In [ ]:
```
import yfinance as yf
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.preprocessing import MinMaxScaler
from sklearn.metrics import mean_squared_error, mean_absolute_error, mean_absolute_percentage_error
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import LSTM, GRU, Dense
from tensorflow.keras.optimizers import Adam
import datetime as dt

# Define the stock symbols
stock_symbols = ['GOOGL', 'META', 'GM', 'GS']

# Define the start and end dates
start_date = dt.datetime(2012, 1, 1)
end_date = dt.datetime(2023, 8, 22)

# Function to fetch historical stock price data
def fetch_stock_data(symbol, start_date, end_date):
    stock_data = yf.download(symbol, start=start_date, end=end_date)
    return stock_data

# Function to preprocess and scale data
def preprocess_data(data):
    scaler = MinMaxScaler()
    scaled_data = scaler.fit_transform(data['Close'].values.reshape(-1, 1))
    return scaled_data, scaler

# Loop through each stock symbol
all_predictions = {}
results = {}

for symbol in stock_symbols:
    # Fetch and preprocess data
    stock_data = fetch_stock_data(symbol, start_date, end_date)
    scaled_data, scaler = preprocess_data(stock_data)

    # Create sequences
    seq_length = 10
    sequences = []
```

```
targets = []
for i in range(len(scaled_data) - seq_length):
    sequences.append(scaled_data[i:i+seq_length])
    targets.append(scaled_data[i+seq_length])

sequences = np.array(sequences)
targets = np.array(targets)

# Split data into train and test sets
train_size = int(0.8 * len(sequences))
train_X, test_X = sequences[:train_size], sequences[train_size:]
train_y, test_y = targets[:train_size], targets[train_size:]

# Create combined GRU-LSTM model
model = Sequential()
model.add(LSTM(units=50, return_sequences=True, input_shape=(seq_length, 1)))
model.add(GRU(units=50))
model.add(Dense(1))
model.compile(optimizer=Adam(learning_rate=0.001), loss='mean_squared_error')

# Train the model
model.fit(train_X, train_y, epochs=50, batch_size=32, verbose=2)

# Make predictions
test_predictions = model.predict(test_X)

# Inverse transform the scaled data to original values
test_predictions = scaler.inverse_transform(test_predictions)
test_y_original = scaler.inverse_transform(test_y.reshape(-1, 1))

# Store predictions for each stock
all_predictions[symbol] = test_predictions

# Calculate evaluation metrics
rmse = np.sqrt(mean_squared_error(test_y_original, test_predictions))
mape = mean_absolute_percentage_error(test_y_original, test_predictions)
mae = mean_absolute_error(test_y_original, test_predictions)

# Store results in a dictionary
results[symbol] = {'RMSE': rmse, 'MAPE': mape, 'MAE': mae}
```

```
[**********100%**********]  1 of 1 completed
Epoch 1/50
73/73 - 6s - loss: 0.0050 - 6s/epoch - 86ms/step
Epoch 2/50
73/73 - 1s - loss: 1.2815e-04 - 550ms/epoch - 8ms/step
Epoch 3/50
73/73 - 1s - loss: 1.1899e-04 - 514ms/epoch - 7ms/step
Epoch 4/50
73/73 - 1s - loss: 1.1870e-04 - 517ms/epoch - 7ms/step
Epoch 5/50
73/73 - 0s - loss: 1.1944e-04 - 469ms/epoch - 6ms/step
Epoch 6/50
73/73 - 1s - loss: 1.1238e-04 - 512ms/epoch - 7ms/step
Epoch 7/50
73/73 - 1s - loss: 1.2242e-04 - 508ms/epoch - 7ms/step
Epoch 8/50
73/73 - 1s - loss: 1.1714e-04 - 519ms/epoch - 7ms/step
Epoch 9/50
73/73 - 0s - loss: 1.1147e-04 - 496ms/epoch - 7ms/step
Epoch 10/50
73/73 - 1s - loss: 1.2496e-04 - 513ms/epoch - 7ms/step
Epoch 11/50
73/73 - 0s - loss: 1.1556e-04 - 500ms/epoch - 7ms/step
Epoch 12/50
73/73 - 1s - loss: 1.1046e-04 - 513ms/epoch - 7ms/step
Epoch 13/50
73/73 - 0s - loss: 1.0574e-04 - 496ms/epoch - 7ms/step
Epoch 14/50
73/73 - 1s - loss: 1.0383e-04 - 525ms/epoch - 7ms/step
Epoch 15/50
73/73 - 0s - loss: 9.9659e-05 - 497ms/epoch - 7ms/step
Epoch 16/50
73/73 - 1s - loss: 1.0300e-04 - 508ms/epoch - 7ms/step
Epoch 17/50
73/73 - 0s - loss: 9.9715e-05 - 486ms/epoch - 7ms/step
Epoch 18/50
73/73 - 0s - loss: 1.0733e-04 - 481ms/epoch - 7ms/step
Epoch 19/50
73/73 - 1s - loss: 9.7687e-05 - 507ms/epoch - 7ms/step
Epoch 20/50
73/73 - 1s - loss: 1.0589e-04 - 624ms/epoch - 9ms/step
```

```
Epoch 21/50
73/73 - 1s - loss: 9.6389e-05 - 689ms/epoch - 9ms/step
Epoch 22/50
73/73 - 1s - loss: 9.4719e-05 - 692ms/epoch - 9ms/step
Epoch 23/50
73/73 - 1s - loss: 8.1281e-05 - 709ms/epoch - 10ms/step
Epoch 24/50
73/73 - 1s - loss: 8.2795e-05 - 520ms/epoch - 7ms/step
Epoch 25/50
73/73 - 1s - loss: 7.7962e-05 - 515ms/epoch - 7ms/step
Epoch 26/50
73/73 - 0s - loss: 7.3205e-05 - 494ms/epoch - 7ms/step
Epoch 27/50
73/73 - 0s - loss: 7.7566e-05 - 491ms/epoch - 7ms/step
Epoch 28/50
73/73 - 1s - loss: 7.4900e-05 - 520ms/epoch - 7ms/step
Epoch 29/50
73/73 - 1s - loss: 6.6582e-05 - 520ms/epoch - 7ms/step
Epoch 30/50
73/73 - 1s - loss: 9.2407e-05 - 521ms/epoch - 7ms/step
Epoch 31/50
73/73 - 0s - loss: 7.1572e-05 - 493ms/epoch - 7ms/step
Epoch 32/50
73/73 - 0s - loss: 6.8835e-05 - 479ms/epoch - 7ms/step
Epoch 33/50
73/73 - 0s - loss: 6.7373e-05 - 497ms/epoch - 7ms/step
Epoch 34/50
73/73 - 1s - loss: 7.5968e-05 - 518ms/epoch - 7ms/step
Epoch 35/50
73/73 - 0s - loss: 5.9545e-05 - 485ms/epoch - 7ms/step
Epoch 36/50
73/73 - 0s - loss: 6.1146e-05 - 494ms/epoch - 7ms/step
Epoch 37/50
73/73 - 0s - loss: 6.9057e-05 - 499ms/epoch - 7ms/step
Epoch 38/50
73/73 - 0s - loss: 6.4216e-05 - 471ms/epoch - 6ms/step
Epoch 39/50
73/73 - 1s - loss: 6.3521e-05 - 521ms/epoch - 7ms/step
Epoch 40/50
73/73 - 1s - loss: 6.0004e-05 - 506ms/epoch - 7ms/step
Epoch 41/50
73/73 - 0s - loss: 5.5137e-05 - 487ms/epoch - 7ms/step
```

```
Epoch 42/50
73/73 - 1s - loss: 5.4906e-05 - 545ms/epoch - 7ms/step
Epoch 43/50
73/73 - 1s - loss: 5.4365e-05 - 607ms/epoch - 8ms/step
Epoch 44/50
73/73 - 1s - loss: 5.5074e-05 - 632ms/epoch - 9ms/step
Epoch 45/50
73/73 - 1s - loss: 5.2602e-05 - 665ms/epoch - 9ms/step
Epoch 46/50
73/73 - 1s - loss: 6.7300e-05 - 673ms/epoch - 9ms/step
Epoch 47/50
73/73 - 1s - loss: 5.2001e-05 - 540ms/epoch - 7ms/step
Epoch 48/50
73/73 - 1s - loss: 4.9407e-05 - 523ms/epoch - 7ms/step
Epoch 49/50
73/73 - 1s - loss: 5.4926e-05 - 502ms/epoch - 7ms/step
Epoch 50/50
73/73 - 0s - loss: 5.0889e-05 - 488ms/epoch - 7ms/step
19/19 [==============================] - 1s 2ms/step
[**********100%**********]  1 of 1 completed
Epoch 1/50
71/71 - 3s - loss: 0.0110 - 3s/epoch - 41ms/step
Epoch 2/50
71/71 - 0s - loss: 2.3787e-04 - 476ms/epoch - 7ms/step
Epoch 3/50
71/71 - 0s - loss: 2.2421e-04 - 487ms/epoch - 7ms/step
Epoch 4/50
71/71 - 1s - loss: 2.2442e-04 - 522ms/epoch - 7ms/step
Epoch 5/50
71/71 - 0s - loss: 2.3777e-04 - 482ms/epoch - 7ms/step
Epoch 6/50
71/71 - 0s - loss: 2.2815e-04 - 472ms/epoch - 7ms/step
Epoch 7/50
71/71 - 0s - loss: 2.1511e-04 - 491ms/epoch - 7ms/step
Epoch 8/50
71/71 - 0s - loss: 2.1678e-04 - 468ms/epoch - 7ms/step
Epoch 9/50
71/71 - 1s - loss: 2.1447e-04 - 538ms/epoch - 8ms/step
Epoch 10/50
71/71 - 1s - loss: 2.2196e-04 - 658ms/epoch - 9ms/step
Epoch 11/50
71/71 - 1s - loss: 2.2473e-04 - 595ms/epoch - 8ms/step
```

```
Epoch 12/50
71/71 - 1s - loss: 2.0756e-04 - 626ms/epoch - 9ms/step
Epoch 13/50
71/71 - 1s - loss: 2.0779e-04 - 637ms/epoch - 9ms/step
Epoch 14/50
71/71 - 0s - loss: 2.0826e-04 - 461ms/epoch - 6ms/step
Epoch 15/50
71/71 - 0s - loss: 1.9593e-04 - 472ms/epoch - 7ms/step
Epoch 16/50
71/71 - 0s - loss: 1.8734e-04 - 473ms/epoch - 7ms/step
Epoch 17/50
71/71 - 0s - loss: 1.9025e-04 - 482ms/epoch - 7ms/step
Epoch 18/50
71/71 - 0s - loss: 1.9541e-04 - 480ms/epoch - 7ms/step
Epoch 19/50
71/71 - 0s - loss: 1.8930e-04 - 463ms/epoch - 7ms/step
Epoch 20/50
71/71 - 0s - loss: 1.6854e-04 - 495ms/epoch - 7ms/step
Epoch 21/50
71/71 - 0s - loss: 1.8728e-04 - 483ms/epoch - 7ms/step
Epoch 22/50
71/71 - 0s - loss: 1.8078e-04 - 457ms/epoch - 6ms/step
Epoch 23/50
71/71 - 0s - loss: 1.6640e-04 - 483ms/epoch - 7ms/step
Epoch 24/50
71/71 - 0s - loss: 1.7755e-04 - 483ms/epoch - 7ms/step
Epoch 25/50
71/71 - 0s - loss: 1.5989e-04 - 465ms/epoch - 7ms/step
Epoch 26/50
71/71 - 0s - loss: 1.7311e-04 - 465ms/epoch - 7ms/step
Epoch 27/50
71/71 - 0s - loss: 1.8526e-04 - 467ms/epoch - 7ms/step
Epoch 28/50
71/71 - 0s - loss: 1.5846e-04 - 468ms/epoch - 7ms/step
Epoch 29/50
71/71 - 0s - loss: 1.4984e-04 - 473ms/epoch - 7ms/step
Epoch 30/50
71/71 - 0s - loss: 1.4563e-04 - 457ms/epoch - 6ms/step
Epoch 31/50
71/71 - 0s - loss: 1.3739e-04 - 474ms/epoch - 7ms/step
Epoch 32/50
71/71 - 0s - loss: 1.3092e-04 - 472ms/epoch - 7ms/step
```

```
Epoch 33/50
71/71 - 0s - loss: 1.3551e-04 - 473ms/epoch - 7ms/step
Epoch 34/50
71/71 - 1s - loss: 1.4292e-04 - 558ms/epoch - 8ms/step
Epoch 35/50
71/71 - 1s - loss: 1.3582e-04 - 643ms/epoch - 9ms/step
Epoch 36/50
71/71 - 1s - loss: 1.3973e-04 - 622ms/epoch - 9ms/step
Epoch 37/50
71/71 - 1s - loss: 1.3371e-04 - 655ms/epoch - 9ms/step
Epoch 38/50
71/71 - 1s - loss: 1.3005e-04 - 574ms/epoch - 8ms/step
Epoch 39/50
71/71 - 0s - loss: 1.3082e-04 - 482ms/epoch - 7ms/step
Epoch 40/50
71/71 - 0s - loss: 1.3420e-04 - 481ms/epoch - 7ms/step
Epoch 41/50
71/71 - 0s - loss: 1.4952e-04 - 494ms/epoch - 7ms/step
Epoch 42/50
71/71 - 0s - loss: 1.1874e-04 - 499ms/epoch - 7ms/step
Epoch 43/50
71/71 - 1s - loss: 1.2968e-04 - 512ms/epoch - 7ms/step
Epoch 44/50
71/71 - 0s - loss: 1.1472e-04 - 497ms/epoch - 7ms/step
Epoch 45/50
71/71 - 0s - loss: 1.1123e-04 - 497ms/epoch - 7ms/step
Epoch 46/50
71/71 - 0s - loss: 1.1807e-04 - 482ms/epoch - 7ms/step
Epoch 47/50
71/71 - 1s - loss: 1.1092e-04 - 505ms/epoch - 7ms/step
Epoch 48/50
71/71 - 0s - loss: 1.5735e-04 - 494ms/epoch - 7ms/step
Epoch 49/50
71/71 - 0s - loss: 1.0316e-04 - 471ms/epoch - 7ms/step
Epoch 50/50
71/71 - 0s - loss: 1.0891e-04 - 485ms/epoch - 7ms/step
18/18 [==============================] - 1s 2ms/step
[**********100%%**********]  1 of 1 completed
Epoch 1/50
73/73 - 4s - loss: 0.0103 - 4s/epoch - 61ms/step
Epoch 2/50
73/73 - 1s - loss: 7.2879e-04 - 696ms/epoch - 10ms/step
```

```
Epoch 3/50
73/73 - 1s - loss: 6.8950e-04 - 572ms/epoch - 8ms/step
Epoch 4/50
73/73 - 1s - loss: 6.6620e-04 - 563ms/epoch - 8ms/step
Epoch 5/50
73/73 - 1s - loss: 6.4821e-04 - 565ms/epoch - 8ms/step
Epoch 6/50
73/73 - 1s - loss: 6.3391e-04 - 574ms/epoch - 8ms/step
Epoch 7/50
73/73 - 1s - loss: 5.9743e-04 - 593ms/epoch - 8ms/step
Epoch 8/50
73/73 - 1s - loss: 5.7978e-04 - 557ms/epoch - 8ms/step
Epoch 9/50
73/73 - 1s - loss: 5.3114e-04 - 566ms/epoch - 8ms/step
Epoch 10/50
73/73 - 1s - loss: 5.2180e-04 - 581ms/epoch - 8ms/step
Epoch 11/50
73/73 - 1s - loss: 4.8085e-04 - 563ms/epoch - 8ms/step
Epoch 12/50
73/73 - 1s - loss: 4.4540e-04 - 556ms/epoch - 8ms/step
Epoch 13/50
73/73 - 1s - loss: 4.3421e-04 - 551ms/epoch - 8ms/step
Epoch 14/50
73/73 - 1s - loss: 4.2004e-04 - 576ms/epoch - 8ms/step
Epoch 15/50
73/73 - 1s - loss: 3.7628e-04 - 553ms/epoch - 8ms/step
Epoch 16/50
73/73 - 1s - loss: 3.7539e-04 - 570ms/epoch - 8ms/step
Epoch 17/50
73/73 - 1s - loss: 3.8284e-04 - 547ms/epoch - 7ms/step
Epoch 18/50
73/73 - 1s - loss: 3.3428e-04 - 566ms/epoch - 8ms/step
Epoch 19/50
73/73 - 1s - loss: 3.3836e-04 - 589ms/epoch - 8ms/step
Epoch 20/50
73/73 - 1s - loss: 3.2527e-04 - 766ms/epoch - 10ms/step
Epoch 21/50
73/73 - 1s - loss: 3.0734e-04 - 733ms/epoch - 10ms/step
Epoch 22/50
73/73 - 1s - loss: 2.9872e-04 - 760ms/epoch - 10ms/step
Epoch 23/50
73/73 - 1s - loss: 3.1181e-04 - 605ms/epoch - 8ms/step
```

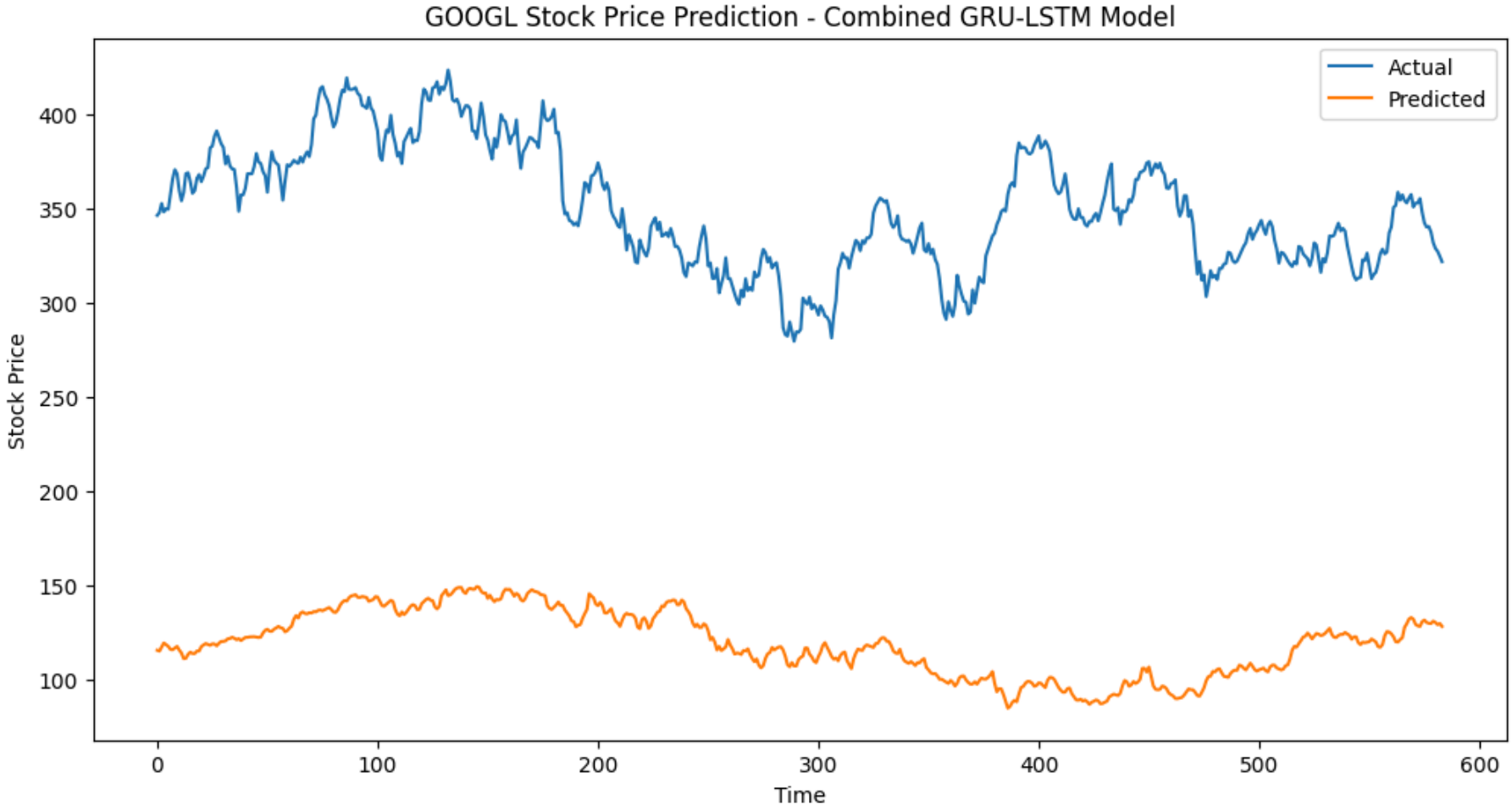
```
Epoch 24/50
73/73 - 1s - loss: 2.9962e-04 - 574ms/epoch - 8ms/step
Epoch 25/50
73/73 - 1s - loss: 2.9186e-04 - 558ms/epoch - 8ms/step
Epoch 26/50
73/73 - 1s - loss: 2.6724e-04 - 570ms/epoch - 8ms/step
Epoch 27/50
73/73 - 1s - loss: 2.6559e-04 - 578ms/epoch - 8ms/step
Epoch 28/50
73/73 - 1s - loss: 2.6531e-04 - 541ms/epoch - 7ms/step
Epoch 29/50
73/73 - 1s - loss: 2.8512e-04 - 557ms/epoch - 8ms/step
Epoch 30/50
73/73 - 1s - loss: 2.5103e-04 - 627ms/epoch - 9ms/step
Epoch 31/50
73/73 - 1s - loss: 2.4536e-04 - 698ms/epoch - 10ms/step
Epoch 32/50
73/73 - 1s - loss: 2.3983e-04 - 546ms/epoch - 7ms/step
Epoch 33/50
73/73 - 1s - loss: 2.4284e-04 - 548ms/epoch - 8ms/step
Epoch 34/50
73/73 - 1s - loss: 2.3398e-04 - 578ms/epoch - 8ms/step
Epoch 35/50
73/73 - 1s - loss: 2.3041e-04 - 586ms/epoch - 8ms/step
Epoch 36/50
73/73 - 1s - loss: 2.2773e-04 - 578ms/epoch - 8ms/step
Epoch 37/50
73/73 - 1s - loss: 2.5244e-04 - 560ms/epoch - 8ms/step
Epoch 38/50
73/73 - 1s - loss: 2.2518e-04 - 577ms/epoch - 8ms/step
Epoch 39/50
73/73 - 1s - loss: 2.2762e-04 - 601ms/epoch - 8ms/step
Epoch 40/50
73/73 - 1s - loss: 2.3072e-04 - 685ms/epoch - 9ms/step
Epoch 41/50
73/73 - 1s - loss: 2.2958e-04 - 744ms/epoch - 10ms/step
Epoch 42/50
73/73 - 1s - loss: 2.2631e-04 - 703ms/epoch - 10ms/step
Epoch 43/50
73/73 - 1s - loss: 2.2246e-04 - 671ms/epoch - 9ms/step
Epoch 44/50
73/73 - 1s - loss: 2.2471e-04 - 543ms/epoch - 7ms/step
```

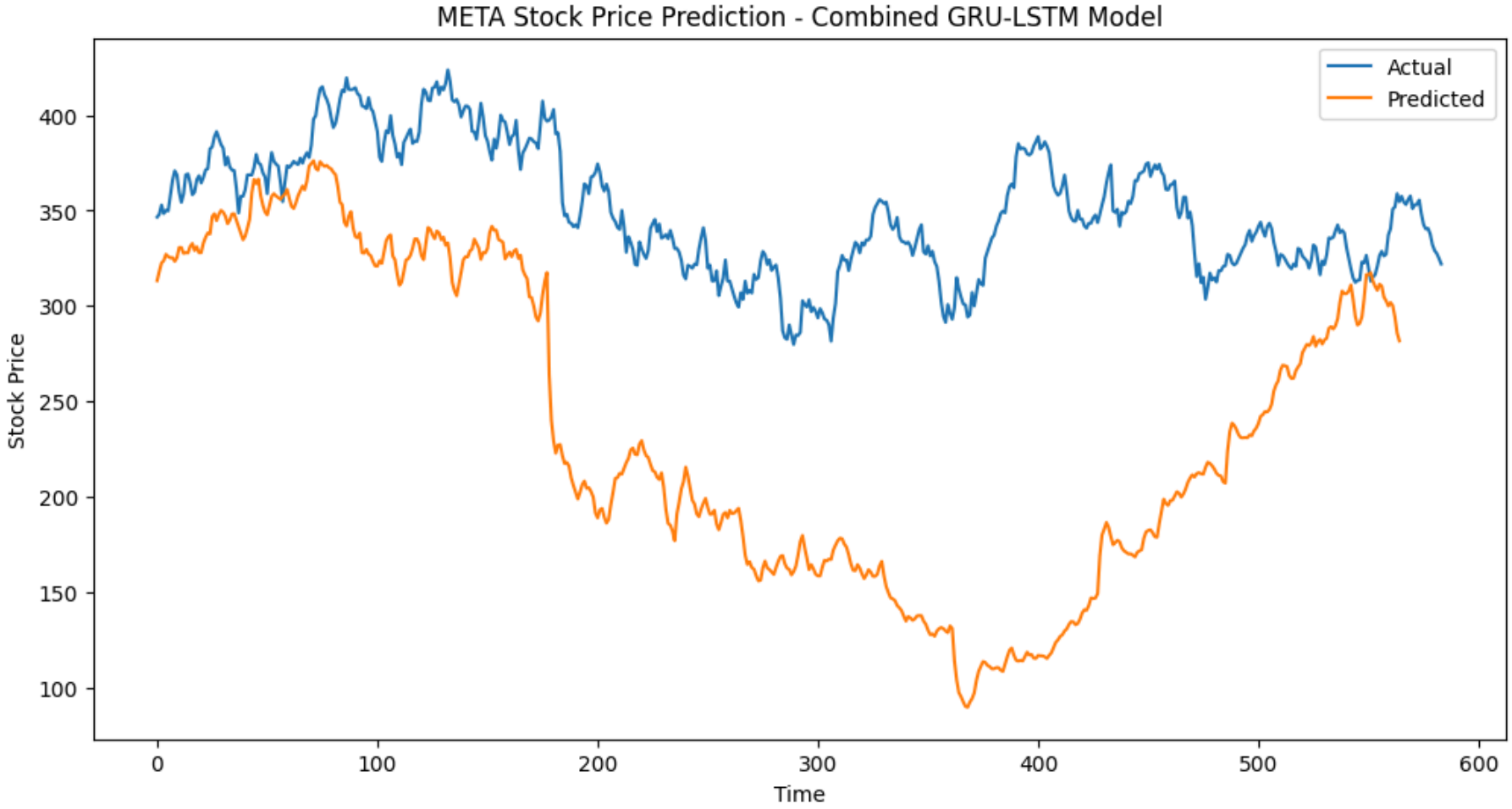
```
Epoch 45/50
73/73 - 1s - loss: 2.2556e-04 - 568ms/epoch - 8ms/step
Epoch 46/50
73/73 - 1s - loss: 2.2465e-04 - 588ms/epoch - 8ms/step
Epoch 47/50
73/73 - 1s - loss: 2.2114e-04 - 561ms/epoch - 8ms/step
Epoch 48/50
73/73 - 1s - loss: 2.2542e-04 - 571ms/epoch - 8ms/step
Epoch 49/50
73/73 - 1s - loss: 2.0710e-04 - 553ms/epoch - 8ms/step
Epoch 50/50
73/73 - 1s - loss: 2.1124e-04 - 556ms/epoch - 8ms/step
19/19 [==============================] - 1s 3ms/step
[**********100%%**********]  1 of 1 completed
Epoch 1/50
73/73 - 3s - loss: 0.0068 - 3s/epoch - 41ms/step
Epoch 2/50
73/73 - 1s - loss: 3.6386e-04 - 524ms/epoch - 7ms/step
Epoch 3/50
73/73 - 0s - loss: 3.6947e-04 - 493ms/epoch - 7ms/step
Epoch 4/50
73/73 - 1s - loss: 3.3655e-04 - 507ms/epoch - 7ms/step
Epoch 5/50
73/73 - 1s - loss: 3.2818e-04 - 672ms/epoch - 9ms/step
Epoch 6/50
73/73 - 1s - loss: 3.1844e-04 - 722ms/epoch - 10ms/step
Epoch 7/50
73/73 - 1s - loss: 3.1987e-04 - 658ms/epoch - 9ms/step
Epoch 8/50
73/73 - 1s - loss: 2.9440e-04 - 671ms/epoch - 9ms/step
Epoch 9/50
73/73 - 1s - loss: 3.0391e-04 - 514ms/epoch - 7ms/step
Epoch 10/50
73/73 - 0s - loss: 2.8501e-04 - 498ms/epoch - 7ms/step
Epoch 11/50
73/73 - 1s - loss: 2.6106e-04 - 525ms/epoch - 7ms/step
Epoch 12/50
73/73 - 0s - loss: 2.5378e-04 - 491ms/epoch - 7ms/step
Epoch 13/50
73/73 - 1s - loss: 2.2910e-04 - 513ms/epoch - 7ms/step
Epoch 14/50
73/73 - 1s - loss: 2.1868e-04 - 512ms/epoch - 7ms/step
```

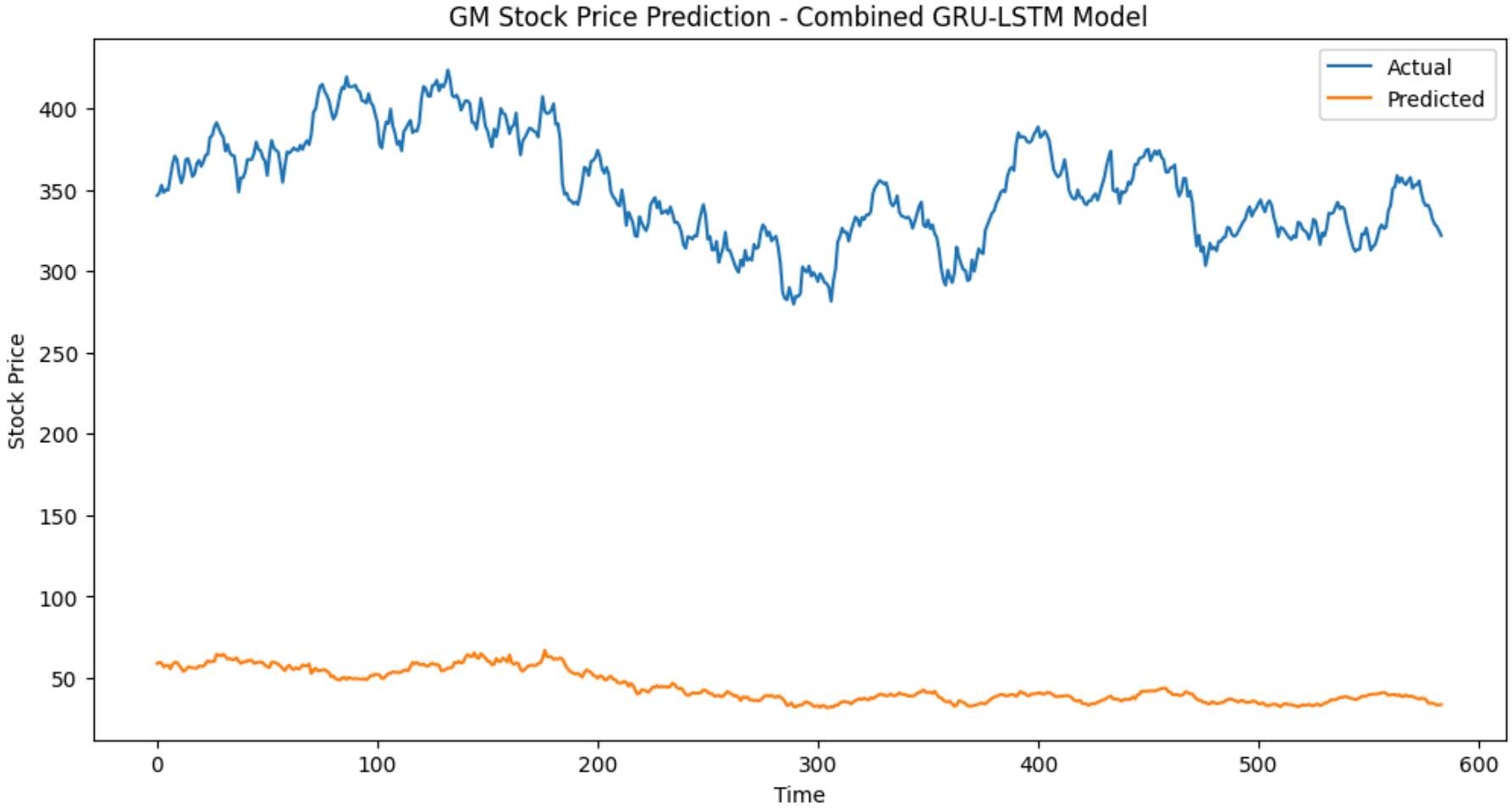
```
Epoch 15/50
73/73 - 1s - loss: 2.0916e-04 - 535ms/epoch - 7ms/step
Epoch 16/50
73/73 - 1s - loss: 1.9742e-04 - 512ms/epoch - 7ms/step
Epoch 17/50
73/73 - 1s - loss: 2.1418e-04 - 510ms/epoch - 7ms/step
Epoch 18/50
73/73 - 0s - loss: 2.4348e-04 - 487ms/epoch - 7ms/step
Epoch 19/50
73/73 - 1s - loss: 2.4152e-04 - 530ms/epoch - 7ms/step
Epoch 20/50
73/73 - 1s - loss: 1.6985e-04 - 511ms/epoch - 7ms/step
Epoch 21/50
73/73 - 1s - loss: 1.8208e-04 - 525ms/epoch - 7ms/step
Epoch 22/50
73/73 - 1s - loss: 1.8863e-04 - 504ms/epoch - 7ms/step
Epoch 23/50
73/73 - 1s - loss: 1.7902e-04 - 511ms/epoch - 7ms/step
Epoch 24/50
73/73 - 1s - loss: 1.6878e-04 - 520ms/epoch - 7ms/step
Epoch 25/50
73/73 - 0s - loss: 1.8175e-04 - 496ms/epoch - 7ms/step
Epoch 26/50
73/73 - 1s - loss: 1.5947e-04 - 513ms/epoch - 7ms/step
Epoch 27/50
73/73 - 0s - loss: 1.5295e-04 - 500ms/epoch - 7ms/step
Epoch 28/50
73/73 - 1s - loss: 1.5120e-04 - 682ms/epoch - 9ms/step
Epoch 29/50
73/73 - 1s - loss: 1.5035e-04 - 693ms/epoch - 9ms/step
Epoch 30/50
73/73 - 1s - loss: 1.5010e-04 - 715ms/epoch - 10ms/step
Epoch 31/50
73/73 - 1s - loss: 1.3690e-04 - 610ms/epoch - 8ms/step
Epoch 32/50
73/73 - 1s - loss: 1.2974e-04 - 720ms/epoch - 10ms/step
Epoch 33/50
73/73 - 1s - loss: 1.4328e-04 - 682ms/epoch - 9ms/step
Epoch 34/50
73/73 - 1s - loss: 1.3769e-04 - 524ms/epoch - 7ms/step
Epoch 35/50
73/73 - 1s - loss: 1.2948e-04 - 505ms/epoch - 7ms/step
```

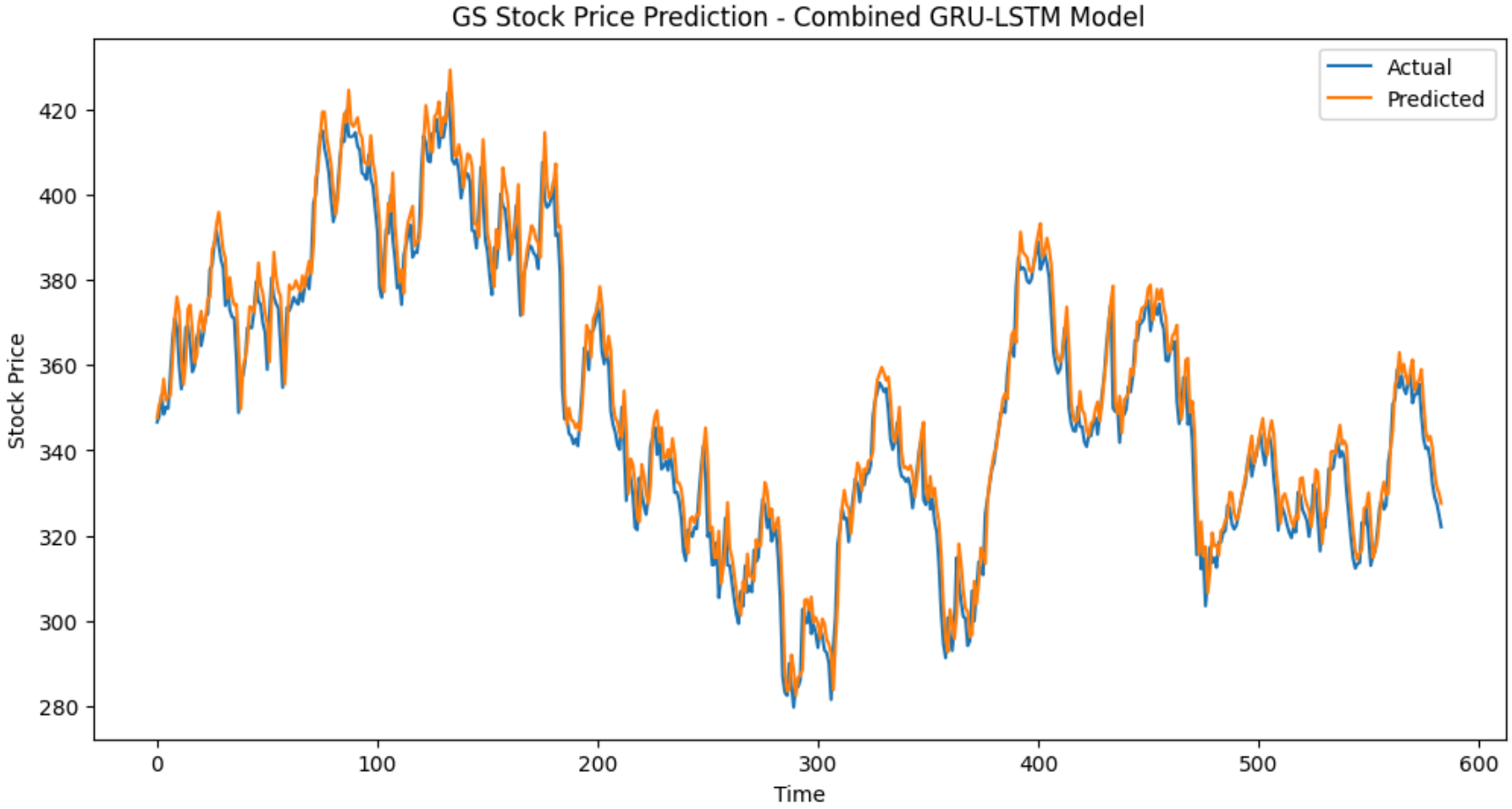
```
Epoch 36/50
73/73 - 1s - loss: 1.3588e-04 - 532ms/epoch - 7ms/step
Epoch 37/50
73/73 - 1s - loss: 1.3404e-04 - 520ms/epoch - 7ms/step
Epoch 38/50
73/73 - 1s - loss: 1.3104e-04 - 521ms/epoch - 7ms/step
Epoch 39/50
73/73 - 0s - loss: 1.1898e-04 - 485ms/epoch - 7ms/step
Epoch 40/50
73/73 - 1s - loss: 1.2073e-04 - 512ms/epoch - 7ms/step
Epoch 41/50
73/73 - 0s - loss: 1.3596e-04 - 494ms/epoch - 7ms/step
Epoch 42/50
73/73 - 1s - loss: 1.1984e-04 - 512ms/epoch - 7ms/step
Epoch 43/50
73/73 - 0s - loss: 1.2074e-04 - 498ms/epoch - 7ms/step
Epoch 44/50
73/73 - 1s - loss: 1.2934e-04 - 519ms/epoch - 7ms/step
Epoch 45/50
73/73 - 0s - loss: 1.1900e-04 - 494ms/epoch - 7ms/step
Epoch 46/50
73/73 - 1s - loss: 1.1214e-04 - 511ms/epoch - 7ms/step
Epoch 47/50
73/73 - 1s - loss: 1.1359e-04 - 515ms/epoch - 7ms/step
Epoch 48/50
73/73 - 0s - loss: 1.3336e-04 - 492ms/epoch - 7ms/step
Epoch 49/50
73/73 - 1s - loss: 1.1135e-04 - 510ms/epoch - 7ms/step
Epoch 50/50
73/73 - 1s - loss: 1.1450e-04 - 650ms/epoch - 9ms/step
19/19 [==============================] - 1s 3ms/step
```

In [ ]:
```
# Loop through each stock symbol and plot actual vs predicted prices separately
for symbol, test_predictions in all_predictions.items():
    plt.figure(figsize=(12, 6))
    plt.plot(test_y_original, label='Actual')
    plt.plot(test_predictions, label='Predicted')
    plt.title(f'{symbol} Stock Price Prediction - Combined GRU-LSTM Model')
    plt.xlabel('Time')
    plt.ylabel('Stock Price')
    plt.legend()
    plt.show()
```


GOOGL Stock Price Prediction - Combined GRU-LSTM Model
Actual
Predicted
Stock Price
Time
400
350
300
250
200
150
100
0
100
200
300
400
500
600


META Stock Price Prediction - Combined GRU-LSTM Model
Actual
Predicted
Stock Price
Time
400
350
300
250
200
150
100
0
100
200
300
400
500
600


GM Stock Price Prediction - Combined GRU-LSTM Model
Actual
Predicted
Stock Price
Time
400
350
300
250
200
150
100
50
0
100
200
300
400
500
600


GS Stock Price Prediction - Combined GRU-LSTM Model
Actual
Predicted
Stock Price
Time
420
400
380
360
340
320
300
280
0
100
200
300
400
500
600

In [ ]:
```
# Print evaluation results for each symbol
for symbol, metrics in results.items():
    print(f"Results for {symbol}:")
    print("RMSE:", metrics['RMSE'], "MAPE:", metrics['MAPE'], "MAE:", metrics['MAE'])
```

```
Results for GOOGL:
RMSE: 2.4591179752260772 MAPE: 0.016308834297867683 MAE: 1.8869515118533617
Results for META:
RMSE: 8.054111705632936 MAPE: 0.026060339100702552 MAE: 5.674015632561878
Results for GM:
RMSE: 1.222418378789646 MAPE: 0.02159144946078588 MAE: 0.9348136921451516
Results for GS:
RMSE: 6.6734205279854155 MAPE: 0.015076789948695673 MAE: 5.228696744735928
```

In [ ]:
```
pip install scikit-learn tensorflow scikit-learn keras
```

```
Requirement already satisfied: scikit-learn in /usr/local/lib/python3.10/dist-packages (1.2.2)
Requirement already satisfied: tensorflow in /usr/local/lib/python3.10/dist-packages (2.14.0)
Requirement already satisfied: keras in /usr/local/lib/python3.10/dist-packages (2.14.0)
Requirement already satisfied: numpy>=1.17.3 in /usr/local/lib/python3.10/dist-packages (from scikit-lear
n) (1.23.5)
Requirement already satisfied: scipy>=1.3.2 in /usr/local/lib/python3.10/dist-packages (from scikit-learn)
(1.11.3)
Requirement already satisfied: joblib>=1.1.1 in /usr/local/lib/python3.10/dist-packages (from scikit-lear
n) (1.3.2)
Requirement already satisfied: threadpoolctl>=2.0.0 in /usr/local/lib/python3.10/dist-packages (from sciki
t-learn) (3.2.0)
Requirement already satisfied: absl-py>=1.0.0 in /usr/local/lib/python3.10/dist-packages (from tensorflow)
(1.4.0)
Requirement already satisfied: astunparse>=1.6.0 in /usr/local/lib/python3.10/dist-packages (from tensorfl
ow) (1.6.3)
Requirement already satisfied: flatbuffers>=23.5.26 in /usr/local/lib/python3.10/dist-packages (from tenso
rflow) (23.5.26)
Requirement already satisfied: gast!=0.5.0,!=0.5.1,!=0.5.2,>=0.2.1 in /usr/local/lib/python3.10/dist-packa
ges (from tensorflow) (0.5.4)
Requirement already satisfied: google-pasta>=0.1.1 in /usr/local/lib/python3.10/dist-packages (from tensor
flow) (0.2.0)
Requirement already satisfied: h5py>=2.9.0 in /usr/local/lib/python3.10/dist-packages (from tensorflow)
(3.9.0)
Requirement already satisfied: libclang>=13.0.0 in /usr/local/lib/python3.10/dist-packages (from tensorflo
w) (16.0.6)
Requirement already satisfied: ml-dtypes==0.2.0 in /usr/local/lib/python3.10/dist-packages (from tensorflo
w) (0.2.0)
Requirement already satisfied: opt-einsum>=2.3.2 in /usr/local/lib/python3.10/dist-packages (from tensorfl
ow) (3.3.0)
Requirement already satisfied: packaging in /usr/local/lib/python3.10/dist-packages (from tensorflow) (23.
2)
Requirement already satisfied: protobuf!=4.21.0,!=4.21.1,!=4.21.2,!=4.21.3,!=4.21.4,!=4.21.5,<5.0.0dev,>=
3.20.3 in /usr/local/lib/python3.10/dist-packages (from tensorflow) (3.20.3)
Requirement already satisfied: setuptools in /usr/local/lib/python3.10/dist-packages (from tensorflow) (6
7.7.2)
Requirement already satisfied: six>=1.12.0 in /usr/local/lib/python3.10/dist-packages (from tensorflow)
(1.16.0)
Requirement already satisfied: termcolor>=1.1.0 in /usr/local/lib/python3.10/dist-packages (from tensorflo
w) (2.3.0)
Requirement already satisfied: typing-extensions>=3.6.6 in /usr/local/lib/python3.10/dist-packages (from t
ensorflow) (4.5.0)
```

```
Requirement already satisfied: wrapt<1.15,>=1.11.0 in /usr/local/lib/python3.10/dist-packages (from tensor
flow) (1.14.1)
Requirement already satisfied: tensorflow-io-gcs-filesystem>=0.23.1 in /usr/local/lib/python3.10/dist-pack
ages (from tensorflow) (0.34.0)
Requirement already satisfied: grpcio<2.0,>=1.24.3 in /usr/local/lib/python3.10/dist-packages (from tensor
flow) (1.59.2)
Requirement already satisfied: tensorboard<2.15,>=2.14 in /usr/local/lib/python3.10/dist-packages (from te
nsorflow) (2.14.1)
Requirement already satisfied: tensorflow-estimator<2.15,>=2.14.0 in /usr/local/lib/python3.10/dist-packag
es (from tensorflow) (2.14.0)
Requirement already satisfied: wheel<1.0,>=0.23.0 in /usr/local/lib/python3.10/dist-packages (from astunpa
rse>=1.6.0->tensorflow) (0.41.3)
Requirement already satisfied: google-auth<3,>=1.6.3 in /usr/local/lib/python3.10/dist-packages (from tens
orboard<2.15,>=2.14->tensorflow) (2.17.3)
Requirement already satisfied: google-auth-oauthlib<1.1,>=0.5 in /usr/local/lib/python3.10/dist-packages
(from tensorboard<2.15,>=2.14->tensorflow) (1.0.0)
Requirement already satisfied: markdown>=2.6.8 in /usr/local/lib/python3.10/dist-packages (from tensorboar
d<2.15,>=2.14->tensorflow) (3.5.1)
Requirement already satisfied: requests<3,>=2.21.0 in /usr/local/lib/python3.10/dist-packages (from tensor
board<2.15,>=2.14->tensorflow) (2.31.0)
Requirement already satisfied: tensorboard-data-server<0.8.0,>=0.7.0 in /usr/local/lib/python3.10/dist-pac
kages (from tensorboard<2.15,>=2.14->tensorflow) (0.7.2)
Requirement already satisfied: werkzeug>=1.0.1 in /usr/local/lib/python3.10/dist-packages (from tensorboar
d<2.15,>=2.14->tensorflow) (3.0.1)
Requirement already satisfied: cachetools<6.0,>=2.0.0 in /usr/local/lib/python3.10/dist-packages (from goo
gle-auth<3,>=1.6.3->tensorboard<2.15,>=2.14->tensorflow) (5.3.2)
Requirement already satisfied: pyasn1-modules>=0.2.1 in /usr/local/lib/python3.10/dist-packages (from goog
le-auth<3,>=1.6.3->tensorboard<2.15,>=2.14->tensorflow) (0.3.0)
Requirement already satisfied: rsa<5,>=3.1.4 in /usr/local/lib/python3.10/dist-packages (from google-auth<
3,>=1.6.3->tensorboard<2.15,>=2.14->tensorflow) (4.9)
Requirement already satisfied: requests-oauthlib>=0.7.0 in /usr/local/lib/python3.10/dist-packages (from g
oogle-auth-oauthlib<1.1,>=0.5->tensorboard<2.15,>=2.14->tensorflow) (1.3.1)
Requirement already satisfied: charset-normalizer<4,>=2 in /usr/local/lib/python3.10/dist-packages (from r
equests<3,>=2.21.0->tensorboard<2.15,>=2.14->tensorflow) (3.3.2)
Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.10/dist-packages (from requests<3,>=
2.21.0->tensorboard<2.15,>=2.14->tensorflow) (3.4)
Requirement already satisfied: urllib3<3,>=1.21.1 in /usr/local/lib/python3.10/dist-packages (from request
s<3,>=2.21.0->tensorboard<2.15,>=2.14->tensorflow) (2.0.7)
Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.10/dist-packages (from request
s<3,>=2.21.0->tensorboard<2.15,>=2.14->tensorflow) (2023.7.22)
Requirement already satisfied: MarkupSafe>=2.1.1 in /usr/local/lib/python3.10/dist-packages (from werkzeug
>=1.0.1->tensorboard<2.15,>=2.14->tensorflow) (2.1.3)
```

```
Requirement already satisfied: pyasn1<0.6.0,>=0.4.6 in /usr/local/lib/python3.10/dist-packages (from pyasn
1-modules>=0.2.1->google-auth<3,>=1.6.3->tensorboard<2.15,>=2.14->tensorflow) (0.5.0)
Requirement already satisfied: oauthlib>=3.0.0 in /usr/local/lib/python3.10/dist-packages (from requests-o
authlib>=0.7.0->google-auth-oauthlib<1.1,>=0.5->tensorboard<2.15,>=2.14->tensorflow) (3.2.2)
```

In [ ]:
```
pip install tensorflow
```

```
Requirement already satisfied: tensorflow in /usr/local/lib/python3.10/dist-packages (2.14.0)
Requirement already satisfied: absl-py>=1.0.0 in /usr/local/lib/python3.10/dist-packages (from tensorflow)
(1.4.0)
Requirement already satisfied: astunparse>=1.6.0 in /usr/local/lib/python3.10/dist-packages (from tensorfl
ow) (1.6.3)
Requirement already satisfied: flatbuffers>=23.5.26 in /usr/local/lib/python3.10/dist-packages (from tenso
rflow) (23.5.26)
Requirement already satisfied: gast!=0.5.0,!=0.5.1,!=0.5.2,>=0.2.1 in /usr/local/lib/python3.10/dist-packa
ges (from tensorflow) (0.5.4)
Requirement already satisfied: google-pasta>=0.1.1 in /usr/local/lib/python3.10/dist-packages (from tensor
flow) (0.2.0)
Requirement already satisfied: h5py>=2.9.0 in /usr/local/lib/python3.10/dist-packages (from tensorflow)
(3.9.0)
Requirement already satisfied: libclang>=13.0.0 in /usr/local/lib/python3.10/dist-packages (from tensorflo
w) (16.0.6)
Requirement already satisfied: ml-dtypes==0.2.0 in /usr/local/lib/python3.10/dist-packages (from tensorflo
w) (0.2.0)
Requirement already satisfied: numpy>=1.23.5 in /usr/local/lib/python3.10/dist-packages (from tensorflow)
(1.23.5)
Requirement already satisfied: opt-einsum>=2.3.2 in /usr/local/lib/python3.10/dist-packages (from tensorfl
ow) (3.3.0)
Requirement already satisfied: packaging in /usr/local/lib/python3.10/dist-packages (from tensorflow) (23.
2)
Requirement already satisfied: protobuf!=4.21.0,!=4.21.1,!=4.21.2,!=4.21.3,!=4.21.4,!=4.21.5,<5.0.0dev,>=
3.20.3 in /usr/local/lib/python3.10/dist-packages (from tensorflow) (3.20.3)
Requirement already satisfied: setuptools in /usr/local/lib/python3.10/dist-packages (from tensorflow) (6
7.7.2)
Requirement already satisfied: six>=1.12.0 in /usr/local/lib/python3.10/dist-packages (from tensorflow)
(1.16.0)
Requirement already satisfied: termcolor>=1.1.0 in /usr/local/lib/python3.10/dist-packages (from tensorflo
w) (2.3.0)
Requirement already satisfied: typing-extensions>=3.6.6 in /usr/local/lib/python3.10/dist-packages (from t
ensorflow) (4.5.0)
Requirement already satisfied: wrapt<1.15,>=1.11.0 in /usr/local/lib/python3.10/dist-packages (from tensor
flow) (1.14.1)
Requirement already satisfied: tensorflow-io-gcs-filesystem>=0.23.1 in /usr/local/lib/python3.10/dist-pack
ages (from tensorflow) (0.34.0)
Requirement already satisfied: grpcio<2.0,>=1.24.3 in /usr/local/lib/python3.10/dist-packages (from tensor
flow) (1.59.2)
Requirement already satisfied: tensorboard<2.15,>=2.14 in /usr/local/lib/python3.10/dist-packages (from te
nsorflow) (2.14.1)
```

```
Requirement already satisfied: tensorflow-estimator<2.15,>=2.14.0 in /usr/local/lib/python3.10/dist-packag
es (from tensorflow) (2.14.0)
Requirement already satisfied: keras<2.15,>=2.14.0 in /usr/local/lib/python3.10/dist-packages (from tensor
flow) (2.14.0)
Requirement already satisfied: wheel<1.0,>=0.23.0 in /usr/local/lib/python3.10/dist-packages (from astunpa
rse>=1.6.0->tensorflow) (0.41.3)
Requirement already satisfied: google-auth<3,>=1.6.3 in /usr/local/lib/python3.10/dist-packages (from tens
orboard<2.15,>=2.14->tensorflow) (2.17.3)
Requirement already satisfied: google-auth-oauthlib<1.1,>=0.5 in /usr/local/lib/python3.10/dist-packages
(from tensorboard<2.15,>=2.14->tensorflow) (1.0.0)
Requirement already satisfied: markdown>=2.6.8 in /usr/local/lib/python3.10/dist-packages (from tensorboar
d<2.15,>=2.14->tensorflow) (3.5.1)
Requirement already satisfied: requests<3,>=2.21.0 in /usr/local/lib/python3.10/dist-packages (from tensor
board<2.15,>=2.14->tensorflow) (2.31.0)
Requirement already satisfied: tensorboard-data-server<0.8.0,>=0.7.0 in /usr/local/lib/python3.10/dist-pac
kages (from tensorboard<2.15,>=2.14->tensorflow) (0.7.2)
Requirement already satisfied: werkzeug>=1.0.1 in /usr/local/lib/python3.10/dist-packages (from tensorboar
d<2.15,>=2.14->tensorflow) (3.0.1)
Requirement already satisfied: cachetools<6.0,>=2.0.0 in /usr/local/lib/python3.10/dist-packages (from goo
gle-auth<3,>=1.6.3->tensorboard<2.15,>=2.14->tensorflow) (5.3.2)
Requirement already satisfied: pyasn1-modules>=0.2.1 in /usr/local/lib/python3.10/dist-packages (from goog
le-auth<3,>=1.6.3->tensorboard<2.15,>=2.14->tensorflow) (0.3.0)
Requirement already satisfied: rsa<5,>=3.1.4 in /usr/local/lib/python3.10/dist-packages (from google-auth<
3,>=1.6.3->tensorboard<2.15,>=2.14->tensorflow) (4.9)
Requirement already satisfied: requests-oauthlib>=0.7.0 in /usr/local/lib/python3.10/dist-packages (from g
oogle-auth-oauthlib<1.1,>=0.5->tensorboard<2.15,>=2.14->tensorflow) (1.3.1)
Requirement already satisfied: charset-normalizer<4,>=2 in /usr/local/lib/python3.10/dist-packages (from r
equests<3,>=2.21.0->tensorboard<2.15,>=2.14->tensorflow) (3.3.2)
Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.10/dist-packages (from requests<3,>=
2.21.0->tensorboard<2.15,>=2.14->tensorflow) (3.4)
Requirement already satisfied: urllib3<3,>=1.21.1 in /usr/local/lib/python3.10/dist-packages (from request
s<3,>=2.21.0->tensorboard<2.15,>=2.14->tensorflow) (2.0.7)
Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.10/dist-packages (from request
s<3,>=2.21.0->tensorboard<2.15,>=2.14->tensorflow) (2023.7.22)
Requirement already satisfied: MarkupSafe>=2.1.1 in /usr/local/lib/python3.10/dist-packages (from werkzeug
>=1.0.1->tensorboard<2.15,>=2.14->tensorflow) (2.1.3)
Requirement already satisfied: pyasn1<0.6.0,>=0.4.6 in /usr/local/lib/python3.10/dist-packages (from pyasn
1-modules>=0.2.1->google-auth<3,>=1.6.3->tensorboard<2.15,>=2.14->tensorflow) (0.5.0)
Requirement already satisfied: oauthlib>=3.0.0 in /usr/local/lib/python3.10/dist-packages (from requests-o
authlib>=0.7.0->google-auth-oauthlib<1.1,>=0.5->tensorboard<2.15,>=2.14->tensorflow) (3.2.2)
```

In [ ]:
```
!pip install tensorflow
```

```
Requirement already satisfied: tensorflow in /usr/local/lib/python3.10/dist-packages (2.14.0)
Requirement already satisfied: absl-py>=1.0.0 in /usr/local/lib/python3.10/dist-packages (from tensorflow)
(1.4.0)
Requirement already satisfied: astunparse>=1.6.0 in /usr/local/lib/python3.10/dist-packages (from tensorfl
ow) (1.6.3)
Requirement already satisfied: flatbuffers>=23.5.26 in /usr/local/lib/python3.10/dist-packages (from tenso
rflow) (23.5.26)
Requirement already satisfied: gast!=0.5.0,!=0.5.1,!=0.5.2,>=0.2.1 in /usr/local/lib/python3.10/dist-packa
ges (from tensorflow) (0.5.4)
Requirement already satisfied: google-pasta>=0.1.1 in /usr/local/lib/python3.10/dist-packages (from tensor
flow) (0.2.0)
Requirement already satisfied: h5py>=2.9.0 in /usr/local/lib/python3.10/dist-packages (from tensorflow)
(3.9.0)
Requirement already satisfied: libclang>=13.0.0 in /usr/local/lib/python3.10/dist-packages (from tensorflo
w) (16.0.6)
Requirement already satisfied: ml-dtypes==0.2.0 in /usr/local/lib/python3.10/dist-packages (from tensorflo
w) (0.2.0)
Requirement already satisfied: numpy>=1.23.5 in /usr/local/lib/python3.10/dist-packages (from tensorflow)
(1.23.5)
Requirement already satisfied: opt-einsum>=2.3.2 in /usr/local/lib/python3.10/dist-packages (from tensorfl
ow) (3.3.0)
Requirement already satisfied: packaging in /usr/local/lib/python3.10/dist-packages (from tensorflow) (23.
2)
Requirement already satisfied: protobuf!=4.21.0,!=4.21.1,!=4.21.2,!=4.21.3,!=4.21.4,!=4.21.5,<5.0.0dev,>=
3.20.3 in /usr/local/lib/python3.10/dist-packages (from tensorflow) (3.20.3)
Requirement already satisfied: setuptools in /usr/local/lib/python3.10/dist-packages (from tensorflow) (6
7.7.2)
Requirement already satisfied: six>=1.12.0 in /usr/local/lib/python3.10/dist-packages (from tensorflow)
(1.16.0)
Requirement already satisfied: termcolor>=1.1.0 in /usr/local/lib/python3.10/dist-packages (from tensorflo
w) (2.3.0)
Requirement already satisfied: typing-extensions>=3.6.6 in /usr/local/lib/python3.10/dist-packages (from t
ensorflow) (4.5.0)
Requirement already satisfied: wrapt<1.15,>=1.11.0 in /usr/local/lib/python3.10/dist-packages (from tensor
flow) (1.14.1)
Requirement already satisfied: tensorflow-io-gcs-filesystem>=0.23.1 in /usr/local/lib/python3.10/dist-pack
ages (from tensorflow) (0.34.0)
Requirement already satisfied: grpcio<2.0,>=1.24.3 in /usr/local/lib/python3.10/dist-packages (from tensor
flow) (1.59.2)
Requirement already satisfied: tensorboard<2.15,>=2.14 in /usr/local/lib/python3.10/dist-packages (from te
nsorflow) (2.14.1)
```

```
Requirement already satisfied: tensorflow-estimator<2.15,>=2.14.0 in /usr/local/lib/python3.10/dist-packag
es (from tensorflow) (2.14.0)
Requirement already satisfied: keras<2.15,>=2.14.0 in /usr/local/lib/python3.10/dist-packages (from tensor
flow) (2.14.0)
Requirement already satisfied: wheel<1.0,>=0.23.0 in /usr/local/lib/python3.10/dist-packages (from astunpa
rse>=1.6.0->tensorflow) (0.41.3)
Requirement already satisfied: google-auth<3,>=1.6.3 in /usr/local/lib/python3.10/dist-packages (from tens
orboard<2.15,>=2.14->tensorflow) (2.17.3)
Requirement already satisfied: google-auth-oauthlib<1.1,>=0.5 in /usr/local/lib/python3.10/dist-packages
(from tensorboard<2.15,>=2.14->tensorflow) (1.0.0)
Requirement already satisfied: markdown>=2.6.8 in /usr/local/lib/python3.10/dist-packages (from tensorboar
d<2.15,>=2.14->tensorflow) (3.5.1)
Requirement already satisfied: requests<3,>=2.21.0 in /usr/local/lib/python3.10/dist-packages (from tensor
board<2.15,>=2.14->tensorflow) (2.31.0)
Requirement already satisfied: tensorboard-data-server<0.8.0,>=0.7.0 in /usr/local/lib/python3.10/dist-pac
kages (from tensorboard<2.15,>=2.14->tensorflow) (0.7.2)
Requirement already satisfied: werkzeug>=1.0.1 in /usr/local/lib/python3.10/dist-packages (from tensorboar
d<2.15,>=2.14->tensorflow) (3.0.1)
Requirement already satisfied: cachetools<6.0,>=2.0.0 in /usr/local/lib/python3.10/dist-packages (from goo
gle-auth<3,>=1.6.3->tensorboard<2.15,>=2.14->tensorflow) (5.3.2)
Requirement already satisfied: pyasn1-modules>=0.2.1 in /usr/local/lib/python3.10/dist-packages (from goog
le-auth<3,>=1.6.3->tensorboard<2.15,>=2.14->tensorflow) (0.3.0)
Requirement already satisfied: rsa<5,>=3.1.4 in /usr/local/lib/python3.10/dist-packages (from google-auth<
3,>=1.6.3->tensorboard<2.15,>=2.14->tensorflow) (4.9)
Requirement already satisfied: requests-oauthlib>=0.7.0 in /usr/local/lib/python3.10/dist-packages (from g
oogle-auth-oauthlib<1.1,>=0.5->tensorboard<2.15,>=2.14->tensorflow) (1.3.1)
Requirement already satisfied: charset-normalizer<4,>=2 in /usr/local/lib/python3.10/dist-packages (from r
equests<3,>=2.21.0->tensorboard<2.15,>=2.14->tensorflow) (3.3.2)
Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.10/dist-packages (from requests<3,>=
2.21.0->tensorboard<2.15,>=2.14->tensorflow) (3.4)
Requirement already satisfied: urllib3<3,>=1.21.1 in /usr/local/lib/python3.10/dist-packages (from request
s<3,>=2.21.0->tensorboard<2.15,>=2.14->tensorflow) (2.0.7)
Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.10/dist-packages (from request
s<3,>=2.21.0->tensorboard<2.15,>=2.14->tensorflow) (2023.7.22)
Requirement already satisfied: MarkupSafe>=2.1.1 in /usr/local/lib/python3.10/dist-packages (from werkzeug
>=1.0.1->tensorboard<2.15,>=2.14->tensorflow) (2.1.3)
Requirement already satisfied: pyasn1<0.6.0,>=0.4.6 in /usr/local/lib/python3.10/dist-packages (from pyasn
1-modules>=0.2.1->google-auth<3,>=1.6.3->tensorboard<2.15,>=2.14->tensorflow) (0.5.0)
Requirement already satisfied: oauthlib>=3.0.0 in /usr/local/lib/python3.10/dist-packages (from requests-o
authlib>=0.7.0->google-auth-oauthlib<1.1,>=0.5->tensorboard<2.15,>=2.14->tensorflow) (3.2.2)
```

In [ ]:
```
#hypertuned

import yfinance as yf
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.preprocessing import MinMaxScaler
from sklearn.metrics import mean_squared_error
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import LSTM, GRU, Dense
from tensorflow.keras.optimizers import Adam
import datetime as dt

# Define the stock symbols
stock_symbols = ['GOOGL', 'META', 'GM', 'GS']

# Define the start and end dates
start_date = dt.datetime(2012, 1, 1)
end_date = dt.datetime(2023, 8, 22)

# Function to fetch historical stock price data
def fetch_stock_data(symbol, start_date, end_date):
    stock_data = yf.download(symbol, start=start_date, end=end_date)
    return stock_data

# Function to preprocess and scale data
def preprocess_data(data):
    scaler = MinMaxScaler()
    scaled_data = scaler.fit_transform(data['Close'].values.reshape(-1, 1))
    return scaled_data, scaler

# Function to create sequences from data
def create_sequences(data, seq_length):
    sequences = []
    targets = []
    for i in range(len(data) - seq_length):
        sequence = data[i:i+seq_length]
        target = data[i+seq_length]
        sequences.append(sequence)
        targets.append(target)
    return np.array(sequences), np.array(targets)
```

```
# Loop through each stock symbol
results = {}
for symbol in stock_symbols:
    # Fetch and preprocess data
    stock_data = fetch_stock_data(symbol, start_date, end_date)
    scaled_data, scaler = preprocess_data(stock_data)
    seq_length = 10  # Set sequence length
    sequences, targets = create_sequences(scaled_data, seq_length)

    # Split the data into training and testing sets
    train_size = int(0.8 * len(sequences))
    train_X, test_X = sequences[:train_size], sequences[train_size:]
    train_y, test_y = targets[:train_size], targets[train_size:]

    # Hyperparameters for tuning
    units = [50, 64, 128]
    learning_rates = [0.001, 0.01, 0.1]

    best_rmse = float('inf')
    best_params = {}

    for unit in units:
        for lr in learning_rates:
            # Create combined GRU-LSTM model
            model = Sequential()
            model.add(LSTM(units=unit, return_sequences=True, input_shape=(seq_length, 1)))
            model.add(GRU(units=unit))
            model.add(Dense(1))
            model.compile(optimizer=Adam(learning_rate=lr), loss='mean_squared_error')

            # Train the model
            model.fit(train_X, train_y, epochs=50, batch_size=32, verbose=0)

            # Make predictions on test set
            test_predictions = model.predict(test_X)
            test_predictions = scaler.inverse_transform(test_predictions)
            test_y_original = scaler.inverse_transform(test_y.reshape(-1, 1))

            # Calculate RMSE
            rmse = np.sqrt(mean_squared_error(test_y_original, test_predictions))
```

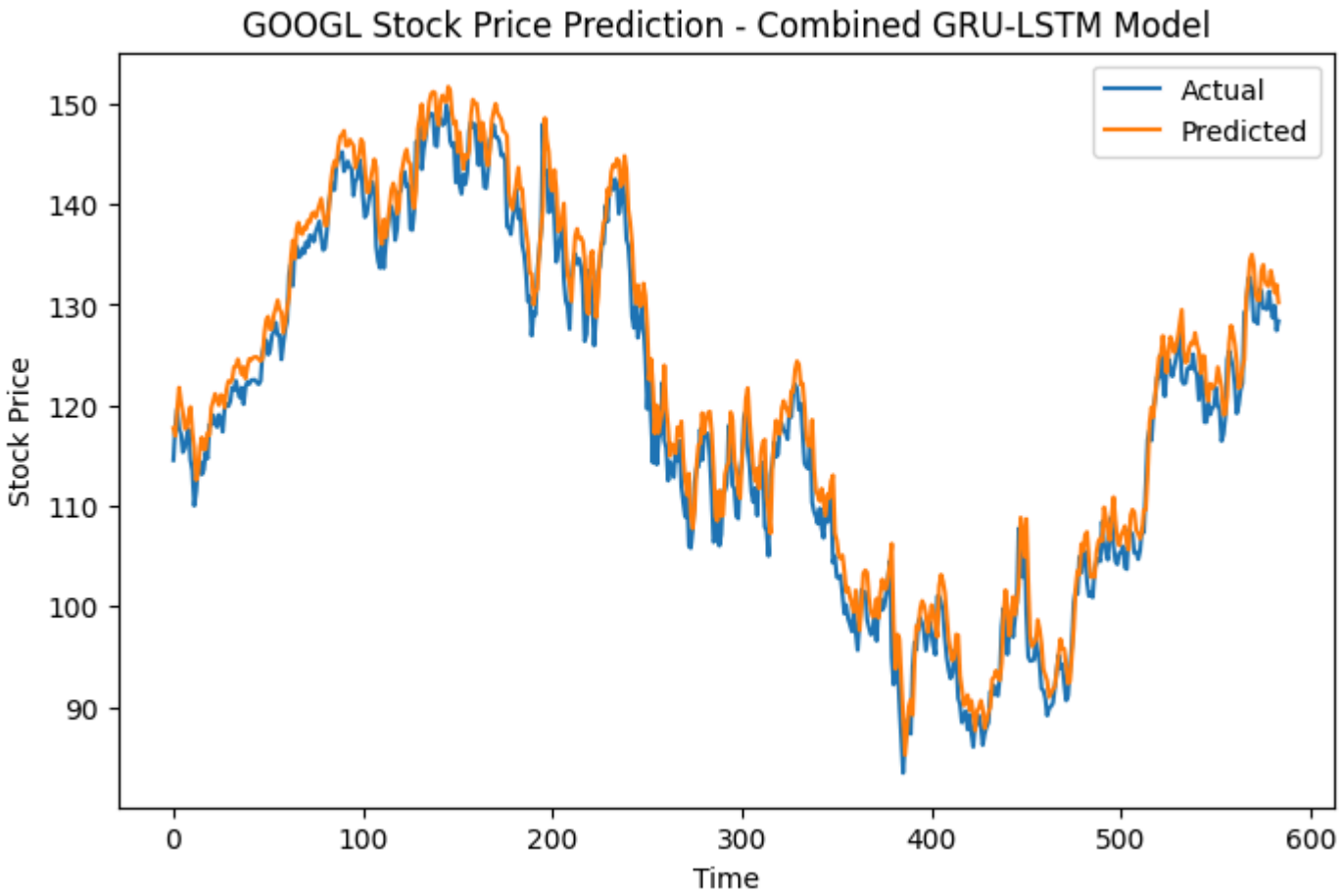
```
            # Check if this combination yields a better result
            if rmse < best_rmse:
                best_rmse = rmse
                best_params = {'units': unit, 'learning_rate': lr}

    # Store best parameters and evaluation metrics
    results[symbol] = {'Best Parameters': best_params, 'RMSE': best_rmse}
```

```
[**********100%**********]  1 of 1 completed
19/19 [==============================] - 1s 2ms/step
19/19 [==============================] - 1s 2ms/step
19/19 [==============================] - 1s 3ms/step
19/19 [==============================] - 1s 3ms/step
19/19 [==============================] - 1s 3ms/step
19/19 [==============================] - 1s 3ms/step
19/19 [==============================] - 1s 4ms/step
19/19 [==============================] - 1s 6ms/step
19/19 [==============================] - 1s 5ms/step
[**********100%**********]  1 of 1 completed
18/18 [==============================] - 1s 3ms/step
18/18 [==============================] - 0s 2ms/step
18/18 [==============================] - 0s 2ms/step
18/18 [==============================] - 1s 3ms/step
18/18 [==============================] - 0s 2ms/step
18/18 [==============================] - 1s 3ms/step
18/18 [==============================] - 1s 4ms/step
18/18 [==============================] - 1s 5ms/step
18/18 [==============================] - 1s 5ms/step
[**********100%**********]  1 of 1 completed
19/19 [==============================] - 1s 2ms/step
19/19 [==============================] - 1s 2ms/step
19/19 [==============================] - 1s 3ms/step
19/19 [==============================] - 1s 2ms/step
19/19 [==============================] - 1s 3ms/step
19/19 [==============================] - 1s 2ms/step
19/19 [==============================] - 1s 5ms/step
19/19 [==============================] - 1s 8ms/step
19/19 [==============================] - 1s 5ms/step
[**********100%**********]  1 of 1 completed
19/19 [==============================] - 1s 2ms/step
19/19 [==============================] - 0s 2ms/step
19/19 [==============================] - 1s 2ms/step
19/19 [==============================] - 1s 2ms/step
19/19 [==============================] - 1s 2ms/step
19/19 [==============================] - 1s 2ms/step
19/19 [==============================] - 1s 5ms/step
19/19 [==============================] - 1s 4ms/step
19/19 [==============================] - 1s 8ms/step
```

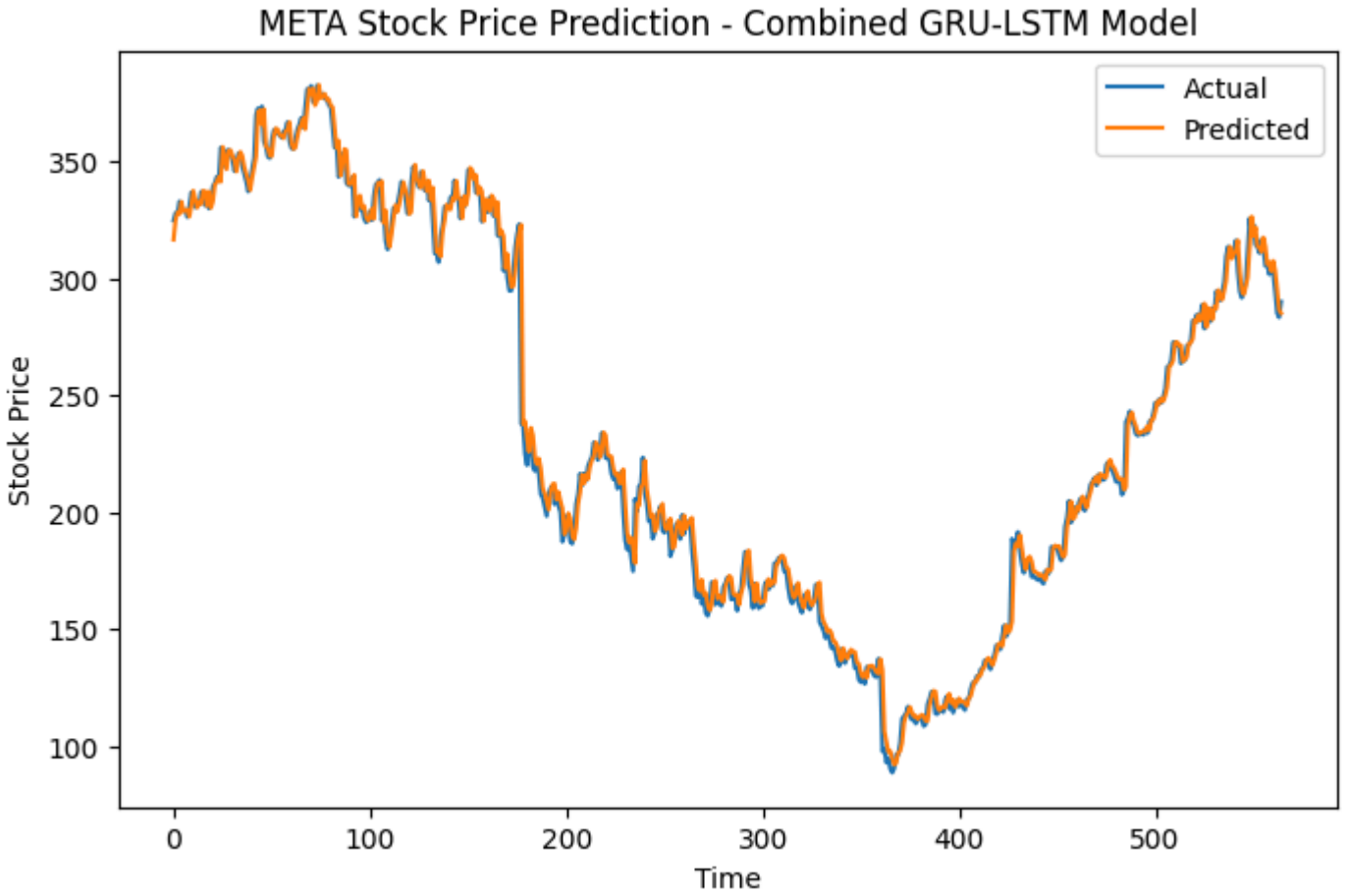
In [ ]:
```
for symbol, data in results.items():
    # Fetch and preprocess data
    stock_data = fetch_stock_data(symbol, start_date, end_date)
    scaled_data, scaler = preprocess_data(stock_data)
    seq_length = 10  # Set sequence length
    sequences, targets = create_sequences(scaled_data, seq_length)

    # Split the data into training and testing sets
    train_size = int(0.8 * len(sequences))
    train_X, test_X = sequences[:train_size], sequences[train_size:]
    train_y, test_y = targets[:train_size], targets[train_size:]

    # Create combined GRU-LSTM model with best parameters
    best_units = data['Best Parameters']['units']
    best_lr = data['Best Parameters']['learning_rate']

    model = Sequential()
    model.add(LSTM(units=best_units, return_sequences=True, input_shape=(seq_length, 1)))
    model.add(GRU(units=best_units))
    model.add(Dense(1))
    model.compile(optimizer=Adam(learning_rate=best_lr), loss='mean_squared_error')

    # Train the model
    model.fit(train_X, train_y, epochs=50, batch_size=32, verbose=0)

    # Make predictions on test set
    test_predictions = model.predict(test_X)
    test_predictions = scaler.inverse_transform(test_predictions)
    test_y_original = scaler.inverse_transform(test_y.reshape(-1, 1))

    # Plot actual vs predicted values
    plt.figure(figsize=(8, 5))
    plt.plot(test_y_original, label="Actual")
    plt.plot(test_predictions, label="Predicted")
    plt.title(f"{symbol} Stock Price Prediction - Combined GRU-LSTM Model")
    plt.xlabel("Time")
    plt.ylabel("Stock Price")
    plt.legend()
    plt.show()

    # Calculate evaluation metrics
```

```
    rmse = np.sqrt(mean_squared_error(test_y_original, test_predictions))
    print(f"Results for {symbol}:")
    print("RMSE:", rmse)
    # You can calculate and print other metrics like MAE, MAPE if needed
```

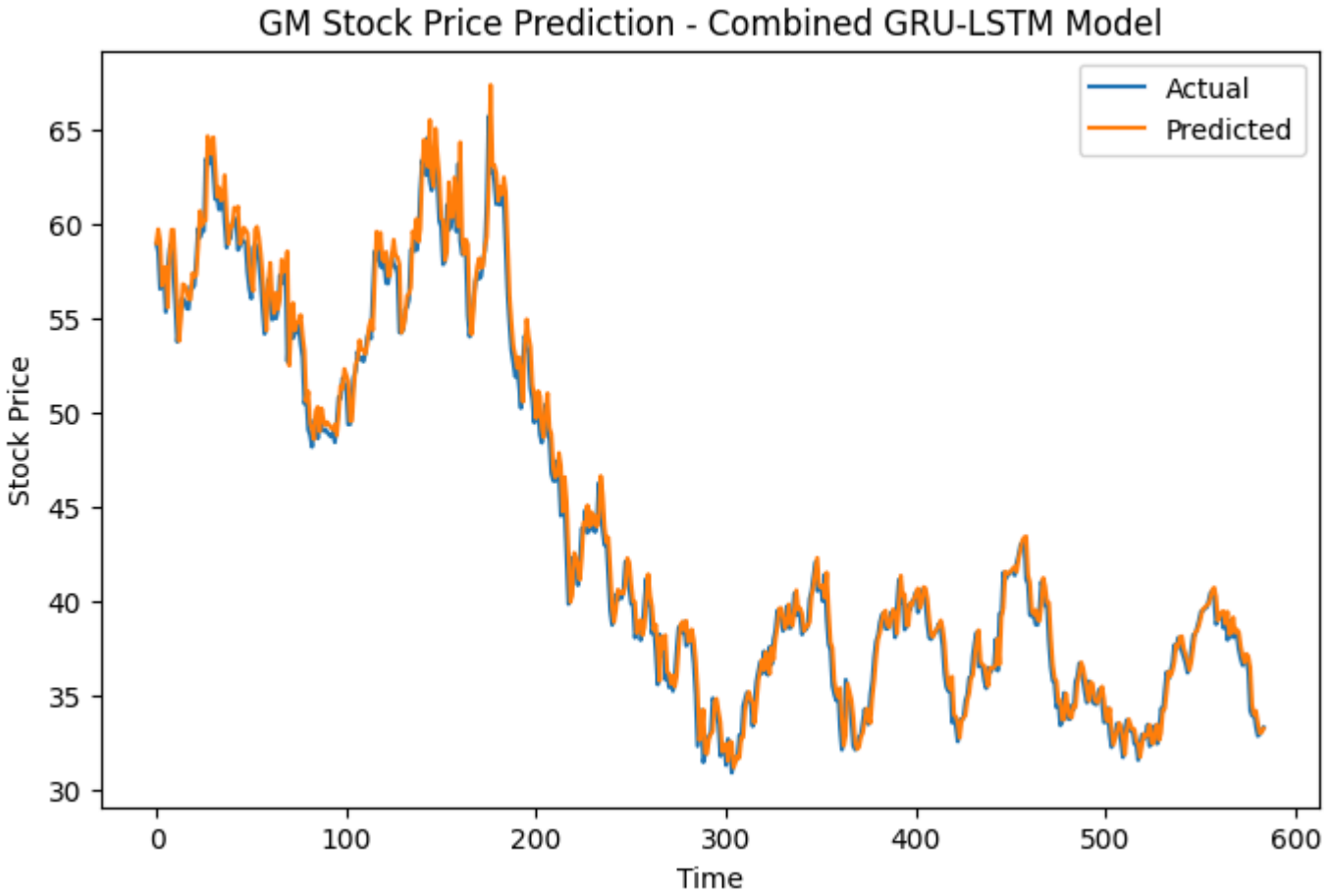
```
[**********100%%**********]  1 of 1 completed
19/19 [==============================] - 1s 3ms/step
```



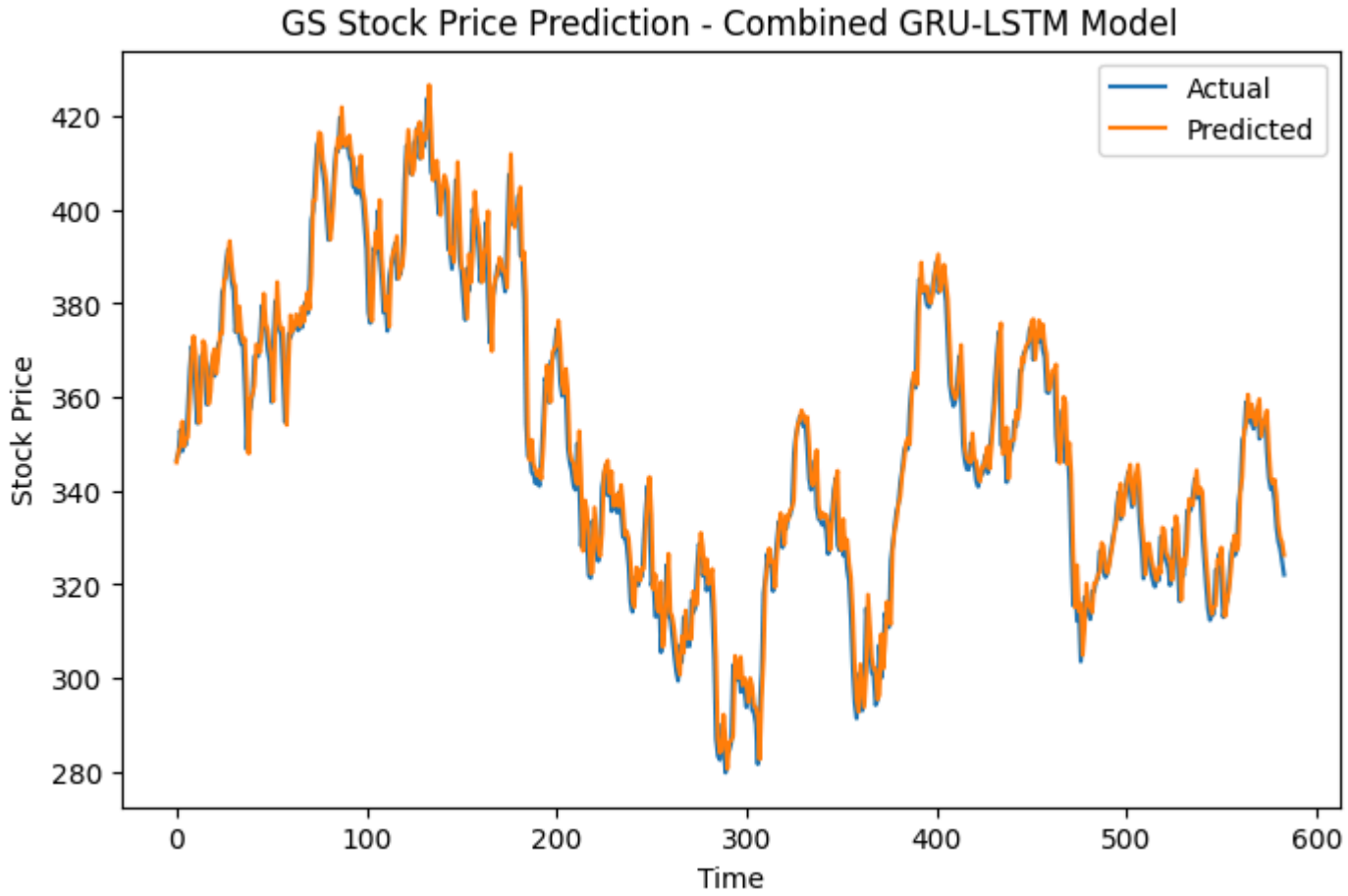

```
Results for GOOGL:
RMSE: 3.202613890943262
[**********100%%**********]  1 of 1 completed
18/18 [==============================] - 1s 7ms/step
```




```
Results for META:
RMSE: 7.1617673895872835
[**********************100%%**********************]  1 of 1 completed
19/19 [==============================] - 1s 2ms/step
```




```
Results for GM:
RMSE: 1.1848616746029572
[**********************100%%**********************]  1 of 1 completed
19/19 [==============================] - 1s 7ms/step
```




```
Results for GS:
RMSE: 5.957081561957631
```

In [ ]:
```
for symbol, data in results.items():
    # Fetch and preprocess data
    stock_data = fetch_stock_data(symbol, start_date, end_date)
    scaled_data, scaler = preprocess_data(stock_data)
    seq_length = 10  # Set sequence length
    sequences, targets = create_sequences(scaled_data, seq_length)

    # Split the data into training and testing sets
    train_size = int(0.8 * len(sequences))
    train_X, test_X = sequences[:train_size], sequences[train_size:]
    train_y, test_y = targets[:train_size], targets[train_size:]

    # Create combined GRU-LSTM model with best parameters
    best_units = data['Best Parameters']['units']
    best_lr = data['Best Parameters']['learning_rate']

    model = Sequential()
    model.add(LSTM(units=best_units, return_sequences=True, input_shape=(seq_length, 1)))
    model.add(GRU(units=best_units))
    model.add(Dense(1))
    model.compile(optimizer=Adam(learning_rate=best_lr), loss='mean_squared_error')

    # Train the model
    model.fit(train_X, train_y, epochs=50, batch_size=32, verbose=0)

    # Make predictions on test set
    test_predictions = model.predict(test_X)
    test_predictions = scaler.inverse_transform(test_predictions)
    test_y_original = scaler.inverse_transform(test_y.reshape(-1, 1))

    # Calculate evaluation metrics
    rmse = np.sqrt(mean_squared_error(test_y_original, test_predictions))
    mae = mean_absolute_error(test_y_original, test_predictions)
    mape = mean_absolute_percentage_error(test_y_original, test_predictions)

    # Print evaluation metrics
    print(f"Results for {symbol}:")
    print("RMSE:", rmse)
    print("MAE:", mae)
    print("MAPE:", mape)
```

```
[**********100%**********]  1 of 1 completed
19/19 [==============================] - 1s 2ms/step
Results for GOOGL:
RMSE: 4.962632615293644
MAE: 4.431765804552052
MAPE: 0.03755553932882122
[**********100%**********]  1 of 1 completed
18/18 [==============================] - 1s 5ms/step
Results for META:
RMSE: 7.283891816357845
MAE: 4.772291227357577
MAPE: 0.02226529937597775
[**********100%**********]  1 of 1 completed
19/19 [==============================] - 1s 2ms/step
Results for GM:
RMSE: 1.1794986743299471
MAE: 0.8900096579773785
MAPE: 0.02045185562779007
[**********100%**********]  1 of 1 completed
19/19 [==============================] - 1s 5ms/step
Results for GS:
RMSE: 9.901581239286438
MAE: 8.523258784045911
MAPE: 0.024188490336679545
```