

```
In [1]: # Install TensorFlow
# !pip install -q tensorflow-gpu==2.0.0-beta1

try:
    %tensorflow_version 2.x # Colab only.
except Exception:
    pass

import tensorflow as tf
print(tf.__version__)
```

Colab only includes TensorFlow 2.x; %tensorflow_version has no effect.
2.12.0

```
In [2]: # More imports
from tensorflow.keras.layers import Input, LSTM, GRU, SimpleRNN, Dense, GlobalMaxPo
from tensorflow.keras.models import Model
from tensorflow.keras.optimizers import SGD, Adam

import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.preprocessing import StandardScaler
```

```
In [3]: # yes, you can read dataframes from URLs!
df = pd.read_csv('https://raw.githubusercontent.com/lazyprogrammer/machine_learning
```

```
In [4]: df.head()
```

```
Out[4]:
```

	date	open	high	low	close	volume	Name
0	2013-02-08	27.920	28.325	27.920	28.185	7146296	SBUX
1	2013-02-11	28.260	28.260	27.930	28.070	5457354	SBUX
2	2013-02-12	28.000	28.275	27.975	28.130	8665592	SBUX
3	2013-02-13	28.230	28.230	27.750	27.915	7022056	SBUX
4	2013-02-14	27.765	27.905	27.675	27.775	8899188	SBUX

```
In [5]: df.tail()
```

```
Out[5]:
```

	date	open	high	low	close	volume	Name
1254	2018-02-01	56.280	56.42	55.89	56.00	14690146	SBUX
1255	2018-02-02	55.900	56.32	55.70	55.77	15358909	SBUX
1256	2018-02-05	55.530	56.26	54.57	54.69	16059955	SBUX
1257	2018-02-06	53.685	56.06	53.56	55.61	17415065	SBUX
1258	2018-02-07	55.080	55.43	54.44	54.46	13927022	SBUX

```
In [6]: # Start by doing the WRONG thing - trying to predict the price itself
series = df['close'].values.reshape(-1, 1)
```

```
In [7]: # Normalize the data
# Note: I didn't think about where the true boundary is, this is just approx.
scaler = StandardScaler()
scaler.fit(series[:len(series) // 2])
series = scaler.transform(series).flatten()
```

```
In [8]: ### build the dataset
# Let's see if we can use T past values to predict the next value
T = 10
D = 1
X = []
Y = []
for t in range(len(series) - T):
    x = series[t:t+T]
    X.append(x)
    y = series[t+T]
    Y.append(y)

X = np.array(X).reshape(-1, T, 1) # Now the data should be N x T x D
Y = np.array(Y)
N = len(X)
print("X.shape", X.shape, "Y.shape", Y.shape)
```

X.shape (1249, 10, 1) Y.shape (1249,)

```
In [9]: ### try autoregressive RNN model
i = Input(shape=(T, 1))
x = LSTM(5)(i)
x = Dense(1)(x)
model = Model(i, x)
model.compile(
    loss='mse',
    optimizer=Adam(learning_rate=0.1),
)

# train the RNN
r = model.fit(
    X[:-N//2], Y[:-N//2],
    epochs=80,
    validation_data=(X[-N//2:], Y[-N//2:]),
)
```

```
Epoch 1/80
20/20 [=====] - 12s 30ms/step - loss: 0.0930 - val_loss: 0.2677
Epoch 2/80
20/20 [=====] - 0s 7ms/step - loss: 0.0167 - val_loss: 0.0607
Epoch 3/80
20/20 [=====] - 0s 8ms/step - loss: 0.0076 - val_loss: 0.0528
Epoch 4/80
20/20 [=====] - 0s 9ms/step - loss: 0.0057 - val_loss: 0.0541
Epoch 5/80
20/20 [=====] - 0s 7ms/step - loss: 0.0058 - val_loss: 0.0491
Epoch 6/80
20/20 [=====] - 0s 8ms/step - loss: 0.0087 - val_loss: 0.0429
Epoch 7/80
20/20 [=====] - 0s 9ms/step - loss: 0.0079 - val_loss: 0.0536
Epoch 8/80
20/20 [=====] - 0s 8ms/step - loss: 0.0087 - val_loss: 0.0421
Epoch 9/80
20/20 [=====] - 0s 7ms/step - loss: 0.0062 - val_loss: 0.0407
Epoch 10/80
20/20 [=====] - 0s 7ms/step - loss: 0.0055 - val_loss: 0.0375
Epoch 11/80
20/20 [=====] - 0s 8ms/step - loss: 0.0056 - val_loss: 0.0435
Epoch 12/80
20/20 [=====] - 0s 7ms/step - loss: 0.0054 - val_loss: 0.0514
Epoch 13/80
20/20 [=====] - 0s 7ms/step - loss: 0.0054 - val_loss: 0.0606
Epoch 14/80
20/20 [=====] - 0s 8ms/step - loss: 0.0080 - val_loss: 0.0851
Epoch 15/80
20/20 [=====] - 0s 8ms/step - loss: 0.0060 - val_loss: 0.0498
Epoch 16/80
20/20 [=====] - 0s 9ms/step - loss: 0.0057 - val_loss: 0.0386
Epoch 17/80
20/20 [=====] - 0s 7ms/step - loss: 0.0060 - val_loss: 0.0433
Epoch 18/80
20/20 [=====] - 0s 7ms/step - loss: 0.0063 - val_loss: 0.0524
Epoch 19/80
20/20 [=====] - 0s 7ms/step - loss: 0.0061 - val_loss: 0.
```

```
0489
Epoch 20/80
20/20 [=====] - 0s 14ms/step - loss: 0.0054 - val_loss:
0.0398
Epoch 21/80
20/20 [=====] - 0s 10ms/step - loss: 0.0057 - val_loss:
0.0309
Epoch 22/80
20/20 [=====] - 0s 15ms/step - loss: 0.0054 - val_loss:
0.0353
Epoch 23/80
20/20 [=====] - 0s 11ms/step - loss: 0.0075 - val_loss:
0.0445
Epoch 24/80
20/20 [=====] - 0s 11ms/step - loss: 0.0053 - val_loss:
0.0295
Epoch 25/80
20/20 [=====] - 0s 10ms/step - loss: 0.0054 - val_loss:
0.0358
Epoch 26/80
20/20 [=====] - 0s 10ms/step - loss: 0.0058 - val_loss:
0.0285
Epoch 27/80
20/20 [=====] - 0s 11ms/step - loss: 0.0091 - val_loss:
0.0324
Epoch 28/80
20/20 [=====] - 0s 10ms/step - loss: 0.0092 - val_loss:
0.0227
Epoch 29/80
20/20 [=====] - 0s 10ms/step - loss: 0.0074 - val_loss:
0.0439
Epoch 30/80
20/20 [=====] - 0s 9ms/step - loss: 0.0090 - val_loss: 0.
0629
Epoch 31/80
20/20 [=====] - 0s 7ms/step - loss: 0.0081 - val_loss: 0.
0437
Epoch 32/80
20/20 [=====] - 0s 8ms/step - loss: 0.0060 - val_loss: 0.
0266
Epoch 33/80
20/20 [=====] - 0s 7ms/step - loss: 0.0058 - val_loss: 0.
0202
Epoch 34/80
20/20 [=====] - 0s 8ms/step - loss: 0.0055 - val_loss: 0.
0224
Epoch 35/80
20/20 [=====] - 0s 8ms/step - loss: 0.0059 - val_loss: 0.
0861
Epoch 36/80
20/20 [=====] - 0s 7ms/step - loss: 0.0067 - val_loss: 0.
0175
Epoch 37/80
20/20 [=====] - 0s 7ms/step - loss: 0.0060 - val_loss: 0.
0169
Epoch 38/80
```

```
20/20 [=====] - 0s 7ms/step - loss: 0.0056 - val_loss: 0.0206
Epoch 39/80
20/20 [=====] - 0s 8ms/step - loss: 0.0052 - val_loss: 0.0338
Epoch 40/80
20/20 [=====] - 0s 8ms/step - loss: 0.0066 - val_loss: 0.0521
Epoch 41/80
20/20 [=====] - 0s 7ms/step - loss: 0.0060 - val_loss: 0.0140
Epoch 42/80
20/20 [=====] - 0s 7ms/step - loss: 0.0066 - val_loss: 0.0169
Epoch 43/80
20/20 [=====] - 0s 7ms/step - loss: 0.0060 - val_loss: 0.0232
Epoch 44/80
20/20 [=====] - 0s 8ms/step - loss: 0.0059 - val_loss: 0.0176
Epoch 45/80
20/20 [=====] - 0s 7ms/step - loss: 0.0055 - val_loss: 0.0281
Epoch 46/80
20/20 [=====] - 0s 8ms/step - loss: 0.0054 - val_loss: 0.0193
Epoch 47/80
20/20 [=====] - 0s 7ms/step - loss: 0.0071 - val_loss: 0.0654
Epoch 48/80
20/20 [=====] - 0s 8ms/step - loss: 0.0076 - val_loss: 0.0331
Epoch 49/80
20/20 [=====] - 0s 8ms/step - loss: 0.0057 - val_loss: 0.0162
Epoch 50/80
20/20 [=====] - 0s 7ms/step - loss: 0.0067 - val_loss: 0.0217
Epoch 51/80
20/20 [=====] - 0s 7ms/step - loss: 0.0066 - val_loss: 0.0449
Epoch 52/80
20/20 [=====] - 0s 9ms/step - loss: 0.0081 - val_loss: 0.0153
Epoch 53/80
20/20 [=====] - 0s 7ms/step - loss: 0.0075 - val_loss: 0.0342
Epoch 54/80
20/20 [=====] - 0s 8ms/step - loss: 0.0074 - val_loss: 0.0195
Epoch 55/80
20/20 [=====] - 0s 7ms/step - loss: 0.0060 - val_loss: 0.0178
Epoch 56/80
20/20 [=====] - 0s 8ms/step - loss: 0.0064 - val_loss: 0.0186
```

Epoch 57/80
20/20 [=====] - 0s 8ms/step - loss: 0.0059 - val_loss: 0.0172

Epoch 58/80
20/20 [=====] - 0s 8ms/step - loss: 0.0052 - val_loss: 0.0194

Epoch 59/80
20/20 [=====] - 0s 7ms/step - loss: 0.0052 - val_loss: 0.0268

Epoch 60/80
20/20 [=====] - 0s 7ms/step - loss: 0.0052 - val_loss: 0.0190

Epoch 61/80
20/20 [=====] - 0s 7ms/step - loss: 0.0058 - val_loss: 0.0212

Epoch 62/80
20/20 [=====] - 0s 7ms/step - loss: 0.0063 - val_loss: 0.0543

Epoch 63/80
20/20 [=====] - 0s 8ms/step - loss: 0.0074 - val_loss: 0.1126

Epoch 64/80
20/20 [=====] - 0s 7ms/step - loss: 0.0077 - val_loss: 0.0411

Epoch 65/80
20/20 [=====] - 0s 8ms/step - loss: 0.0059 - val_loss: 0.0235

Epoch 66/80
20/20 [=====] - 0s 9ms/step - loss: 0.0054 - val_loss: 0.0240

Epoch 67/80
20/20 [=====] - 0s 7ms/step - loss: 0.0054 - val_loss: 0.0283

Epoch 68/80
20/20 [=====] - 0s 8ms/step - loss: 0.0062 - val_loss: 0.0183

Epoch 69/80
20/20 [=====] - 0s 7ms/step - loss: 0.0067 - val_loss: 0.0282

Epoch 70/80
20/20 [=====] - 0s 8ms/step - loss: 0.0065 - val_loss: 0.0170

Epoch 71/80
20/20 [=====] - 0s 7ms/step - loss: 0.0052 - val_loss: 0.0166

Epoch 72/80
20/20 [=====] - 0s 8ms/step - loss: 0.0057 - val_loss: 0.0186

Epoch 73/80
20/20 [=====] - 0s 7ms/step - loss: 0.0053 - val_loss: 0.0236

Epoch 74/80
20/20 [=====] - 0s 7ms/step - loss: 0.0055 - val_loss: 0.0222

Epoch 75/80
20/20 [=====] - 0s 9ms/step - loss: 0.0093 - val_loss: 0.

```

0336
Epoch 76/80
20/20 [=====] - 0s 8ms/step - loss: 0.0104 - val_loss: 0.
0442
Epoch 77/80
20/20 [=====] - 0s 9ms/step - loss: 0.0171 - val_loss: 0.
0617
Epoch 78/80
20/20 [=====] - 0s 9ms/step - loss: 0.0112 - val_loss: 0.
0395
Epoch 79/80
20/20 [=====] - 0s 7ms/step - loss: 0.0075 - val_loss: 0.
0202
Epoch 80/80
20/20 [=====] - 0s 8ms/step - loss: 0.0071 - val_loss: 0.
0695

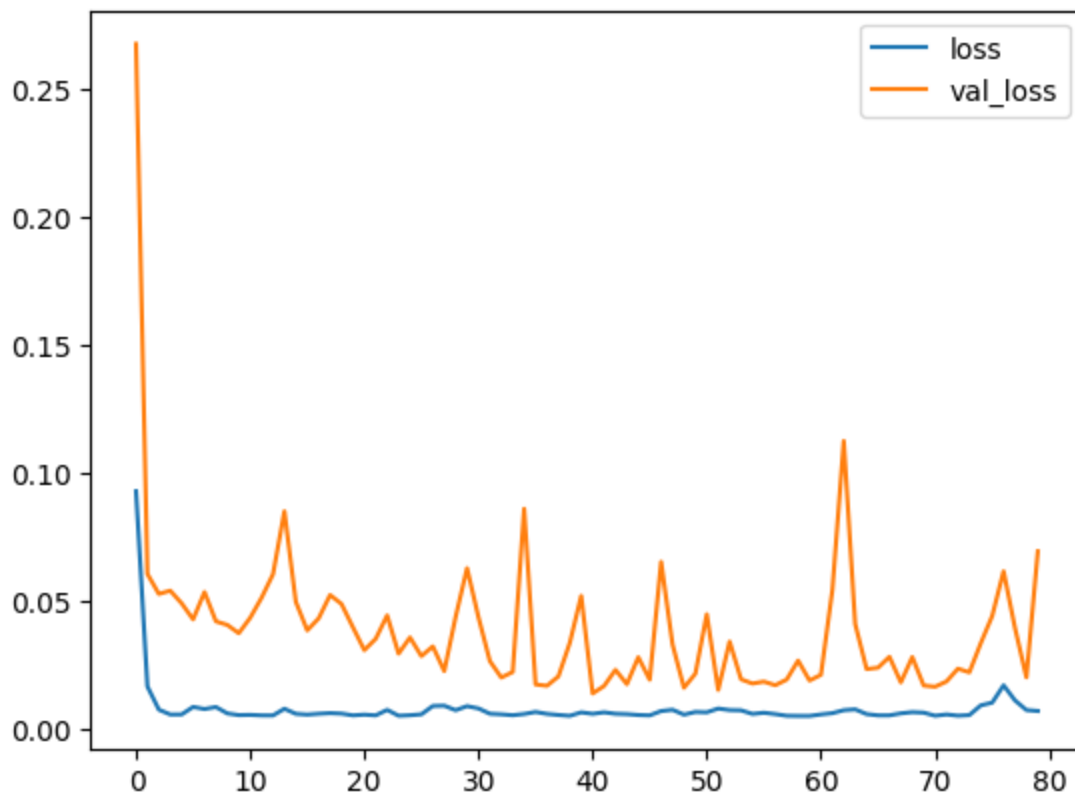
```

```

In [10]: # Plot loss per iteration
import matplotlib.pyplot as plt
plt.plot(r.history['loss'], label='loss')
plt.plot(r.history['val_loss'], label='val_loss')
plt.legend()

```

Out[10]: <matplotlib.legend.Legend at 0x7f5bac8af700>



```

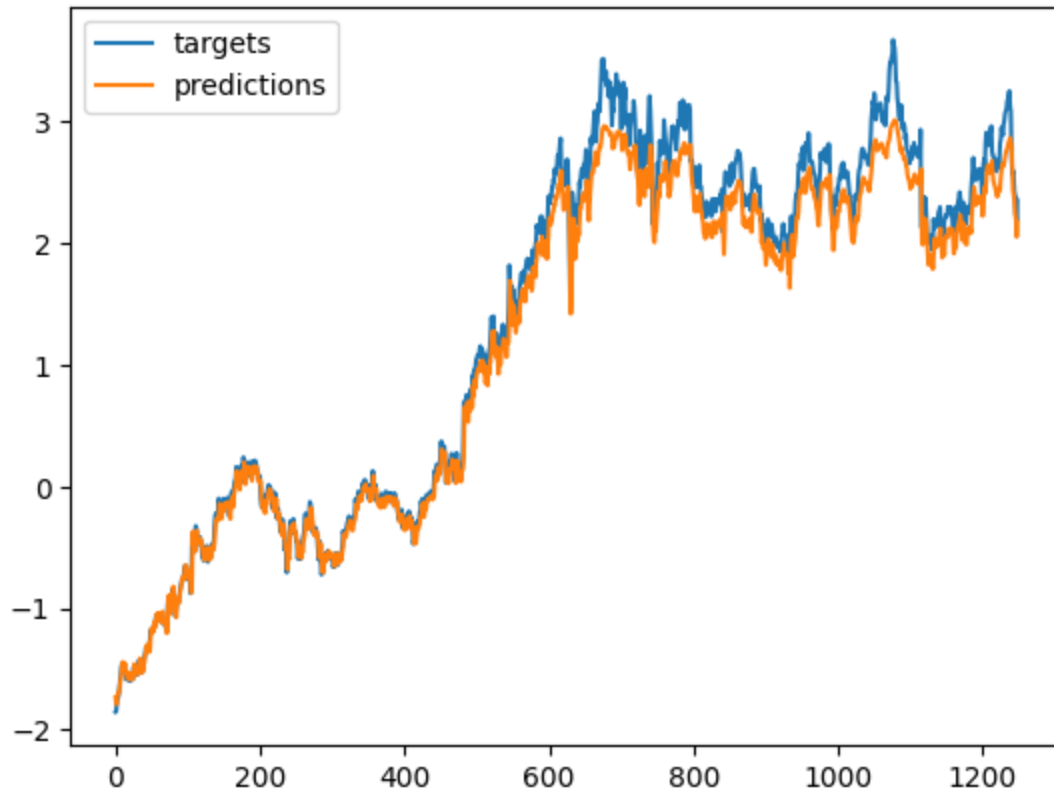
In [11]: # One-step forecast using true targets
outputs = model.predict(X)
print(outputs.shape)
predictions = outputs[:,0]

plt.plot(Y, label='targets')

```

```
plt.plot(predictions, label='predictions')
plt.legend()
plt.show()
```

40/40 [=====] - 0s 2ms/step
(1249, 1)



```
In [12]: # Multi-step forecast
validation_target = Y[-N//2:]
validation_predictions = []

# first validation input
last_x = X[-N//2] # 1-D array of length T

while len(validation_predictions) < len(validation_target):
    p = model.predict(last_x.reshape(1, T, 1))[0,0] # 1x1 array -> scalar

    # update the predictions list
    validation_predictions.append(p)

    # make the new input
    last_x = np.roll(last_x, -1)
    last_x[-1] = p

plt.plot(validation_target, label='forecast target')
plt.plot(validation_predictions, label='forecast prediction')
plt.legend()
```



```
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 23ms/step
1/1 [=====] - 0s 29ms/step
1/1 [=====] - 0s 28ms/step
1/1 [=====] - 0s 34ms/step
1/1 [=====] - 0s 32ms/step
1/1 [=====] - 0s 27ms/step
1/1 [=====] - 0s 33ms/step
1/1 [=====] - 0s 26ms/step
1/1 [=====] - 0s 29ms/step
1/1 [=====] - 0s 32ms/step
1/1 [=====] - 0s 25ms/step
1/1 [=====] - 0s 25ms/step
1/1 [=====] - 0s 26ms/step
1/1 [=====] - 0s 26ms/step
1/1 [=====] - 0s 27ms/step
1/1 [=====] - 0s 29ms/step
1/1 [=====] - 0s 26ms/step
1/1 [=====] - 0s 26ms/step
1/1 [=====] - 0s 37ms/step
1/1 [=====] - 0s 27ms/step
1/1 [=====] - 0s 25ms/step
1/1 [=====] - 0s 26ms/step
1/1 [=====] - 0s 27ms/step
1/1 [=====] - 0s 27ms/step
1/1 [=====] - 0s 27ms/step
1/1 [=====] - 0s 31ms/step
1/1 [=====] - 0s 29ms/step
1/1 [=====] - 0s 28ms/step
1/1 [=====] - 0s 29ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 23ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 23ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 23ms/step
```

```
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 22ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 22ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 22ms/step
1/1 [=====] - 0s 22ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 23ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 22ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 23ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 23ms/step
1/1 [=====] - 0s 19ms/step
```

```
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 23ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 24ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 27ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 23ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 26ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 22ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 23ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 19ms/step
```

```
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 16ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 29ms/step
1/1 [=====] - 0s 31ms/step
1/1 [=====] - 0s 27ms/step
1/1 [=====] - 0s 37ms/step
1/1 [=====] - 0s 26ms/step
1/1 [=====] - 0s 27ms/step
1/1 [=====] - 0s 25ms/step
1/1 [=====] - 0s 26ms/step
1/1 [=====] - 0s 25ms/step
1/1 [=====] - 0s 27ms/step
1/1 [=====] - 0s 29ms/step
1/1 [=====] - 0s 26ms/step
1/1 [=====] - 0s 33ms/step
1/1 [=====] - 0s 26ms/step
1/1 [=====] - 0s 29ms/step
1/1 [=====] - 0s 26ms/step
1/1 [=====] - 0s 26ms/step
```

```
1/1 [=====] - 0s 26ms/step
1/1 [=====] - 0s 28ms/step
1/1 [=====] - 0s 25ms/step
1/1 [=====] - 0s 32ms/step
1/1 [=====] - 0s 25ms/step
1/1 [=====] - 0s 27ms/step
1/1 [=====] - 0s 31ms/step
1/1 [=====] - 0s 25ms/step
1/1 [=====] - 0s 27ms/step
1/1 [=====] - 0s 28ms/step
1/1 [=====] - 0s 26ms/step
1/1 [=====] - 0s 27ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 24ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 23ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 22ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 23ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 27ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 24ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 24ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 24ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 24ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 20ms/step
```

```
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 28ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 23ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 23ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 24ms/step
1/1 [=====] - 0s 22ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 24ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 22ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 16ms/step
```

```
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 23ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 22ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 22ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 24ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 24ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 25ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 22ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 23ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 23ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 25ms/step
1/1 [=====] - 0s 22ms/step
1/1 [=====] - 0s 23ms/step
1/1 [=====] - 0s 22ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 22ms/step
```

```
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 23ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 26ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 31ms/step
1/1 [=====] - 0s 31ms/step
1/1 [=====] - 0s 27ms/step
1/1 [=====] - 0s 28ms/step
1/1 [=====] - 0s 32ms/step
1/1 [=====] - 0s 30ms/step
1/1 [=====] - 0s 29ms/step
1/1 [=====] - 0s 27ms/step
1/1 [=====] - 0s 36ms/step
1/1 [=====] - 0s 27ms/step
1/1 [=====] - 0s 26ms/step
1/1 [=====] - 0s 25ms/step
1/1 [=====] - 0s 27ms/step
1/1 [=====] - 0s 26ms/step
1/1 [=====] - 0s 30ms/step
1/1 [=====] - 0s 32ms/step
1/1 [=====] - 0s 24ms/step
1/1 [=====] - 0s 34ms/step
1/1 [=====] - 0s 26ms/step
1/1 [=====] - 0s 36ms/step
1/1 [=====] - 0s 26ms/step
1/1 [=====] - 0s 26ms/step
1/1 [=====] - 0s 26ms/step
1/1 [=====] - 0s 35ms/step
1/1 [=====] - 0s 27ms/step
1/1 [=====] - 0s 26ms/step
1/1 [=====] - 0s 43ms/step
1/1 [=====] - 0s 28ms/step
1/1 [=====] - 0s 27ms/step
1/1 [=====] - 0s 28ms/step
1/1 [=====] - 0s 22ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 30ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 19ms/step
```



```
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 26ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 24ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 24ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 22ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 23ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 28ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 28ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 21ms/step
```

```
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 25ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 25ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 24ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 22ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 20ms/step
```

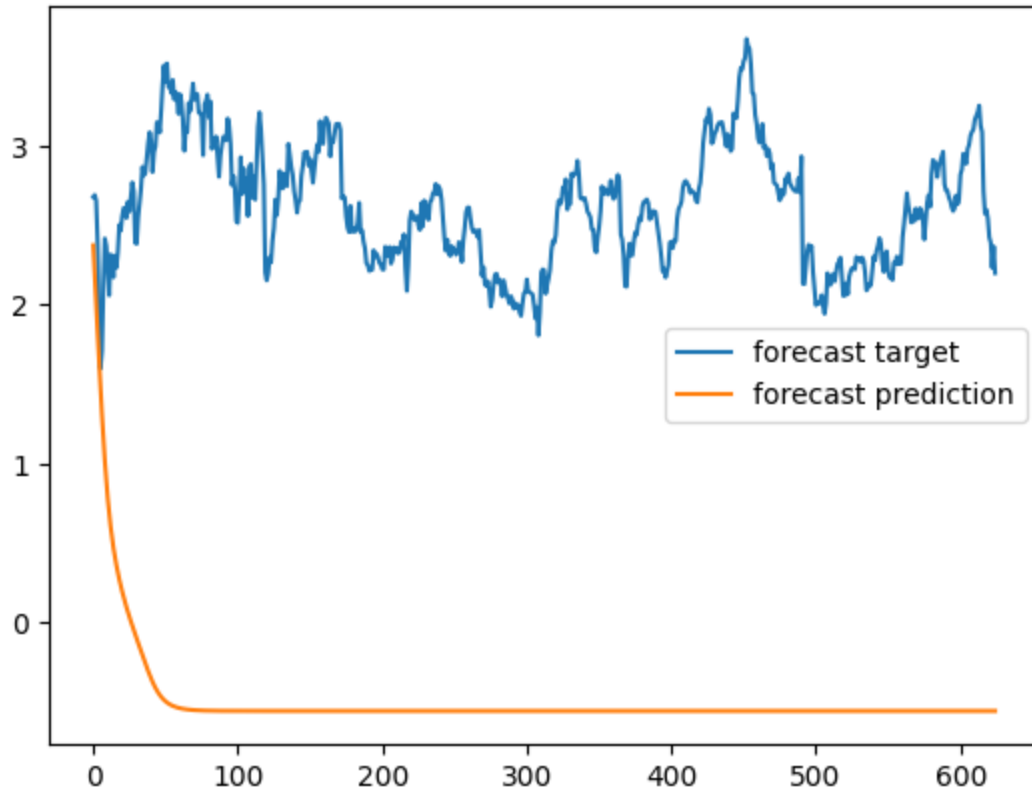
```
1/1 [=====] - 0s 26ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 22ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 22ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 23ms/step
1/1 [=====] - 0s 24ms/step
1/1 [=====] - 0s 22ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 22ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 23ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 22ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 22ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 30ms/step
1/1 [=====] - 0s 34ms/step
1/1 [=====] - 0s 40ms/step
1/1 [=====] - 0s 30ms/step
1/1 [=====] - 0s 26ms/step
1/1 [=====] - 0s 26ms/step
1/1 [=====] - 0s 38ms/step
```

```

1/1 [=====] - 0s 26ms/step
1/1 [=====] - 0s 26ms/step
1/1 [=====] - 0s 28ms/step
1/1 [=====] - 0s 25ms/step
1/1 [=====] - 0s 31ms/step
1/1 [=====] - 0s 26ms/step
1/1 [=====] - 0s 26ms/step
1/1 [=====] - 0s 26ms/step
1/1 [=====] - 0s 26ms/step

```

Out[12]: <matplotlib.legend.Legend at 0x7f5b19ca0a00>



```

In [13]: # calculate returns by first shifting the data
df['PrevClose'] = df['close'].shift(1) # move everything up 1

# so now it's like
# close / prev close
# x[2] x[1]
# x[3] x[2]
# x[4] x[3]
# ...
# x[t] x[t-1]

```

```

In [14]: df.head()

```

Out[14]:

	date	open	high	low	close	volume	Name	PrevClose
0	2013-02-08	27.920	28.325	27.920	28.185	7146296	SBUX	NaN
1	2013-02-11	28.260	28.260	27.930	28.070	5457354	SBUX	28.185
2	2013-02-12	28.000	28.275	27.975	28.130	8665592	SBUX	28.070
3	2013-02-13	28.230	28.230	27.750	27.915	7022056	SBUX	28.130
4	2013-02-14	27.765	27.905	27.675	27.775	8899188	SBUX	27.915

In [15]: *# then the return is*
(x[t] - x[t-1]) / x[t-1]
`df['Return'] = (df['close'] - df['PrevClose']) / df['PrevClose']`

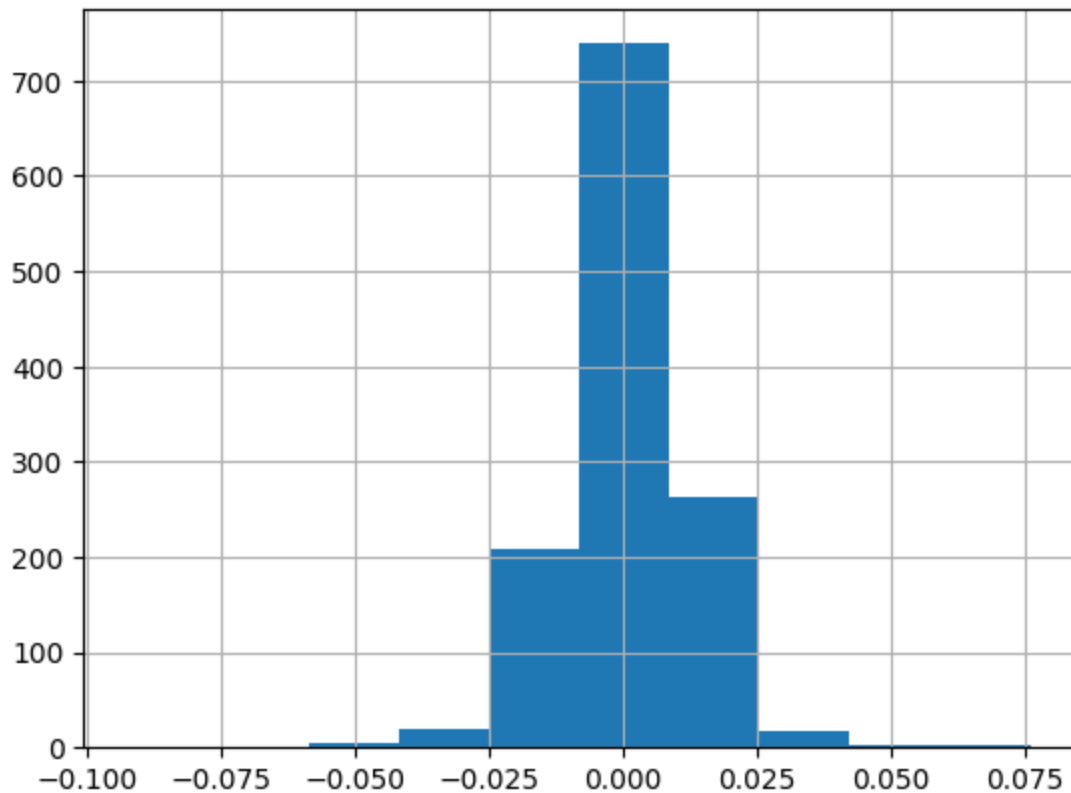
In [16]: `df.head()`

Out[16]:

	date	open	high	low	close	volume	Name	PrevClose	Return
0	2013-02-08	27.920	28.325	27.920	28.185	7146296	SBUX	NaN	NaN
1	2013-02-11	28.260	28.260	27.930	28.070	5457354	SBUX	28.185	-0.004080
2	2013-02-12	28.000	28.275	27.975	28.130	8665592	SBUX	28.070	0.002138
3	2013-02-13	28.230	28.230	27.750	27.915	7022056	SBUX	28.130	-0.007643
4	2013-02-14	27.765	27.905	27.675	27.775	8899188	SBUX	27.915	-0.005015

In [17]: *# Now Let's try an LSTM to predict returns*
`df['Return'].hist()`

Out[17]: <Axes: >



```
In [18]: series = df['Return'].values[1:].reshape(-1, 1)

# Normalize the data
# Note: I didn't think about where the true boundary is, this is just approx.
scaler = StandardScaler()
scaler.fit(series[:len(series) // 2])
series = scaler.transform(series).flatten()
```

```
In [19]: ### build the dataset
# Let's see if we can use T past values to predict the next value
T = 10
D = 1
X = []
Y = []
for t in range(len(series) - T):
    x = series[t:t+T]
    X.append(x)
    y = series[t+T]
    Y.append(y)

X = np.array(X).reshape(-1, T, 1) # Now the data should be N x T x D
Y = np.array(Y)
N = len(X)
print("X.shape", X.shape, "Y.shape", Y.shape)

X.shape (1248, 10, 1) Y.shape (1248,)
```

```
In [20]: ### try autoregressive RNN model
i = Input(shape=(T, 1))
x = LSTM(5)(i)
x = Dense(1)(x)
```

```
model = Model(i, x)
model.compile(
    loss='mse',
    optimizer=Adam(learning_rate=0.01),
)

# train the RNN
r = model.fit(
    X[:-N//2], Y[:-N//2],
    epochs=80,
    validation_data=(X[-N//2:], Y[-N//2:]),
)
```

```
Epoch 1/80
20/20 [=====] - 3s 33ms/step - loss: 0.9964 - val_loss: 1.1614
Epoch 2/80
20/20 [=====] - 0s 8ms/step - loss: 0.9856 - val_loss: 1.1569
Epoch 3/80
20/20 [=====] - 0s 9ms/step - loss: 0.9849 - val_loss: 1.1493
Epoch 4/80
20/20 [=====] - 0s 8ms/step - loss: 0.9827 - val_loss: 1.1573
Epoch 5/80
20/20 [=====] - 0s 7ms/step - loss: 0.9814 - val_loss: 1.1511
Epoch 6/80
20/20 [=====] - 0s 7ms/step - loss: 0.9823 - val_loss: 1.1619
Epoch 7/80
20/20 [=====] - 0s 9ms/step - loss: 0.9795 - val_loss: 1.1544
Epoch 8/80
20/20 [=====] - 0s 8ms/step - loss: 0.9792 - val_loss: 1.1531
Epoch 9/80
20/20 [=====] - 0s 9ms/step - loss: 0.9767 - val_loss: 1.1566
Epoch 10/80
20/20 [=====] - 0s 9ms/step - loss: 0.9806 - val_loss: 1.1729
Epoch 11/80
20/20 [=====] - 0s 8ms/step - loss: 0.9705 - val_loss: 1.1611
Epoch 12/80
20/20 [=====] - 0s 7ms/step - loss: 0.9669 - val_loss: 1.1903
Epoch 13/80
20/20 [=====] - 0s 7ms/step - loss: 0.9570 - val_loss: 1.1995
Epoch 14/80
20/20 [=====] - 0s 7ms/step - loss: 0.9484 - val_loss: 1.2564
Epoch 15/80
20/20 [=====] - 0s 9ms/step - loss: 0.9634 - val_loss: 1.2144
Epoch 16/80
20/20 [=====] - 0s 7ms/step - loss: 0.9477 - val_loss: 1.2303
Epoch 17/80
20/20 [=====] - 0s 8ms/step - loss: 0.9390 - val_loss: 1.2477
Epoch 18/80
20/20 [=====] - 0s 7ms/step - loss: 0.9316 - val_loss: 1.2671
Epoch 19/80
20/20 [=====] - 0s 7ms/step - loss: 0.9339 - val_loss: 1.
```



```
2667
Epoch 20/80
20/20 [=====] - 0s 7ms/step - loss: 0.9486 - val_loss: 1.
2279
Epoch 21/80
20/20 [=====] - 0s 7ms/step - loss: 0.9269 - val_loss: 1.
2790
Epoch 22/80
20/20 [=====] - 0s 8ms/step - loss: 0.9231 - val_loss: 1.
2889
Epoch 23/80
20/20 [=====] - 0s 7ms/step - loss: 0.9176 - val_loss: 1.
2996
Epoch 24/80
20/20 [=====] - 0s 7ms/step - loss: 0.9179 - val_loss: 1.
3022
Epoch 25/80
20/20 [=====] - 0s 7ms/step - loss: 0.9178 - val_loss: 1.
3075
Epoch 26/80
20/20 [=====] - 0s 9ms/step - loss: 0.9096 - val_loss: 1.
3244
Epoch 27/80
20/20 [=====] - 0s 8ms/step - loss: 0.9176 - val_loss: 1.
3132
Epoch 28/80
20/20 [=====] - 0s 9ms/step - loss: 0.9083 - val_loss: 1.
3416
Epoch 29/80
20/20 [=====] - 0s 7ms/step - loss: 0.8996 - val_loss: 1.
3599
Epoch 30/80
20/20 [=====] - 0s 8ms/step - loss: 0.8980 - val_loss: 1.
3391
Epoch 31/80
20/20 [=====] - 0s 8ms/step - loss: 0.8945 - val_loss: 1.
3594
Epoch 32/80
20/20 [=====] - 0s 8ms/step - loss: 0.8999 - val_loss: 1.
3621
Epoch 33/80
20/20 [=====] - 0s 7ms/step - loss: 0.8935 - val_loss: 1.
3864
Epoch 34/80
20/20 [=====] - 0s 8ms/step - loss: 0.9062 - val_loss: 1.
3337
Epoch 35/80
20/20 [=====] - 0s 9ms/step - loss: 0.8965 - val_loss: 1.
3699
Epoch 36/80
20/20 [=====] - 0s 8ms/step - loss: 0.8961 - val_loss: 1.
3568
Epoch 37/80
20/20 [=====] - 0s 8ms/step - loss: 0.8844 - val_loss: 1.
3793
Epoch 38/80
```

```
20/20 [=====] - 0s 9ms/step - loss: 0.8804 - val_loss: 1.3982
Epoch 39/80
20/20 [=====] - 0s 9ms/step - loss: 0.8878 - val_loss: 1.4160
Epoch 40/80
20/20 [=====] - 0s 9ms/step - loss: 0.8802 - val_loss: 1.4053
Epoch 41/80
20/20 [=====] - 0s 9ms/step - loss: 0.8768 - val_loss: 1.3845
Epoch 42/80
20/20 [=====] - 0s 7ms/step - loss: 0.8672 - val_loss: 1.4420
Epoch 43/80
20/20 [=====] - 0s 7ms/step - loss: 0.8611 - val_loss: 1.3964
Epoch 44/80
20/20 [=====] - 0s 8ms/step - loss: 0.8654 - val_loss: 1.4124
Epoch 45/80
20/20 [=====] - 0s 8ms/step - loss: 0.8591 - val_loss: 1.4253
Epoch 46/80
20/20 [=====] - 0s 8ms/step - loss: 0.8510 - val_loss: 1.4755
Epoch 47/80
20/20 [=====] - 0s 9ms/step - loss: 0.8546 - val_loss: 1.4703
Epoch 48/80
20/20 [=====] - 0s 10ms/step - loss: 0.8523 - val_loss: 1.4232
Epoch 49/80
20/20 [=====] - 0s 10ms/step - loss: 0.8329 - val_loss: 1.5295
Epoch 50/80
20/20 [=====] - 0s 10ms/step - loss: 0.8546 - val_loss: 1.4382
Epoch 51/80
20/20 [=====] - 0s 10ms/step - loss: 0.8332 - val_loss: 1.4456
Epoch 52/80
20/20 [=====] - 0s 11ms/step - loss: 0.8360 - val_loss: 1.4854
Epoch 53/80
20/20 [=====] - 0s 11ms/step - loss: 0.8280 - val_loss: 1.4590
Epoch 54/80
20/20 [=====] - 0s 10ms/step - loss: 0.8222 - val_loss: 1.5020
Epoch 55/80
20/20 [=====] - 0s 11ms/step - loss: 0.8140 - val_loss: 1.4739
Epoch 56/80
20/20 [=====] - 0s 15ms/step - loss: 0.8224 - val_loss: 1.4798
```

```
Epoch 57/80
20/20 [=====] - 0s 15ms/step - loss: 0.8030 - val_loss: 1.5697
Epoch 58/80
20/20 [=====] - 0s 7ms/step - loss: 0.8031 - val_loss: 1.4842
Epoch 59/80
20/20 [=====] - 0s 8ms/step - loss: 0.8187 - val_loss: 1.5244
Epoch 60/80
20/20 [=====] - 0s 7ms/step - loss: 0.8154 - val_loss: 1.5157
Epoch 61/80
20/20 [=====] - 0s 7ms/step - loss: 0.7871 - val_loss: 1.5203
Epoch 62/80
20/20 [=====] - 0s 8ms/step - loss: 0.7958 - val_loss: 1.5468
Epoch 63/80
20/20 [=====] - 0s 8ms/step - loss: 0.7903 - val_loss: 1.5494
Epoch 64/80
20/20 [=====] - 0s 7ms/step - loss: 0.7817 - val_loss: 1.5709
Epoch 65/80
20/20 [=====] - 0s 8ms/step - loss: 0.7857 - val_loss: 1.5003
Epoch 66/80
20/20 [=====] - 0s 8ms/step - loss: 0.7891 - val_loss: 1.5321
Epoch 67/80
20/20 [=====] - 0s 8ms/step - loss: 0.8017 - val_loss: 1.5326
Epoch 68/80
20/20 [=====] - 0s 9ms/step - loss: 0.7710 - val_loss: 1.5165
Epoch 69/80
20/20 [=====] - 0s 8ms/step - loss: 0.7680 - val_loss: 1.5938
Epoch 70/80
20/20 [=====] - 0s 7ms/step - loss: 0.7589 - val_loss: 1.5839
Epoch 71/80
20/20 [=====] - 0s 9ms/step - loss: 0.7528 - val_loss: 1.5732
Epoch 72/80
20/20 [=====] - 0s 8ms/step - loss: 0.7447 - val_loss: 1.5561
Epoch 73/80
20/20 [=====] - 0s 7ms/step - loss: 0.7593 - val_loss: 1.5635
Epoch 74/80
20/20 [=====] - 0s 9ms/step - loss: 0.7356 - val_loss: 1.6066
Epoch 75/80
20/20 [=====] - 0s 8ms/step - loss: 0.7456 - val_loss: 1.
```

```

5616
Epoch 76/80
20/20 [=====] - 0s 9ms/step - loss: 0.7246 - val_loss: 1.
5838
Epoch 77/80
20/20 [=====] - 0s 7ms/step - loss: 0.7325 - val_loss: 1.
5861
Epoch 78/80
20/20 [=====] - 0s 8ms/step - loss: 0.7141 - val_loss: 1.
6468
Epoch 79/80
20/20 [=====] - 0s 8ms/step - loss: 0.7365 - val_loss: 1.
6255
Epoch 80/80
20/20 [=====] - 0s 7ms/step - loss: 0.7357 - val_loss: 1.
6218

```

```

In [21]: # Plot loss per iteration
import matplotlib.pyplot as plt
plt.plot(r.history['loss'], label='loss')
plt.plot(r.history['val_loss'], label='val_loss')
plt.legend()

```

Out[21]: <matplotlib.legend.Legend at 0x7f5b18633f40>



```

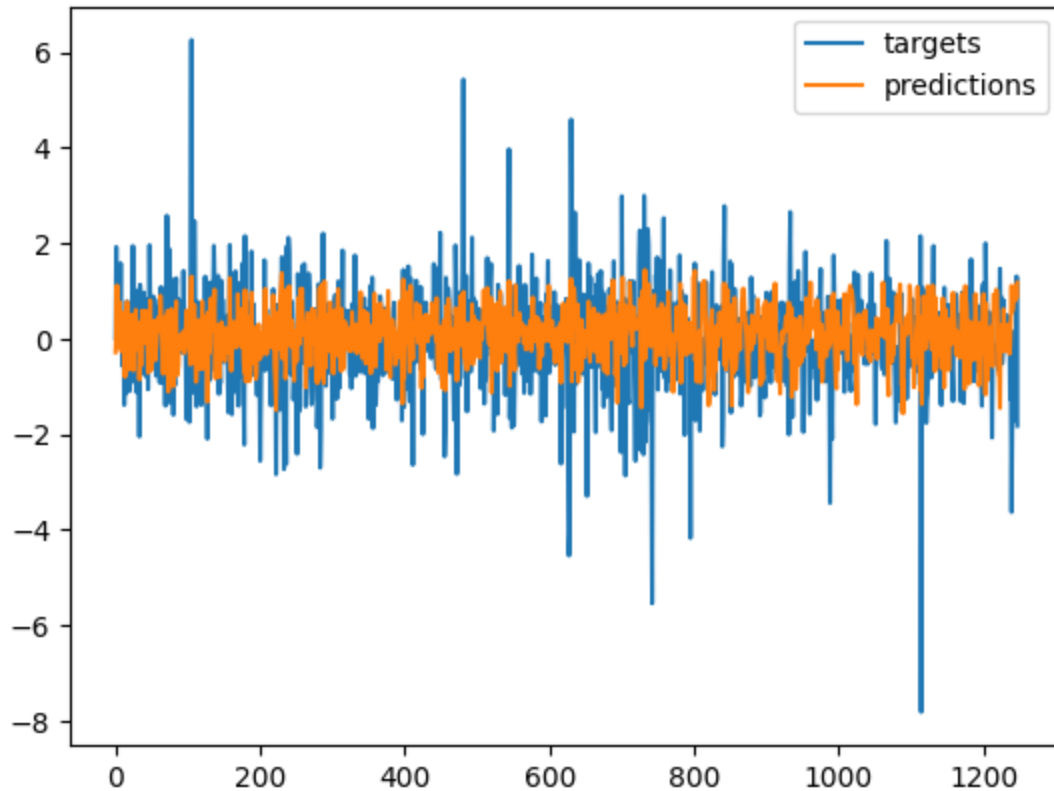
In [22]: # One-step forecast using true targets
outputs = model.predict(X)
print(outputs.shape)
predictions = outputs[:,0]

plt.plot(Y, label='targets')

```

```
plt.plot(predictions, label='predictions')
plt.legend()
plt.show()
```

39/39 [=====] - 0s 2ms/step
(1248, 1)



```
In [23]: # Multi-step forecast
validation_target = Y[-N//2:]
validation_predictions = []

# first validation input
last_x = X[-N//2] # 1-D array of length T

while len(validation_predictions) < len(validation_target):
    p = model.predict(last_x.reshape(1, T, 1))[0,0] # 1x1 array -> scalar

    # update the predictions list
    validation_predictions.append(p)

    # make the new input
    last_x = np.roll(last_x, -1)
    last_x[-1] = p

plt.plot(validation_target, label='forecast target')
plt.plot(validation_predictions, label='forecast prediction')
plt.legend()
```

```
1/1 [=====] - 0s 388ms/step
1/1 [=====] - 0s 22ms/step
1/1 [=====] - 0s 23ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 25ms/step
1/1 [=====] - 0s 24ms/step
1/1 [=====] - 0s 22ms/step
1/1 [=====] - 0s 35ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 22ms/step
1/1 [=====] - 0s 23ms/step
1/1 [=====] - 0s 26ms/step
1/1 [=====] - 0s 22ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 22ms/step
1/1 [=====] - 0s 27ms/step
1/1 [=====] - 0s 26ms/step
1/1 [=====] - 0s 25ms/step
1/1 [=====] - 0s 29ms/step
1/1 [=====] - 0s 22ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 23ms/step
1/1 [=====] - 0s 26ms/step
1/1 [=====] - 0s 22ms/step
1/1 [=====] - 0s 27ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 22ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 23ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 22ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 24ms/step
1/1 [=====] - 0s 25ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 24ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 22ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 22ms/step
1/1 [=====] - 0s 24ms/step
1/1 [=====] - 0s 25ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 22ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 18ms/step
```

```
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 23ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 29ms/step
1/1 [=====] - 0s 32ms/step
1/1 [=====] - 0s 30ms/step
1/1 [=====] - 0s 29ms/step
1/1 [=====] - 0s 35ms/step
1/1 [=====] - 0s 28ms/step
1/1 [=====] - 0s 33ms/step
1/1 [=====] - 0s 26ms/step
1/1 [=====] - 0s 31ms/step
1/1 [=====] - 0s 26ms/step
1/1 [=====] - 0s 36ms/step
1/1 [=====] - 0s 35ms/step
1/1 [=====] - 0s 26ms/step
1/1 [=====] - 0s 27ms/step
1/1 [=====] - 0s 27ms/step
1/1 [=====] - 0s 36ms/step
1/1 [=====] - 0s 27ms/step
1/1 [=====] - 0s 26ms/step
1/1 [=====] - 0s 30ms/step
1/1 [=====] - 0s 26ms/step
1/1 [=====] - 0s 27ms/step
1/1 [=====] - 0s 35ms/step
1/1 [=====] - 0s 28ms/step
1/1 [=====] - 0s 31ms/step
1/1 [=====] - 0s 27ms/step
1/1 [=====] - 0s 27ms/step
1/1 [=====] - 0s 38ms/step
1/1 [=====] - 0s 26ms/step
1/1 [=====] - 0s 29ms/step
1/1 [=====] - 0s 25ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 23ms/step
```

```
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 22ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 23ms/step
1/1 [=====] - 0s 22ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 24ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 24ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 22ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 22ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 22ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 19ms/step
```



```
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 31ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 23ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 25ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 22ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 36ms/step
1/1 [=====] - 0s 58ms/step
1/1 [=====] - 0s 75ms/step
1/1 [=====] - 0s 51ms/step
1/1 [=====] - 0s 23ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 66ms/step
1/1 [=====] - 0s 80ms/step
1/1 [=====] - 0s 59ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 22ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 30ms/step
1/1 [=====] - 0s 89ms/step
1/1 [=====] - 0s 34ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 22ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 33ms/step
```

```
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 24ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 22ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 22ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 25ms/step
1/1 [=====] - 0s 22ms/step
1/1 [=====] - 0s 24ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 24ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 23ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 29ms/step
1/1 [=====] - 0s 31ms/step
1/1 [=====] - 0s 37ms/step
1/1 [=====] - 0s 25ms/step
1/1 [=====] - 0s 28ms/step
1/1 [=====] - 0s 32ms/step
1/1 [=====] - 0s 26ms/step
1/1 [=====] - 0s 25ms/step
1/1 [=====] - 0s 33ms/step
1/1 [=====] - 0s 31ms/step
1/1 [=====] - 0s 25ms/step
1/1 [=====] - 0s 26ms/step
1/1 [=====] - 0s 26ms/step
1/1 [=====] - 0s 29ms/step
1/1 [=====] - 0s 27ms/step
1/1 [=====] - 0s 26ms/step
1/1 [=====] - 0s 35ms/step
```

```
1/1 [=====] - 0s 29ms/step
1/1 [=====] - 0s 25ms/step
1/1 [=====] - 0s 25ms/step
1/1 [=====] - 0s 25ms/step
1/1 [=====] - 0s 25ms/step
1/1 [=====] - 0s 39ms/step
1/1 [=====] - 0s 38ms/step
1/1 [=====] - 0s 27ms/step
1/1 [=====] - 0s 28ms/step
1/1 [=====] - 0s 28ms/step
1/1 [=====] - 0s 26ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 23ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 22ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 24ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 16ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 26ms/step
```

```
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 22ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 23ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 22ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 25ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 22ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 25ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 22ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 24ms/step
```

```
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 25ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 25ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 22ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 22ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 23ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 22ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 22ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 22ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 23ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 20ms/step
```

```
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 28ms/step
1/1 [=====] - 0s 25ms/step
1/1 [=====] - 0s 28ms/step
1/1 [=====] - 0s 30ms/step
1/1 [=====] - 0s 26ms/step
1/1 [=====] - 0s 27ms/step
1/1 [=====] - 0s 28ms/step
1/1 [=====] - 0s 32ms/step
1/1 [=====] - 0s 29ms/step
1/1 [=====] - 0s 27ms/step
1/1 [=====] - 0s 27ms/step
1/1 [=====] - 0s 25ms/step
1/1 [=====] - 0s 25ms/step
1/1 [=====] - 0s 36ms/step
1/1 [=====] - 0s 27ms/step
1/1 [=====] - 0s 27ms/step
1/1 [=====] - 0s 32ms/step
1/1 [=====] - 0s 34ms/step
1/1 [=====] - 0s 27ms/step
1/1 [=====] - 0s 28ms/step
1/1 [=====] - 0s 26ms/step
1/1 [=====] - 0s 29ms/step
1/1 [=====] - 0s 29ms/step
1/1 [=====] - 0s 35ms/step
1/1 [=====] - 0s 28ms/step
1/1 [=====] - 0s 29ms/step
1/1 [=====] - 0s 30ms/step
1/1 [=====] - 0s 26ms/step
1/1 [=====] - 0s 27ms/step
1/1 [=====] - 0s 26ms/step
1/1 [=====] - 0s 26ms/step
1/1 [=====] - 0s 28ms/step
1/1 [=====] - 0s 28ms/step
1/1 [=====] - 0s 29ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 17ms/step
```

```
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 22ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 27ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 27ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 22ms/step
1/1 [=====] - 0s 19ms/step
```

```
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 22ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 23ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 24ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 22ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 24ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 17ms/step
1/1 [=====] - 0s 26ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 23ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 22ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 27ms/step
1/1 [=====] - 0s 24ms/step
1/1 [=====] - 0s 24ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 18ms/step
```

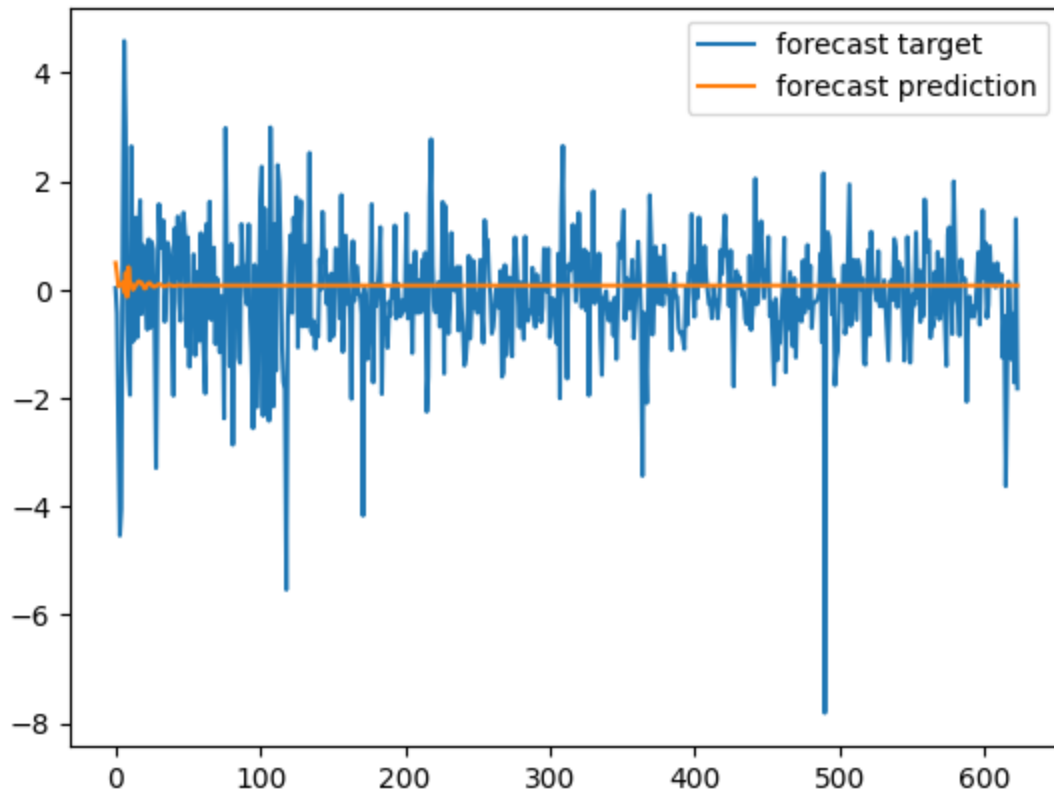


```

1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 18ms/step
1/1 [=====] - 0s 20ms/step
1/1 [=====] - 0s 18ms/step

```

Out[23]: <matplotlib.legend.Legend at 0x7f5b201bfd30>



In [24]: *# Now turn the full data into numpy arrays*

```

# Not yet in the final "X" format!
input_data = df[['open', 'high', 'low', 'close', 'volume']].values
targets = df['Return'].values

```

In [25]: *# Now make the actual data which will go into the neural network*

```

T = 10 # the number of time steps to look at to make a prediction for the next day
D = input_data.shape[1]
N = len(input_data) - T # (e.g. if T=10 and you have 11 data points then you'd only

```

In [26]: *# normalize the inputs*

```

Ntrain = len(input_data) * 2 // 3
scaler = StandardScaler()
scaler.fit(input_data[:Ntrain + T - 1])
input_data = scaler.transform(input_data)

```

In [27]: *# Setup X_train and Y_train*

```

X_train = np.zeros((Ntrain, T, D))
Y_train = np.zeros(Ntrain)

```

```
for t in range(Ntrain):
    X_train[t, :, :] = input_data[t:t+T]
    Y_train[t] = (targets[t+T] > 0)
```

```
In [28]: # Setup X_test and Y_test
X_test = np.zeros((N - Ntrain, T, D))
Y_test = np.zeros(N - Ntrain)

for u in range(N - Ntrain):
    # u counts from 0...(N - Ntrain)
    # t counts from Ntrain...N
    t = u + Ntrain
    X_test[u, :, :] = input_data[t:t+T]
    Y_test[u] = (targets[t+T] > 0)
```

```
In [29]: # make the RNN
i = Input(shape=(T, D))
x = LSTM(50)(i)
x = Dense(1, activation='sigmoid')(x)
model = Model(i, x)
model.compile(
    loss='binary_crossentropy',
    optimizer=Adam(learning_rate=0.001),
    metrics=['accuracy'],
)
```

```
In [30]: # train the RNN
r = model.fit(
    X_train, Y_train,
    batch_size=32,
    epochs=300,
    validation_data=(X_test, Y_test),
)
```

Epoch 1/300
27/27 [=====] - 3s 34ms/step - loss: 0.6964 - accuracy: 0.4887 - val_loss: 0.6949 - val_accuracy: 0.4805

Epoch 2/300
27/27 [=====] - 0s 9ms/step - loss: 0.6936 - accuracy: 0.5280 - val_loss: 0.6978 - val_accuracy: 0.4854

Epoch 3/300
27/27 [=====] - 0s 10ms/step - loss: 0.6920 - accuracy: 0.5256 - val_loss: 0.6940 - val_accuracy: 0.5171

Epoch 4/300
27/27 [=====] - 0s 10ms/step - loss: 0.6907 - accuracy: 0.5352 - val_loss: 0.6959 - val_accuracy: 0.4829

Epoch 5/300
27/27 [=====] - 0s 8ms/step - loss: 0.6910 - accuracy: 0.5197 - val_loss: 0.6943 - val_accuracy: 0.5049

Epoch 6/300
27/27 [=====] - 0s 7ms/step - loss: 0.6928 - accuracy: 0.5352 - val_loss: 0.6981 - val_accuracy: 0.4927

Epoch 7/300
27/27 [=====] - 0s 8ms/step - loss: 0.6902 - accuracy: 0.5364 - val_loss: 0.6971 - val_accuracy: 0.4902

Epoch 8/300
27/27 [=====] - 0s 7ms/step - loss: 0.6882 - accuracy: 0.5423 - val_loss: 0.6925 - val_accuracy: 0.5098

Epoch 9/300
27/27 [=====] - 0s 8ms/step - loss: 0.6886 - accuracy: 0.5423 - val_loss: 0.6922 - val_accuracy: 0.5122

Epoch 10/300
27/27 [=====] - 0s 8ms/step - loss: 0.6885 - accuracy: 0.5495 - val_loss: 0.6905 - val_accuracy: 0.5390

Epoch 11/300
27/27 [=====] - 0s 8ms/step - loss: 0.6880 - accuracy: 0.5352 - val_loss: 0.6920 - val_accuracy: 0.5073

Epoch 12/300
27/27 [=====] - 0s 7ms/step - loss: 0.6873 - accuracy: 0.5328 - val_loss: 0.7006 - val_accuracy: 0.4951

Epoch 13/300
27/27 [=====] - 0s 7ms/step - loss: 0.6882 - accuracy: 0.5411 - val_loss: 0.6909 - val_accuracy: 0.5390

Epoch 14/300
27/27 [=====] - 0s 7ms/step - loss: 0.6895 - accuracy: 0.5244 - val_loss: 0.6933 - val_accuracy: 0.5000

Epoch 15/300
27/27 [=====] - 0s 7ms/step - loss: 0.6885 - accuracy: 0.5304 - val_loss: 0.6980 - val_accuracy: 0.4878

Epoch 16/300
27/27 [=====] - 0s 8ms/step - loss: 0.6883 - accuracy: 0.5364 - val_loss: 0.6932 - val_accuracy: 0.5122

Epoch 17/300
27/27 [=====] - 0s 7ms/step - loss: 0.6876 - accuracy: 0.5375 - val_loss: 0.6970 - val_accuracy: 0.4902

Epoch 18/300
27/27 [=====] - 0s 6ms/step - loss: 0.6858 - accuracy: 0.5459 - val_loss: 0.6968 - val_accuracy: 0.4878

Epoch 19/300
27/27 [=====] - 0s 8ms/step - loss: 0.6853 - accuracy: 0.

```
5459 - val_loss: 0.6963 - val_accuracy: 0.4829
Epoch 20/300
27/27 [=====] - 0s 7ms/step - loss: 0.6855 - accuracy: 0.
5578 - val_loss: 0.6924 - val_accuracy: 0.5268
Epoch 21/300
27/27 [=====] - 0s 7ms/step - loss: 0.6848 - accuracy: 0.
5542 - val_loss: 0.6996 - val_accuracy: 0.4756
Epoch 22/300
27/27 [=====] - 0s 6ms/step - loss: 0.6866 - accuracy: 0.
5399 - val_loss: 0.6943 - val_accuracy: 0.5146
Epoch 23/300
27/27 [=====] - 0s 6ms/step - loss: 0.6839 - accuracy: 0.
5626 - val_loss: 0.6954 - val_accuracy: 0.5122
Epoch 24/300
27/27 [=====] - 0s 8ms/step - loss: 0.6839 - accuracy: 0.
5578 - val_loss: 0.6937 - val_accuracy: 0.5073
Epoch 25/300
27/27 [=====] - 0s 7ms/step - loss: 0.6827 - accuracy: 0.
5721 - val_loss: 0.6941 - val_accuracy: 0.5220
Epoch 26/300
27/27 [=====] - 0s 6ms/step - loss: 0.6845 - accuracy: 0.
5638 - val_loss: 0.6942 - val_accuracy: 0.5171
Epoch 27/300
27/27 [=====] - 0s 6ms/step - loss: 0.6830 - accuracy: 0.
5423 - val_loss: 0.6983 - val_accuracy: 0.4878
Epoch 28/300
27/27 [=====] - 0s 6ms/step - loss: 0.6833 - accuracy: 0.
5638 - val_loss: 0.6958 - val_accuracy: 0.5220
Epoch 29/300
27/27 [=====] - 0s 6ms/step - loss: 0.6818 - accuracy: 0.
5888 - val_loss: 0.6980 - val_accuracy: 0.4902
Epoch 30/300
27/27 [=====] - 0s 7ms/step - loss: 0.6825 - accuracy: 0.
5745 - val_loss: 0.6976 - val_accuracy: 0.4878
Epoch 31/300
27/27 [=====] - 0s 8ms/step - loss: 0.6828 - accuracy: 0.
5495 - val_loss: 0.6940 - val_accuracy: 0.5293
Epoch 32/300
27/27 [=====] - 0s 8ms/step - loss: 0.6818 - accuracy: 0.
5781 - val_loss: 0.6966 - val_accuracy: 0.5220
Epoch 33/300
27/27 [=====] - 0s 6ms/step - loss: 0.6816 - accuracy: 0.
5769 - val_loss: 0.6956 - val_accuracy: 0.5171
Epoch 34/300
27/27 [=====] - 0s 8ms/step - loss: 0.6806 - accuracy: 0.
5662 - val_loss: 0.6997 - val_accuracy: 0.4780
Epoch 35/300
27/27 [=====] - 0s 7ms/step - loss: 0.6804 - accuracy: 0.
5709 - val_loss: 0.6980 - val_accuracy: 0.5122
Epoch 36/300
27/27 [=====] - 0s 7ms/step - loss: 0.6788 - accuracy: 0.
5626 - val_loss: 0.6964 - val_accuracy: 0.5220
Epoch 37/300
27/27 [=====] - 0s 8ms/step - loss: 0.6789 - accuracy: 0.
5709 - val_loss: 0.6990 - val_accuracy: 0.5171
Epoch 38/300
```

```
27/27 [=====] - 0s 6ms/step - loss: 0.6782 - accuracy: 0.5816 - val_loss: 0.6992 - val_accuracy: 0.4927
Epoch 39/300
27/27 [=====] - 0s 6ms/step - loss: 0.6802 - accuracy: 0.5852 - val_loss: 0.6967 - val_accuracy: 0.5122
Epoch 40/300
27/27 [=====] - 0s 8ms/step - loss: 0.6764 - accuracy: 0.5662 - val_loss: 0.6957 - val_accuracy: 0.5171
Epoch 41/300
27/27 [=====] - 0s 8ms/step - loss: 0.6779 - accuracy: 0.5530 - val_loss: 0.6953 - val_accuracy: 0.5293
Epoch 42/300
27/27 [=====] - 0s 6ms/step - loss: 0.6765 - accuracy: 0.5614 - val_loss: 0.6979 - val_accuracy: 0.5171
Epoch 43/300
27/27 [=====] - 0s 7ms/step - loss: 0.6763 - accuracy: 0.5650 - val_loss: 0.6962 - val_accuracy: 0.5195
Epoch 44/300
27/27 [=====] - 0s 7ms/step - loss: 0.6748 - accuracy: 0.5721 - val_loss: 0.7018 - val_accuracy: 0.4854
Epoch 45/300
27/27 [=====] - 0s 7ms/step - loss: 0.6770 - accuracy: 0.5518 - val_loss: 0.7037 - val_accuracy: 0.4902
Epoch 46/300
27/27 [=====] - 0s 7ms/step - loss: 0.6743 - accuracy: 0.5614 - val_loss: 0.6976 - val_accuracy: 0.5171
Epoch 47/300
27/27 [=====] - 0s 7ms/step - loss: 0.6735 - accuracy: 0.5673 - val_loss: 0.7009 - val_accuracy: 0.5024
Epoch 48/300
27/27 [=====] - 0s 6ms/step - loss: 0.6734 - accuracy: 0.5685 - val_loss: 0.6971 - val_accuracy: 0.5171
Epoch 49/300
27/27 [=====] - 0s 6ms/step - loss: 0.6735 - accuracy: 0.5733 - val_loss: 0.6975 - val_accuracy: 0.5146
Epoch 50/300
27/27 [=====] - 0s 8ms/step - loss: 0.6741 - accuracy: 0.5685 - val_loss: 0.6985 - val_accuracy: 0.5146
Epoch 51/300
27/27 [=====] - 0s 7ms/step - loss: 0.6724 - accuracy: 0.5697 - val_loss: 0.7027 - val_accuracy: 0.4902
Epoch 52/300
27/27 [=====] - 0s 8ms/step - loss: 0.6698 - accuracy: 0.5781 - val_loss: 0.6984 - val_accuracy: 0.5195
Epoch 53/300
27/27 [=====] - 0s 6ms/step - loss: 0.6694 - accuracy: 0.5662 - val_loss: 0.7020 - val_accuracy: 0.5049
Epoch 54/300
27/27 [=====] - 0s 8ms/step - loss: 0.6709 - accuracy: 0.5650 - val_loss: 0.7087 - val_accuracy: 0.4805
Epoch 55/300
27/27 [=====] - 0s 7ms/step - loss: 0.6707 - accuracy: 0.5650 - val_loss: 0.6973 - val_accuracy: 0.5122
Epoch 56/300
27/27 [=====] - 0s 9ms/step - loss: 0.6678 - accuracy: 0.5816 - val_loss: 0.7008 - val_accuracy: 0.5024
```

Epoch 57/300
27/27 [=====] - 0s 10ms/step - loss: 0.6664 - accuracy: 0.5757 - val_loss: 0.7022 - val_accuracy: 0.4951

Epoch 58/300
27/27 [=====] - 0s 10ms/step - loss: 0.6668 - accuracy: 0.5709 - val_loss: 0.7018 - val_accuracy: 0.5024

Epoch 59/300
27/27 [=====] - 0s 10ms/step - loss: 0.6675 - accuracy: 0.5936 - val_loss: 0.7023 - val_accuracy: 0.5171

Epoch 60/300
27/27 [=====] - 0s 10ms/step - loss: 0.6667 - accuracy: 0.5805 - val_loss: 0.7025 - val_accuracy: 0.4976

Epoch 61/300
27/27 [=====] - 0s 10ms/step - loss: 0.6645 - accuracy: 0.5745 - val_loss: 0.7061 - val_accuracy: 0.4829

Epoch 62/300
27/27 [=====] - 0s 10ms/step - loss: 0.6630 - accuracy: 0.5900 - val_loss: 0.7083 - val_accuracy: 0.4951

Epoch 63/300
27/27 [=====] - 0s 10ms/step - loss: 0.6641 - accuracy: 0.5900 - val_loss: 0.7101 - val_accuracy: 0.4829

Epoch 64/300
27/27 [=====] - 0s 10ms/step - loss: 0.6643 - accuracy: 0.5793 - val_loss: 0.7017 - val_accuracy: 0.4902

Epoch 65/300
27/27 [=====] - 0s 6ms/step - loss: 0.6632 - accuracy: 0.5924 - val_loss: 0.7116 - val_accuracy: 0.4854

Epoch 66/300
27/27 [=====] - 0s 8ms/step - loss: 0.6587 - accuracy: 0.5924 - val_loss: 0.7080 - val_accuracy: 0.5049

Epoch 67/300
27/27 [=====] - 0s 6ms/step - loss: 0.6600 - accuracy: 0.5864 - val_loss: 0.7100 - val_accuracy: 0.4854

Epoch 68/300
27/27 [=====] - 0s 7ms/step - loss: 0.6597 - accuracy: 0.5840 - val_loss: 0.7097 - val_accuracy: 0.5000

Epoch 69/300
27/27 [=====] - 0s 7ms/step - loss: 0.6544 - accuracy: 0.5971 - val_loss: 0.7129 - val_accuracy: 0.4829

Epoch 70/300
27/27 [=====] - 0s 8ms/step - loss: 0.6581 - accuracy: 0.5959 - val_loss: 0.7159 - val_accuracy: 0.4829

Epoch 71/300
27/27 [=====] - 0s 8ms/step - loss: 0.6533 - accuracy: 0.5995 - val_loss: 0.7126 - val_accuracy: 0.4780

Epoch 72/300
27/27 [=====] - 0s 7ms/step - loss: 0.6549 - accuracy: 0.6007 - val_loss: 0.7130 - val_accuracy: 0.4805

Epoch 73/300
27/27 [=====] - 0s 7ms/step - loss: 0.6521 - accuracy: 0.6019 - val_loss: 0.7210 - val_accuracy: 0.4780

Epoch 74/300
27/27 [=====] - 0s 7ms/step - loss: 0.6498 - accuracy: 0.6162 - val_loss: 0.7127 - val_accuracy: 0.4902

Epoch 75/300
27/27 [=====] - 0s 7ms/step - loss: 0.6510 - accuracy: 0.

```
6114 - val_loss: 0.7165 - val_accuracy: 0.4854
Epoch 76/300
27/27 [=====] - 0s 7ms/step - loss: 0.6457 - accuracy: 0.
6055 - val_loss: 0.7137 - val_accuracy: 0.4927
Epoch 77/300
27/27 [=====] - 0s 7ms/step - loss: 0.6440 - accuracy: 0.
6114 - val_loss: 0.7165 - val_accuracy: 0.4927
Epoch 78/300
27/27 [=====] - 0s 7ms/step - loss: 0.6440 - accuracy: 0.
6043 - val_loss: 0.7227 - val_accuracy: 0.4805
Epoch 79/300
27/27 [=====] - 0s 7ms/step - loss: 0.6436 - accuracy: 0.
6114 - val_loss: 0.7137 - val_accuracy: 0.5073
Epoch 80/300
27/27 [=====] - 0s 7ms/step - loss: 0.6436 - accuracy: 0.
6019 - val_loss: 0.7181 - val_accuracy: 0.4878
Epoch 81/300
27/27 [=====] - 0s 7ms/step - loss: 0.6380 - accuracy: 0.
6234 - val_loss: 0.7265 - val_accuracy: 0.4829
Epoch 82/300
27/27 [=====] - 0s 8ms/step - loss: 0.6403 - accuracy: 0.
6150 - val_loss: 0.7236 - val_accuracy: 0.4902
Epoch 83/300
27/27 [=====] - 0s 8ms/step - loss: 0.6395 - accuracy: 0.
6246 - val_loss: 0.7282 - val_accuracy: 0.4829
Epoch 84/300
27/27 [=====] - 0s 6ms/step - loss: 0.6341 - accuracy: 0.
6234 - val_loss: 0.7203 - val_accuracy: 0.4902
Epoch 85/300
27/27 [=====] - 0s 8ms/step - loss: 0.6322 - accuracy: 0.
6114 - val_loss: 0.7294 - val_accuracy: 0.4878
Epoch 86/300
27/27 [=====] - 0s 7ms/step - loss: 0.6300 - accuracy: 0.
6293 - val_loss: 0.7350 - val_accuracy: 0.4805
Epoch 87/300
27/27 [=====] - 0s 7ms/step - loss: 0.6330 - accuracy: 0.
6174 - val_loss: 0.7361 - val_accuracy: 0.4805
Epoch 88/300
27/27 [=====] - 0s 8ms/step - loss: 0.6278 - accuracy: 0.
6389 - val_loss: 0.7217 - val_accuracy: 0.4951
Epoch 89/300
27/27 [=====] - 0s 7ms/step - loss: 0.6256 - accuracy: 0.
6234 - val_loss: 0.7276 - val_accuracy: 0.4805
Epoch 90/300
27/27 [=====] - 0s 7ms/step - loss: 0.6185 - accuracy: 0.
6484 - val_loss: 0.7446 - val_accuracy: 0.4683
Epoch 91/300
27/27 [=====] - 0s 7ms/step - loss: 0.6212 - accuracy: 0.
6281 - val_loss: 0.7529 - val_accuracy: 0.4829
Epoch 92/300
27/27 [=====] - 0s 8ms/step - loss: 0.6260 - accuracy: 0.
6222 - val_loss: 0.7363 - val_accuracy: 0.4854
Epoch 93/300
27/27 [=====] - 0s 8ms/step - loss: 0.6200 - accuracy: 0.
6520 - val_loss: 0.7425 - val_accuracy: 0.4878
Epoch 94/300
```

```
27/27 [=====] - 0s 8ms/step - loss: 0.6157 - accuracy: 0.6269 - val_loss: 0.7462 - val_accuracy: 0.4829
Epoch 95/300
27/27 [=====] - 0s 7ms/step - loss: 0.6094 - accuracy: 0.6448 - val_loss: 0.7497 - val_accuracy: 0.4829
Epoch 96/300
27/27 [=====] - 0s 8ms/step - loss: 0.6103 - accuracy: 0.6389 - val_loss: 0.7517 - val_accuracy: 0.4805
Epoch 97/300
27/27 [=====] - 0s 7ms/step - loss: 0.6076 - accuracy: 0.6544 - val_loss: 0.7483 - val_accuracy: 0.4927
Epoch 98/300
27/27 [=====] - 0s 7ms/step - loss: 0.6072 - accuracy: 0.6496 - val_loss: 0.7675 - val_accuracy: 0.4829
Epoch 99/300
27/27 [=====] - 0s 7ms/step - loss: 0.6052 - accuracy: 0.6472 - val_loss: 0.7493 - val_accuracy: 0.4780
Epoch 100/300
27/27 [=====] - 0s 8ms/step - loss: 0.6032 - accuracy: 0.6496 - val_loss: 0.7554 - val_accuracy: 0.4878
Epoch 101/300
27/27 [=====] - 0s 7ms/step - loss: 0.6047 - accuracy: 0.6544 - val_loss: 0.7601 - val_accuracy: 0.4756
Epoch 102/300
27/27 [=====] - 0s 8ms/step - loss: 0.6017 - accuracy: 0.6555 - val_loss: 0.7420 - val_accuracy: 0.5049
Epoch 103/300
27/27 [=====] - 0s 7ms/step - loss: 0.5978 - accuracy: 0.6532 - val_loss: 0.7529 - val_accuracy: 0.4854
Epoch 104/300
27/27 [=====] - 0s 7ms/step - loss: 0.5926 - accuracy: 0.6651 - val_loss: 0.7690 - val_accuracy: 0.4829
Epoch 105/300
27/27 [=====] - 0s 7ms/step - loss: 0.5941 - accuracy: 0.6615 - val_loss: 0.7564 - val_accuracy: 0.4878
Epoch 106/300
27/27 [=====] - 0s 8ms/step - loss: 0.5931 - accuracy: 0.6555 - val_loss: 0.7652 - val_accuracy: 0.4780
Epoch 107/300
27/27 [=====] - 0s 8ms/step - loss: 0.5930 - accuracy: 0.6532 - val_loss: 0.7556 - val_accuracy: 0.5024
Epoch 108/300
27/27 [=====] - 0s 7ms/step - loss: 0.5845 - accuracy: 0.6651 - val_loss: 0.7554 - val_accuracy: 0.4805
Epoch 109/300
27/27 [=====] - 0s 8ms/step - loss: 0.5882 - accuracy: 0.6639 - val_loss: 0.7617 - val_accuracy: 0.4976
Epoch 110/300
27/27 [=====] - 0s 7ms/step - loss: 0.5862 - accuracy: 0.6865 - val_loss: 0.7830 - val_accuracy: 0.4756
Epoch 111/300
27/27 [=====] - 0s 7ms/step - loss: 0.5788 - accuracy: 0.6758 - val_loss: 0.7664 - val_accuracy: 0.4878
Epoch 112/300
27/27 [=====] - 0s 7ms/step - loss: 0.5791 - accuracy: 0.6746 - val_loss: 0.7761 - val_accuracy: 0.5000
```


Epoch 113/300
27/27 [=====] - 0s 7ms/step - loss: 0.5778 - accuracy: 0.6770 - val_loss: 0.7762 - val_accuracy: 0.4683

Epoch 114/300
27/27 [=====] - 0s 8ms/step - loss: 0.5754 - accuracy: 0.6663 - val_loss: 0.7809 - val_accuracy: 0.4780

Epoch 115/300
27/27 [=====] - 0s 9ms/step - loss: 0.5743 - accuracy: 0.6794 - val_loss: 0.7806 - val_accuracy: 0.4683

Epoch 116/300
27/27 [=====] - 0s 9ms/step - loss: 0.5718 - accuracy: 0.6830 - val_loss: 0.7791 - val_accuracy: 0.4756

Epoch 117/300
27/27 [=====] - 0s 9ms/step - loss: 0.5647 - accuracy: 0.6853 - val_loss: 0.7813 - val_accuracy: 0.4780

Epoch 118/300
27/27 [=====] - 0s 10ms/step - loss: 0.5651 - accuracy: 0.6758 - val_loss: 0.7799 - val_accuracy: 0.4732

Epoch 119/300
27/27 [=====] - 0s 10ms/step - loss: 0.5589 - accuracy: 0.6901 - val_loss: 0.7903 - val_accuracy: 0.4854

Epoch 120/300
27/27 [=====] - 0s 10ms/step - loss: 0.5640 - accuracy: 0.6818 - val_loss: 0.7803 - val_accuracy: 0.4634

Epoch 121/300
27/27 [=====] - 0s 9ms/step - loss: 0.5589 - accuracy: 0.6853 - val_loss: 0.7858 - val_accuracy: 0.4902

Epoch 122/300
27/27 [=====] - 0s 10ms/step - loss: 0.5548 - accuracy: 0.6853 - val_loss: 0.7852 - val_accuracy: 0.4902

Epoch 123/300
27/27 [=====] - 0s 10ms/step - loss: 0.5510 - accuracy: 0.6865 - val_loss: 0.7997 - val_accuracy: 0.4780

Epoch 124/300
27/27 [=====] - 0s 8ms/step - loss: 0.5513 - accuracy: 0.6901 - val_loss: 0.7947 - val_accuracy: 0.4878

Epoch 125/300
27/27 [=====] - 0s 7ms/step - loss: 0.5470 - accuracy: 0.7008 - val_loss: 0.7879 - val_accuracy: 0.5024

Epoch 126/300
27/27 [=====] - 0s 6ms/step - loss: 0.5478 - accuracy: 0.7008 - val_loss: 0.8054 - val_accuracy: 0.4927

Epoch 127/300
27/27 [=====] - 0s 7ms/step - loss: 0.5430 - accuracy: 0.6961 - val_loss: 0.8004 - val_accuracy: 0.4976

Epoch 128/300
27/27 [=====] - 0s 7ms/step - loss: 0.5479 - accuracy: 0.6937 - val_loss: 0.7950 - val_accuracy: 0.4902

Epoch 129/300
27/27 [=====] - 0s 7ms/step - loss: 0.5411 - accuracy: 0.7080 - val_loss: 0.8000 - val_accuracy: 0.4732

Epoch 130/300
27/27 [=====] - 0s 7ms/step - loss: 0.5323 - accuracy: 0.7068 - val_loss: 0.8174 - val_accuracy: 0.4878

Epoch 131/300
27/27 [=====] - 0s 8ms/step - loss: 0.5275 - accuracy: 0.

7151 - val_loss: 0.8101 - val_accuracy: 0.4805
Epoch 132/300
27/27 [=====] - 0s 7ms/step - loss: 0.5278 - accuracy: 0.
7080 - val_loss: 0.8021 - val_accuracy: 0.4951
Epoch 133/300
27/27 [=====] - 0s 8ms/step - loss: 0.5319 - accuracy: 0.
7271 - val_loss: 0.8145 - val_accuracy: 0.4854
Epoch 134/300
27/27 [=====] - 0s 7ms/step - loss: 0.5295 - accuracy: 0.
7163 - val_loss: 0.8242 - val_accuracy: 0.4927
Epoch 135/300
27/27 [=====] - 0s 7ms/step - loss: 0.5364 - accuracy: 0.
7080 - val_loss: 0.8136 - val_accuracy: 0.4902
Epoch 136/300
27/27 [=====] - 0s 8ms/step - loss: 0.5362 - accuracy: 0.
7044 - val_loss: 0.8161 - val_accuracy: 0.4683
Epoch 137/300
27/27 [=====] - 0s 7ms/step - loss: 0.5265 - accuracy: 0.
7044 - val_loss: 0.8539 - val_accuracy: 0.4854
Epoch 138/300
27/27 [=====] - 0s 7ms/step - loss: 0.5141 - accuracy: 0.
7187 - val_loss: 0.8262 - val_accuracy: 0.4976
Epoch 139/300
27/27 [=====] - 0s 7ms/step - loss: 0.5108 - accuracy: 0.
7294 - val_loss: 0.8236 - val_accuracy: 0.4976
Epoch 140/300
27/27 [=====] - 0s 7ms/step - loss: 0.5111 - accuracy: 0.
7223 - val_loss: 0.8541 - val_accuracy: 0.5000
Epoch 141/300
27/27 [=====] - 0s 7ms/step - loss: 0.5124 - accuracy: 0.
7211 - val_loss: 0.8279 - val_accuracy: 0.4732
Epoch 142/300
27/27 [=====] - 0s 8ms/step - loss: 0.5034 - accuracy: 0.
7247 - val_loss: 0.8348 - val_accuracy: 0.4854
Epoch 143/300
27/27 [=====] - 0s 8ms/step - loss: 0.5036 - accuracy: 0.
7318 - val_loss: 0.8398 - val_accuracy: 0.4927
Epoch 144/300
27/27 [=====] - 0s 7ms/step - loss: 0.4967 - accuracy: 0.
7414 - val_loss: 0.8533 - val_accuracy: 0.4951
Epoch 145/300
27/27 [=====] - 0s 8ms/step - loss: 0.4946 - accuracy: 0.
7437 - val_loss: 0.8253 - val_accuracy: 0.4878
Epoch 146/300
27/27 [=====] - 0s 8ms/step - loss: 0.4972 - accuracy: 0.
7414 - val_loss: 0.8400 - val_accuracy: 0.4829
Epoch 147/300
27/27 [=====] - 0s 6ms/step - loss: 0.4898 - accuracy: 0.
7461 - val_loss: 0.8626 - val_accuracy: 0.4878
Epoch 148/300
27/27 [=====] - 0s 8ms/step - loss: 0.4862 - accuracy: 0.
7426 - val_loss: 0.8380 - val_accuracy: 0.4805
Epoch 149/300
27/27 [=====] - 0s 8ms/step - loss: 0.4823 - accuracy: 0.
7497 - val_loss: 0.8530 - val_accuracy: 0.4976
Epoch 150/300

```
27/27 [=====] - 0s 8ms/step - loss: 0.4820 - accuracy: 0.7461 - val_loss: 0.8437 - val_accuracy: 0.5098
Epoch 151/300
27/27 [=====] - 0s 9ms/step - loss: 0.4843 - accuracy: 0.7390 - val_loss: 0.8364 - val_accuracy: 0.4805
Epoch 152/300
27/27 [=====] - 0s 7ms/step - loss: 0.4796 - accuracy: 0.7485 - val_loss: 0.8389 - val_accuracy: 0.4902
Epoch 153/300
27/27 [=====] - 0s 7ms/step - loss: 0.4741 - accuracy: 0.7604 - val_loss: 0.8487 - val_accuracy: 0.4829
Epoch 154/300
27/27 [=====] - 0s 7ms/step - loss: 0.4774 - accuracy: 0.7509 - val_loss: 0.8455 - val_accuracy: 0.4951
Epoch 155/300
27/27 [=====] - 0s 7ms/step - loss: 0.4675 - accuracy: 0.7569 - val_loss: 0.8573 - val_accuracy: 0.4756
Epoch 156/300
27/27 [=====] - 0s 7ms/step - loss: 0.4681 - accuracy: 0.7557 - val_loss: 0.8643 - val_accuracy: 0.4951
Epoch 157/300
27/27 [=====] - 0s 8ms/step - loss: 0.4670 - accuracy: 0.7580 - val_loss: 0.8623 - val_accuracy: 0.4829
Epoch 158/300
27/27 [=====] - 0s 7ms/step - loss: 0.4677 - accuracy: 0.7509 - val_loss: 0.8786 - val_accuracy: 0.4878
Epoch 159/300
27/27 [=====] - 0s 7ms/step - loss: 0.4533 - accuracy: 0.7700 - val_loss: 0.8625 - val_accuracy: 0.4805
Epoch 160/300
27/27 [=====] - 0s 7ms/step - loss: 0.4495 - accuracy: 0.7712 - val_loss: 0.8664 - val_accuracy: 0.4951
Epoch 161/300
27/27 [=====] - 0s 7ms/step - loss: 0.4499 - accuracy: 0.7676 - val_loss: 0.8674 - val_accuracy: 0.4878
Epoch 162/300
27/27 [=====] - 0s 7ms/step - loss: 0.4615 - accuracy: 0.7592 - val_loss: 0.8874 - val_accuracy: 0.4902
Epoch 163/300
27/27 [=====] - 0s 7ms/step - loss: 0.4425 - accuracy: 0.7747 - val_loss: 0.8733 - val_accuracy: 0.4732
Epoch 164/300
27/27 [=====] - 0s 8ms/step - loss: 0.4458 - accuracy: 0.7616 - val_loss: 0.8792 - val_accuracy: 0.4927
Epoch 165/300
27/27 [=====] - 0s 7ms/step - loss: 0.4476 - accuracy: 0.7771 - val_loss: 0.9127 - val_accuracy: 0.4951
Epoch 166/300
27/27 [=====] - 0s 7ms/step - loss: 0.4353 - accuracy: 0.7807 - val_loss: 0.8795 - val_accuracy: 0.4902
Epoch 167/300
27/27 [=====] - 0s 7ms/step - loss: 0.4398 - accuracy: 0.7676 - val_loss: 0.8937 - val_accuracy: 0.4951
Epoch 168/300
27/27 [=====] - 0s 7ms/step - loss: 0.4316 - accuracy: 0.7783 - val_loss: 0.8840 - val_accuracy: 0.4732
```

Epoch 169/300
27/27 [=====] - 0s 7ms/step - loss: 0.4299 - accuracy: 0.7807 - val_loss: 0.9091 - val_accuracy: 0.4683

Epoch 170/300
27/27 [=====] - 0s 7ms/step - loss: 0.4291 - accuracy: 0.7807 - val_loss: 0.9210 - val_accuracy: 0.4756

Epoch 171/300
27/27 [=====] - 0s 7ms/step - loss: 0.4343 - accuracy: 0.7807 - val_loss: 0.9306 - val_accuracy: 0.4854

Epoch 172/300
27/27 [=====] - 0s 8ms/step - loss: 0.4257 - accuracy: 0.7843 - val_loss: 0.8919 - val_accuracy: 0.4878

Epoch 173/300
27/27 [=====] - 0s 7ms/step - loss: 0.4132 - accuracy: 0.7974 - val_loss: 0.9084 - val_accuracy: 0.5073

Epoch 174/300
27/27 [=====] - 0s 12ms/step - loss: 0.4076 - accuracy: 0.8010 - val_loss: 0.9325 - val_accuracy: 0.4976

Epoch 175/300
27/27 [=====] - 0s 10ms/step - loss: 0.4073 - accuracy: 0.7914 - val_loss: 0.9377 - val_accuracy: 0.5024

Epoch 176/300
27/27 [=====] - 0s 9ms/step - loss: 0.3999 - accuracy: 0.8033 - val_loss: 0.9278 - val_accuracy: 0.5000

Epoch 177/300
27/27 [=====] - 0s 10ms/step - loss: 0.4067 - accuracy: 0.7914 - val_loss: 0.9065 - val_accuracy: 0.5098

Epoch 178/300
27/27 [=====] - 0s 10ms/step - loss: 0.4049 - accuracy: 0.7890 - val_loss: 0.9258 - val_accuracy: 0.4927

Epoch 179/300
27/27 [=====] - 0s 10ms/step - loss: 0.3936 - accuracy: 0.8069 - val_loss: 0.9213 - val_accuracy: 0.4780

Epoch 180/300
27/27 [=====] - 0s 8ms/step - loss: 0.3907 - accuracy: 0.8057 - val_loss: 0.9170 - val_accuracy: 0.4878

Epoch 181/300
27/27 [=====] - 0s 10ms/step - loss: 0.3955 - accuracy: 0.8033 - val_loss: 0.9645 - val_accuracy: 0.4829

Epoch 182/300
27/27 [=====] - 0s 11ms/step - loss: 0.3886 - accuracy: 0.8069 - val_loss: 0.9428 - val_accuracy: 0.4878

Epoch 183/300
27/27 [=====] - 0s 8ms/step - loss: 0.3813 - accuracy: 0.8176 - val_loss: 0.9172 - val_accuracy: 0.4976

Epoch 184/300
27/27 [=====] - 0s 7ms/step - loss: 0.3784 - accuracy: 0.8212 - val_loss: 0.9280 - val_accuracy: 0.5049

Epoch 185/300
27/27 [=====] - 0s 7ms/step - loss: 0.3834 - accuracy: 0.8141 - val_loss: 0.9701 - val_accuracy: 0.4902

Epoch 186/300
27/27 [=====] - 0s 7ms/step - loss: 0.3668 - accuracy: 0.8212 - val_loss: 0.9612 - val_accuracy: 0.4902

Epoch 187/300
27/27 [=====] - 0s 7ms/step - loss: 0.3751 - accuracy: 0.

```
8176 - val_loss: 0.9577 - val_accuracy: 0.5024
Epoch 188/300
27/27 [=====] - 0s 8ms/step - loss: 0.3655 - accuracy: 0.
8248 - val_loss: 0.9773 - val_accuracy: 0.4854
Epoch 189/300
27/27 [=====] - 0s 7ms/step - loss: 0.3704 - accuracy: 0.
8188 - val_loss: 0.9733 - val_accuracy: 0.5073
Epoch 190/300
27/27 [=====] - 0s 8ms/step - loss: 0.3609 - accuracy: 0.
8272 - val_loss: 0.9816 - val_accuracy: 0.4951
Epoch 191/300
27/27 [=====] - 0s 7ms/step - loss: 0.3696 - accuracy: 0.
8296 - val_loss: 0.9553 - val_accuracy: 0.4902
Epoch 192/300
27/27 [=====] - 0s 8ms/step - loss: 0.3634 - accuracy: 0.
8296 - val_loss: 1.0165 - val_accuracy: 0.5098
Epoch 193/300
27/27 [=====] - 0s 8ms/step - loss: 0.3530 - accuracy: 0.
8403 - val_loss: 0.9799 - val_accuracy: 0.4780
Epoch 194/300
27/27 [=====] - 0s 8ms/step - loss: 0.3544 - accuracy: 0.
8415 - val_loss: 1.0201 - val_accuracy: 0.4927
Epoch 195/300
27/27 [=====] - 0s 7ms/step - loss: 0.3552 - accuracy: 0.
8403 - val_loss: 0.9748 - val_accuracy: 0.4878
Epoch 196/300
27/27 [=====] - 0s 8ms/step - loss: 0.3530 - accuracy: 0.
8403 - val_loss: 1.0098 - val_accuracy: 0.4854
Epoch 197/300
27/27 [=====] - 0s 7ms/step - loss: 0.3466 - accuracy: 0.
8403 - val_loss: 0.9912 - val_accuracy: 0.5073
Epoch 198/300
27/27 [=====] - 0s 8ms/step - loss: 0.3549 - accuracy: 0.
8308 - val_loss: 1.0297 - val_accuracy: 0.4902
Epoch 199/300
27/27 [=====] - 0s 7ms/step - loss: 0.3410 - accuracy: 0.
8427 - val_loss: 1.0230 - val_accuracy: 0.4878
Epoch 200/300
27/27 [=====] - 0s 9ms/step - loss: 0.3311 - accuracy: 0.
8570 - val_loss: 1.0000 - val_accuracy: 0.4951
Epoch 201/300
27/27 [=====] - 0s 8ms/step - loss: 0.3350 - accuracy: 0.
8546 - val_loss: 1.0496 - val_accuracy: 0.5073
Epoch 202/300
27/27 [=====] - 0s 7ms/step - loss: 0.3305 - accuracy: 0.
8534 - val_loss: 0.9984 - val_accuracy: 0.5000
Epoch 203/300
27/27 [=====] - 0s 6ms/step - loss: 0.3290 - accuracy: 0.
8474 - val_loss: 1.0503 - val_accuracy: 0.4756
Epoch 204/300
27/27 [=====] - 0s 8ms/step - loss: 0.3288 - accuracy: 0.
8474 - val_loss: 1.0433 - val_accuracy: 0.4683
Epoch 205/300
27/27 [=====] - 0s 7ms/step - loss: 0.3320 - accuracy: 0.
8570 - val_loss: 1.0356 - val_accuracy: 0.4927
Epoch 206/300
```

```
27/27 [=====] - 0s 7ms/step - loss: 0.3250 - accuracy: 0.
8486 - val_loss: 1.0262 - val_accuracy: 0.5024
Epoch 207/300
27/27 [=====] - 0s 7ms/step - loss: 0.3198 - accuracy: 0.
8510 - val_loss: 1.0142 - val_accuracy: 0.4756
Epoch 208/300
27/27 [=====] - 0s 8ms/step - loss: 0.3113 - accuracy: 0.
8594 - val_loss: 1.0265 - val_accuracy: 0.4805
Epoch 209/300
27/27 [=====] - 0s 7ms/step - loss: 0.3130 - accuracy: 0.
8582 - val_loss: 1.0317 - val_accuracy: 0.5000
Epoch 210/300
27/27 [=====] - 0s 7ms/step - loss: 0.3495 - accuracy: 0.
8474 - val_loss: 1.0883 - val_accuracy: 0.4878
Epoch 211/300
27/27 [=====] - 0s 7ms/step - loss: 0.3425 - accuracy: 0.
8474 - val_loss: 1.0495 - val_accuracy: 0.4951
Epoch 212/300
27/27 [=====] - 0s 7ms/step - loss: 0.3217 - accuracy: 0.
8486 - val_loss: 1.0498 - val_accuracy: 0.4951
Epoch 213/300
27/27 [=====] - 0s 8ms/step - loss: 0.3092 - accuracy: 0.
8665 - val_loss: 1.0475 - val_accuracy: 0.4976
Epoch 214/300
27/27 [=====] - 0s 7ms/step - loss: 0.3073 - accuracy: 0.
8629 - val_loss: 1.0424 - val_accuracy: 0.4878
Epoch 215/300
27/27 [=====] - 0s 7ms/step - loss: 0.3062 - accuracy: 0.
8725 - val_loss: 1.0478 - val_accuracy: 0.4902
Epoch 216/300
27/27 [=====] - 0s 8ms/step - loss: 0.2983 - accuracy: 0.
8689 - val_loss: 1.0445 - val_accuracy: 0.4878
Epoch 217/300
27/27 [=====] - 0s 8ms/step - loss: 0.2991 - accuracy: 0.
8701 - val_loss: 1.0794 - val_accuracy: 0.4902
Epoch 218/300
27/27 [=====] - 0s 7ms/step - loss: 0.2901 - accuracy: 0.
8713 - val_loss: 1.0756 - val_accuracy: 0.4902
Epoch 219/300
27/27 [=====] - 0s 7ms/step - loss: 0.2844 - accuracy: 0.
8856 - val_loss: 1.0666 - val_accuracy: 0.4878
Epoch 220/300
27/27 [=====] - 0s 7ms/step - loss: 0.3039 - accuracy: 0.
8605 - val_loss: 1.1139 - val_accuracy: 0.4659
Epoch 221/300
27/27 [=====] - 0s 8ms/step - loss: 0.2856 - accuracy: 0.
8713 - val_loss: 1.0738 - val_accuracy: 0.4805
Epoch 222/300
27/27 [=====] - 0s 8ms/step - loss: 0.2835 - accuracy: 0.
8856 - val_loss: 1.1023 - val_accuracy: 0.4976
Epoch 223/300
27/27 [=====] - 0s 8ms/step - loss: 0.2834 - accuracy: 0.
8808 - val_loss: 1.0708 - val_accuracy: 0.4927
Epoch 224/300
27/27 [=====] - 0s 8ms/step - loss: 0.2790 - accuracy: 0.
8808 - val_loss: 1.1035 - val_accuracy: 0.4927
```

Epoch 225/300
27/27 [=====] - 0s 8ms/step - loss: 0.2973 - accuracy: 0.8784 - val_loss: 1.0982 - val_accuracy: 0.4732

Epoch 226/300
27/27 [=====] - 0s 7ms/step - loss: 0.2945 - accuracy: 0.8677 - val_loss: 1.1020 - val_accuracy: 0.5049

Epoch 227/300
27/27 [=====] - 0s 7ms/step - loss: 0.2787 - accuracy: 0.8820 - val_loss: 1.1083 - val_accuracy: 0.4805

Epoch 228/300
27/27 [=====] - 0s 9ms/step - loss: 0.2714 - accuracy: 0.8820 - val_loss: 1.1165 - val_accuracy: 0.4829

Epoch 229/300
27/27 [=====] - 0s 7ms/step - loss: 0.2676 - accuracy: 0.8880 - val_loss: 1.0915 - val_accuracy: 0.4927

Epoch 230/300
27/27 [=====] - 0s 9ms/step - loss: 0.2739 - accuracy: 0.8737 - val_loss: 1.1013 - val_accuracy: 0.4732

Epoch 231/300
27/27 [=====] - 0s 9ms/step - loss: 0.2760 - accuracy: 0.8832 - val_loss: 1.1062 - val_accuracy: 0.5049

Epoch 232/300
27/27 [=====] - 0s 10ms/step - loss: 0.2859 - accuracy: 0.8725 - val_loss: 1.0810 - val_accuracy: 0.5098

Epoch 233/300
27/27 [=====] - 0s 9ms/step - loss: 0.2605 - accuracy: 0.8844 - val_loss: 1.1280 - val_accuracy: 0.4780

Epoch 234/300
27/27 [=====] - 0s 9ms/step - loss: 0.2637 - accuracy: 0.8892 - val_loss: 1.1111 - val_accuracy: 0.5049

Epoch 235/300
27/27 [=====] - 0s 10ms/step - loss: 0.2620 - accuracy: 0.8880 - val_loss: 1.1118 - val_accuracy: 0.4976

Epoch 236/300
27/27 [=====] - 0s 9ms/step - loss: 0.2533 - accuracy: 0.8999 - val_loss: 1.1658 - val_accuracy: 0.4854

Epoch 237/300
27/27 [=====] - 0s 9ms/step - loss: 0.2516 - accuracy: 0.9023 - val_loss: 1.1323 - val_accuracy: 0.4780

Epoch 238/300
27/27 [=====] - 0s 9ms/step - loss: 0.2543 - accuracy: 0.8951 - val_loss: 1.1517 - val_accuracy: 0.4756

Epoch 239/300
27/27 [=====] - 0s 10ms/step - loss: 0.2538 - accuracy: 0.9011 - val_loss: 1.1396 - val_accuracy: 0.4927

Epoch 240/300
27/27 [=====] - 0s 10ms/step - loss: 0.2584 - accuracy: 0.8892 - val_loss: 1.1486 - val_accuracy: 0.4756

Epoch 241/300
27/27 [=====] - 0s 8ms/step - loss: 0.2618 - accuracy: 0.8844 - val_loss: 1.1496 - val_accuracy: 0.4878

Epoch 242/300
27/27 [=====] - 0s 7ms/step - loss: 0.2765 - accuracy: 0.8749 - val_loss: 1.1678 - val_accuracy: 0.4634

Epoch 243/300
27/27 [=====] - 0s 7ms/step - loss: 0.2764 - accuracy: 0.

```
8892 - val_loss: 1.1346 - val_accuracy: 0.5073
Epoch 244/300
27/27 [=====] - 0s 8ms/step - loss: 0.2645 - accuracy: 0.
8868 - val_loss: 1.1573 - val_accuracy: 0.4902
Epoch 245/300
27/27 [=====] - 0s 7ms/step - loss: 0.2375 - accuracy: 0.
9023 - val_loss: 1.1860 - val_accuracy: 0.4854
Epoch 246/300
27/27 [=====] - 0s 7ms/step - loss: 0.2388 - accuracy: 0.
9011 - val_loss: 1.1472 - val_accuracy: 0.4976
Epoch 247/300
27/27 [=====] - 0s 7ms/step - loss: 0.2439 - accuracy: 0.
8927 - val_loss: 1.1578 - val_accuracy: 0.4780
Epoch 248/300
27/27 [=====] - 0s 7ms/step - loss: 0.2444 - accuracy: 0.
8951 - val_loss: 1.1391 - val_accuracy: 0.4902
Epoch 249/300
27/27 [=====] - 0s 8ms/step - loss: 0.2415 - accuracy: 0.
8975 - val_loss: 1.1644 - val_accuracy: 0.4805
Epoch 250/300
27/27 [=====] - 0s 8ms/step - loss: 0.2337 - accuracy: 0.
9035 - val_loss: 1.1877 - val_accuracy: 0.4683
Epoch 251/300
27/27 [=====] - 0s 8ms/step - loss: 0.2290 - accuracy: 0.
9058 - val_loss: 1.1825 - val_accuracy: 0.4854
Epoch 252/300
27/27 [=====] - 0s 8ms/step - loss: 0.2266 - accuracy: 0.
9070 - val_loss: 1.2203 - val_accuracy: 0.4707
Epoch 253/300
27/27 [=====] - 0s 8ms/step - loss: 0.2236 - accuracy: 0.
9011 - val_loss: 1.1816 - val_accuracy: 0.4829
Epoch 254/300
27/27 [=====] - 0s 7ms/step - loss: 0.2165 - accuracy: 0.
9118 - val_loss: 1.1979 - val_accuracy: 0.4780
Epoch 255/300
27/27 [=====] - 0s 8ms/step - loss: 0.2194 - accuracy: 0.
9118 - val_loss: 1.2188 - val_accuracy: 0.4902
Epoch 256/300
27/27 [=====] - 0s 7ms/step - loss: 0.2152 - accuracy: 0.
9082 - val_loss: 1.2177 - val_accuracy: 0.4805
Epoch 257/300
27/27 [=====] - 0s 7ms/step - loss: 0.2098 - accuracy: 0.
9201 - val_loss: 1.1876 - val_accuracy: 0.4902
Epoch 258/300
27/27 [=====] - 0s 7ms/step - loss: 0.2154 - accuracy: 0.
9094 - val_loss: 1.1987 - val_accuracy: 0.4927
Epoch 259/300
27/27 [=====] - 0s 7ms/step - loss: 0.2243 - accuracy: 0.
9011 - val_loss: 1.1957 - val_accuracy: 0.4732
Epoch 260/300
27/27 [=====] - 0s 7ms/step - loss: 0.2128 - accuracy: 0.
9094 - val_loss: 1.2217 - val_accuracy: 0.4756
Epoch 261/300
27/27 [=====] - 0s 8ms/step - loss: 0.2138 - accuracy: 0.
9070 - val_loss: 1.2445 - val_accuracy: 0.4878
Epoch 262/300
```



```
27/27 [=====] - 0s 8ms/step - loss: 0.2073 - accuracy: 0.
9118 - val_loss: 1.2129 - val_accuracy: 0.4902
Epoch 263/300
27/27 [=====] - 0s 7ms/step - loss: 0.2121 - accuracy: 0.
9130 - val_loss: 1.2541 - val_accuracy: 0.4780
Epoch 264/300
27/27 [=====] - 0s 9ms/step - loss: 0.2036 - accuracy: 0.
9118 - val_loss: 1.2068 - val_accuracy: 0.4780
Epoch 265/300
27/27 [=====] - 0s 8ms/step - loss: 0.1989 - accuracy: 0.
9237 - val_loss: 1.2303 - val_accuracy: 0.4659
Epoch 266/300
27/27 [=====] - 0s 7ms/step - loss: 0.1923 - accuracy: 0.
9201 - val_loss: 1.2626 - val_accuracy: 0.4780
Epoch 267/300
27/27 [=====] - 0s 7ms/step - loss: 0.1906 - accuracy: 0.
9237 - val_loss: 1.2640 - val_accuracy: 0.4854
Epoch 268/300
27/27 [=====] - 0s 7ms/step - loss: 0.1910 - accuracy: 0.
9249 - val_loss: 1.2693 - val_accuracy: 0.4829
Epoch 269/300
27/27 [=====] - 0s 8ms/step - loss: 0.2035 - accuracy: 0.
9166 - val_loss: 1.3117 - val_accuracy: 0.4854
Epoch 270/300
27/27 [=====] - 0s 7ms/step - loss: 0.1966 - accuracy: 0.
9190 - val_loss: 1.3092 - val_accuracy: 0.4854
Epoch 271/300
27/27 [=====] - 0s 8ms/step - loss: 0.1962 - accuracy: 0.
9166 - val_loss: 1.3029 - val_accuracy: 0.4902
Epoch 272/300
27/27 [=====] - 0s 7ms/step - loss: 0.1905 - accuracy: 0.
9237 - val_loss: 1.3052 - val_accuracy: 0.4805
Epoch 273/300
27/27 [=====] - 0s 7ms/step - loss: 0.1881 - accuracy: 0.
9166 - val_loss: 1.2788 - val_accuracy: 0.4951
Epoch 274/300
27/27 [=====] - 0s 7ms/step - loss: 0.2050 - accuracy: 0.
9154 - val_loss: 1.2671 - val_accuracy: 0.4878
Epoch 275/300
27/27 [=====] - 0s 7ms/step - loss: 0.1890 - accuracy: 0.
9237 - val_loss: 1.3159 - val_accuracy: 0.4756
Epoch 276/300
27/27 [=====] - 0s 7ms/step - loss: 0.1795 - accuracy: 0.
9261 - val_loss: 1.3009 - val_accuracy: 0.4854
Epoch 277/300
27/27 [=====] - 0s 8ms/step - loss: 0.1810 - accuracy: 0.
9285 - val_loss: 1.3556 - val_accuracy: 0.4780
Epoch 278/300
27/27 [=====] - 0s 7ms/step - loss: 0.1876 - accuracy: 0.
9249 - val_loss: 1.3417 - val_accuracy: 0.4732
Epoch 279/300
27/27 [=====] - 0s 9ms/step - loss: 0.1800 - accuracy: 0.
9201 - val_loss: 1.3190 - val_accuracy: 0.4659
Epoch 280/300
27/27 [=====] - 0s 7ms/step - loss: 0.1687 - accuracy: 0.
9321 - val_loss: 1.3237 - val_accuracy: 0.4951
```

Epoch 281/300
27/27 [=====] - 0s 8ms/step - loss: 0.1837 - accuracy: 0.9225 - val_loss: 1.3214 - val_accuracy: 0.4927

Epoch 282/300
27/27 [=====] - 0s 8ms/step - loss: 0.1799 - accuracy: 0.9273 - val_loss: 1.3021 - val_accuracy: 0.4927

Epoch 283/300
27/27 [=====] - 0s 8ms/step - loss: 0.1811 - accuracy: 0.9225 - val_loss: 1.3608 - val_accuracy: 0.4780

Epoch 284/300
27/27 [=====] - 0s 7ms/step - loss: 0.1738 - accuracy: 0.9285 - val_loss: 1.3591 - val_accuracy: 0.4805

Epoch 285/300
27/27 [=====] - 0s 7ms/step - loss: 0.1800 - accuracy: 0.9261 - val_loss: 1.3298 - val_accuracy: 0.4780

Epoch 286/300
27/27 [=====] - 0s 7ms/step - loss: 0.1613 - accuracy: 0.9344 - val_loss: 1.3889 - val_accuracy: 0.4707

Epoch 287/300
27/27 [=====] - 0s 7ms/step - loss: 0.1574 - accuracy: 0.9356 - val_loss: 1.3742 - val_accuracy: 0.4756

Epoch 288/300
27/27 [=====] - 0s 7ms/step - loss: 0.1582 - accuracy: 0.9344 - val_loss: 1.3842 - val_accuracy: 0.4878

Epoch 289/300
27/27 [=====] - 0s 9ms/step - loss: 0.2208 - accuracy: 0.8999 - val_loss: 1.3860 - val_accuracy: 0.4756

Epoch 290/300
27/27 [=====] - 0s 10ms/step - loss: 0.1999 - accuracy: 0.9142 - val_loss: 1.3886 - val_accuracy: 0.4610

Epoch 291/300
27/27 [=====] - 0s 10ms/step - loss: 0.1807 - accuracy: 0.9321 - val_loss: 1.3807 - val_accuracy: 0.4927

Epoch 292/300
27/27 [=====] - 0s 10ms/step - loss: 0.1953 - accuracy: 0.9237 - val_loss: 1.3934 - val_accuracy: 0.4902

Epoch 293/300
27/27 [=====] - 0s 10ms/step - loss: 0.2078 - accuracy: 0.9130 - val_loss: 1.3815 - val_accuracy: 0.4732

Epoch 294/300
27/27 [=====] - 0s 10ms/step - loss: 0.1693 - accuracy: 0.9297 - val_loss: 1.3876 - val_accuracy: 0.4805

Epoch 295/300
27/27 [=====] - 0s 10ms/step - loss: 0.1530 - accuracy: 0.9416 - val_loss: 1.4073 - val_accuracy: 0.4854

Epoch 296/300
27/27 [=====] - 0s 10ms/step - loss: 0.1559 - accuracy: 0.9440 - val_loss: 1.3756 - val_accuracy: 0.4854

Epoch 297/300
27/27 [=====] - 0s 10ms/step - loss: 0.1487 - accuracy: 0.9452 - val_loss: 1.4286 - val_accuracy: 0.4780

Epoch 298/300
27/27 [=====] - 0s 10ms/step - loss: 0.1462 - accuracy: 0.9404 - val_loss: 1.4199 - val_accuracy: 0.4732

Epoch 299/300
27/27 [=====] - 0s 8ms/step - loss: 0.1418 - accuracy: 0.

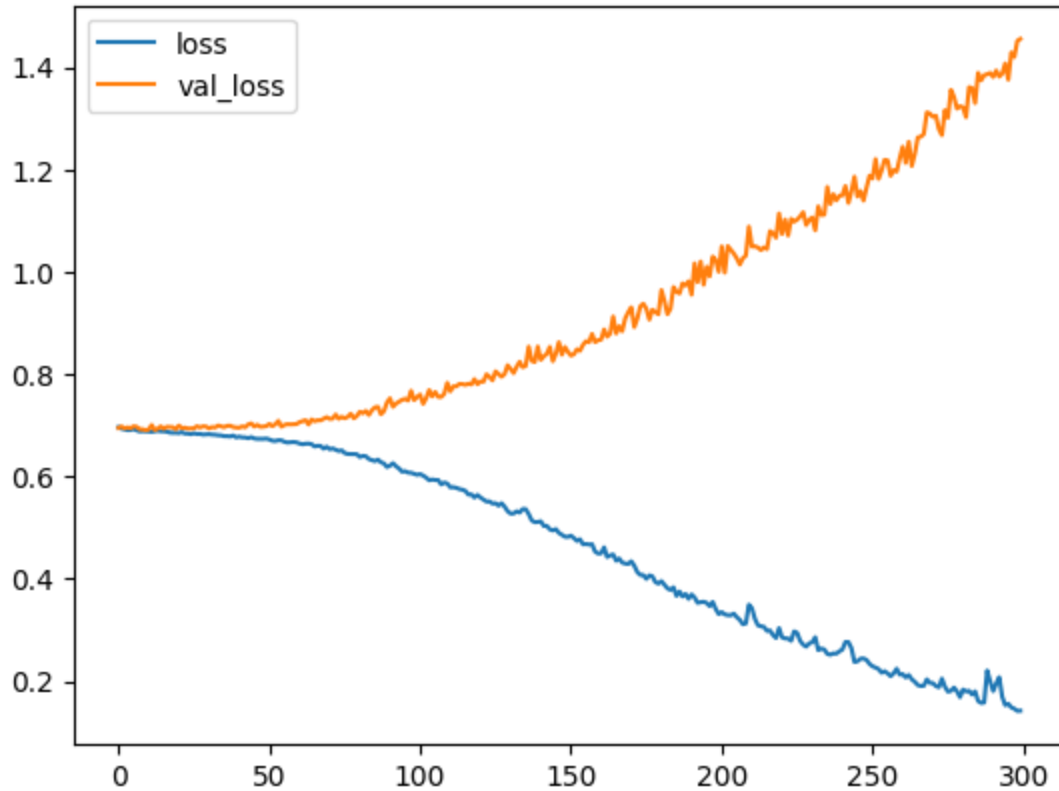
9428 - val_loss: 1.4513 - val_accuracy: 0.4634

Epoch 300/300

27/27 [=====] - 0s 7ms/step - loss: 0.1422 - accuracy: 0.

9368 - val_loss: 1.4551 - val_accuracy: 0.4854

```
In [31]: # plot the loss
plt.plot(r.history['loss'], label='loss')
plt.plot(r.history['val_loss'], label='val_loss')
plt.legend()
plt.show()
```



```
In [32]: # Plot accuracy per iteration
plt.plot(r.history['accuracy'], label='accuracy')
plt.plot(r.history['val_accuracy'], label='val_accuracy')
plt.legend()
plt.show()
```

