiddenLayer2
input 2 (InputLayer)
                            (None, 11)
dense 4 (Dense)
                            (None, 10)
                                                      120
                                                      55
dense 5 (Dense)
                            (None, 5)
                                                                                                OutputLayer
dense 6 (Dense)
                             (None, 1)
Total params: 181
Trainable params: 181
```

Non-trainable params: 0

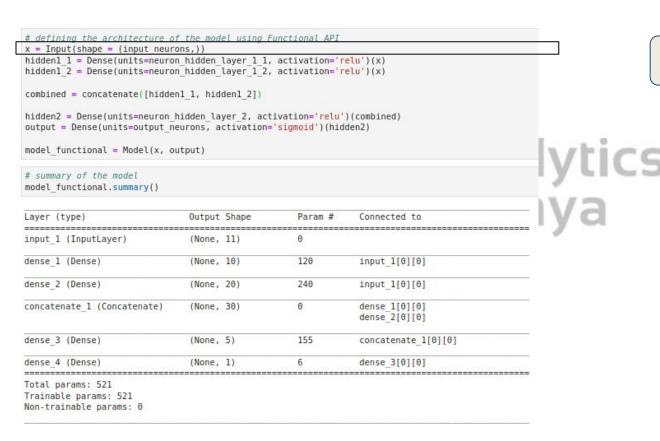


```
# defining the architecture of the model using Functional API
x = Input(shape = (input neurons,))
                                                                                                 Input Layer
hidden1 = Dense(units=neuron hidden layer 1, activation='relu')(x)
hidden2 = Dense(units=neuron hidden layer 2, activation='relu')(hidden1)
output = Dense(units=output neurons, activation='sigmoid')(hidden2)
model functional = Model(x, output)
                                                                                               HiddenLayer1
# summary of the model
model functional.summary()
                            Output Shape
                                                      Param #
                                                                                               HiddenLayer2
input 2 (InputLayer)
                             (None, 11)
dense 4 (Dense)
                             (None, 10)
                                                      120
                                                      55
dense 5 (Dense)
                             (None, 5)
                                                                                                OutputLayer
dense 6 (Dense)
                             (None, 1)
Total params: 181
Trainable params: 181
Non-trainable params: 0
```



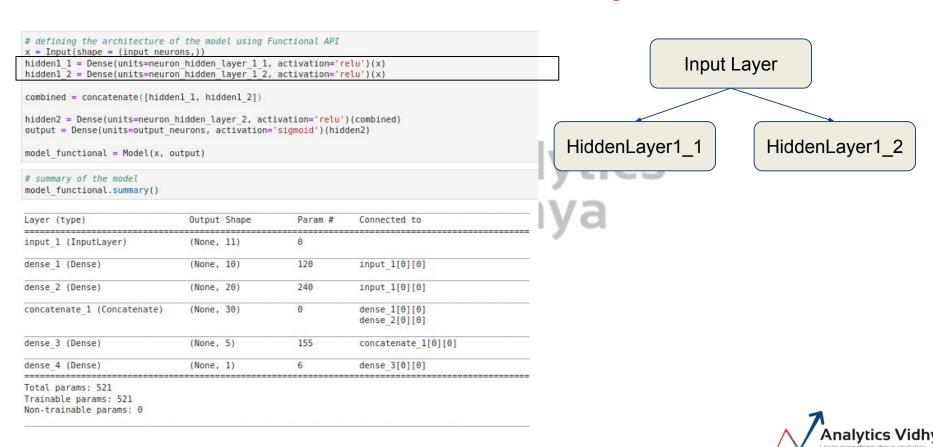




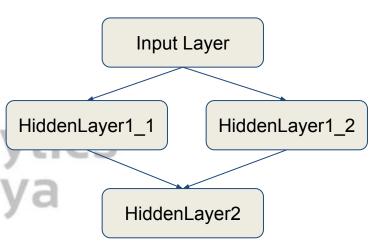


Input Layer





```
# defining the architecture of the model using Functional API
x = Input(shape = (input neurons,))
hidden1 1 = Dense(units=neuron hidden layer 1 1, activation='relu')(x)
hidden1 2 = Dense(units=neuron hidden layer 1 2, activation='relu')(x)
combined = concatenate([hidden1 1, hidden1 2])
hidden2 = Dense(units=neuron hidden layer 2, activation='relu')(combined)
output = Dense(units=output neurons, activation='sigmoid')(hidden2)
model functional = Model(x, output)
# summary of the model
model functional.summary()
Layer (type)
                                 Output Shape
                                                      Param #
                                                                  Connected to
input 1 (InputLaver)
                                 (None, 11)
dense 1 (Dense)
                                                                  input 1[0][0]
                                 (None, 10)
                                                      120
dense 2 (Dense)
                                 (None, 20)
                                                      240
                                                                  input 1[0][0]
concatenate 1 (Concatenate)
                                                      0
                                                                  dense 1[0][0]
                                 (None, 30)
                                                                  dense 2[0][0]
dense 3 (Dense)
                                 (None, 5)
                                                      155
                                                                  concatenate 1[0][0]
dense 4 (Dense)
                                 (None, 1)
                                                                  dense 3[0][0]
Total params: 521
Trainable params: 521
Non-trainable params: 0
```





```
# defining the architecture of the model using Functional API
x = Input(shape = (input neurons,))
                                                                                                                            Input Layer
hidden1 1 = Dense(units=neuron hidden layer 1 1, activation='relu')(x)
hidden1 2 = Dense(units=neuron hidden layer 1 2, activation='relu')(x)
combined = concatenate([hidden1 1, hidden1 2])
hidden2 = Dense(units=neuron hidden layer 2, activation='relu')(combined)
output = Dense(units=output neurons, activation='sigmoid')(hidden2)
                                                                                                     HiddenLayer1 1
                                                                                                                                           HiddenLayer1 2
model functional = Model(x, output)
# summary of the model
model functional.summary()
Layer (type)
                               Output Shape
                                                   Param #
                                                               Connected to
                                                                                                                          HiddenLayer2
input 1 (InputLaver)
                               (None, 11)
dense 1 (Dense)
                                                              input 1[0][0]
                               (None, 10)
                                                   120
dense 2 (Dense)
                               (None, 20)
                                                   240
                                                              input 1[0][0]
concatenate 1 (Concatenate)
                                                   0
                                                               dense 1[0][0]
                               (None, 30)
                                                                                                                          OutputLayer
                                                              dense 2[0][0]
dense 3 (Dense)
                               (None, 5)
                                                   155
                                                              concatenate 1[0][0]
dense 4 (Dense)
                               (None, 1)
                                                               dense 3[0][0]
Total params: 521
Trainable params: 521
Non-trainable params: 0
```



```
# defining the architecture of the model using Functional API
x = Input(shape = (input neurons,))
                                                                                                                           Input Layer
hidden1 1 = Dense(units=neuron hidden layer 1 1, activation='relu')(x)
hidden1 2 = Dense(units=neuron hidden layer 1 2, activation='relu')(x)
combined = concatenate([hidden1 1, hidden1 2])
hidden2 = Dense(units=neuron hidden layer 2, activation='relu')(combined)
output = Dense(units=output neurons, activation='sigmoid')(hidden2)
                                                                                                     HiddenLayer1 1
                                                                                                                                           HiddenLayer1 2
model functional = Model(x, output)
# summary of the model
model functional.summary()
Layer (type)
                               Output Shape
                                                   Param #
                                                               Connected to
                                                                                                                         HiddenLayer2
input 1 (InputLayer)
                               (None, 11)
dense 1 (Dense)
                                                              input 1[0][0]
                               (None, 10)
                                                   120
dense 2 (Dense)
                               (None, 20)
                                                   240
                                                              input 1[0][0]
concatenate 1 (Concatenate)
                                                   0
                                                              dense 1[0][0]
                               (None, 30)
                                                                                                                          OutputLayer
                                                              dense 2[0][0]
dense 3 (Dense)
                               (None, 5)
                                                   155
                                                              concatenate 1[0][0]
dense 4 (Dense)
                               (None, 1)
                                                               dense 3[0][0]
Total params: 521
Trainable params: 521
Non-trainable params: 0
```





