In [19]:
```
import tensorflow as tf
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import GRU, Dense

# Enable eager execution
tf.config.run_functions_eagerly(True)

# Define the model
model = Sequential([
    GRU(32, input_shape=(10, 1)),
    Dense(1, activation='sigmoid')
])

# Compile the model with the correct metric name
model.compile(optimizer='adam', loss='binary_crossentropy', metrics=['accuracy'])

# Generate some random data for training
import numpy as np
X = np.random.rand(100, 10, 1)
y = np.random.randint(0, 2, (100, 1))

# Train the model
model.fit(X, y, epochs=10, batch_size=32)
```

```
Epoch 1/10
4/4 [==============================] - 1s 173ms/step - loss: 0.6923 - accuracy: 0.5300
Epoch 2/10
4/4 [==============================] - 1s 172ms/step - loss: 0.6917 - accuracy: 0.5300
Epoch 3/10
4/4 [==============================] - 1s 176ms/step - loss: 0.6916 - accuracy: 0.5300
Epoch 4/10
4/4 [==============================] - 1s 183ms/step - loss: 0.6916 - accuracy: 0.5300
Epoch 5/10
4/4 [==============================] - 1s 197ms/step - loss: 0.6919 - accuracy: 0.5300
Epoch 6/10
4/4 [==============================] - 1s 191ms/step - loss: 0.6920 - accuracy: 0.5300
Epoch 7/10
4/4 [==============================] - 1s 202ms/step - loss: 0.6912 - accuracy: 0.5300
Epoch 8/10
4/4 [==============================] - 1s 225ms/step - loss: 0.6906 - accuracy: 0.5300
Epoch 9/10
4/4 [==============================] - 1s 200ms/step - loss: 0.6906 - accuracy: 0.5300
Epoch 10/10
4/4 [==============================] - 1s 183ms/step - loss: 0.6905 - accuracy: 0.5300
```

Out[19]:
```
<keras.src.callbacks.History at 0x20c8d9d2c50>
```

In [21]:
```
loss, accuracy = model.evaluate(X, y)
print(f'Test loss: {loss}, Test accuracy: {accuracy}')
```

```
4/4 [==============================] - 0s 85ms/step - loss: 0.6903 - accuracy: 0.5300
Test loss: 0.6903257966041565, Test accuracy: 0.5299999713897705
```