# This version just included the past info to predict surface temperature (did not include the weather forecast)

1st Epoch is 40

# LSTM Model Construction  
sample\_input\_shape = next(iter(train\_X.values())).shape[1:]  
model = Sequential()  
model.add(LSTM(50, return\_sequences=True, input\_shape=sample\_input\_shape))  
model.add(Dropout(0.2))  
model.add(LSTM(50))  
model.add(Dropout(0.2))  
model.add(Dense(12))  
model.compile(optimizer='adam', loss='mean\_squared\_error', metrics=['mse', 'mae'])

A graph with numbers and lines

Description automatically generated

A graph showing the temperature of a station

Description automatically generated

Your results show that the model is learning, as indicated by the decrease in loss, mean squared error (MSE), and mean absolute error (MAE) over epochs. However, the prediction plot indicates that the model may not be capturing the variation in the actual data accurately. Here are some strategies you could consider to improve the prediction:

1. **Increase the Epochs**: If the validation loss is still decreasing and hasn't plateaued, you might benefit from more training epochs. Be cautious of overfitting, though. If the training loss continues to decrease but the validation loss starts to increase, that's a sign of overfitting.
2. **Adjust Model Complexity**: If your model is too simple, it might not capture the complexities of the data (underfitting). Conversely, if it's too complex, it might capture noise in the training data (overfitting). Adjusting the number of layers, the number of units in each LSTM layer, or the dropout rate could help.
3. **Feature Engineering**: You could potentially improve model performance by adding more relevant features or removing irrelevant ones. Feature normalization or standardization could also be beneficial.
4. **Hyperparameter Tuning**: Experiment with different learning rates, batch sizes, and other hyperparameters. Automated tools like Keras Tuner or Optuna can help with this.
5. **Regularization**: If overfitting is an issue, consider adding L1/L2 regularization to the LSTM layers or increasing the dropout rate.
6. **Sequence Length**: Experiment with different lengths for the input sequences. Sometimes shorter or longer look-back periods can provide better context for predictions.
7. **Data Augmentation**: For time series, this might involve adding noise, inverting sequences, or using time-based smoothing to create additional training data.
8. **Learning Rate Schedules**: Use learning rate schedules or learning rate decay to adjust the learning rate during training, which can lead to better convergence.
9. **Ensemble Methods**: Combine predictions from multiple models to potentially improve performance.
10. **Alternative Architectures**: Try different RNN architectures like GRU, or even other approaches like Temporal Convolutional Networks (TCNs) or Transformer-based models.
11. **Data Quality**: Ensure that the data is clean, relevant, and that any temporal dynamics are appropriately accounted for.
12. **Evaluation on Test Set**: Finally, make sure to evaluate your model on a test set that the model hasn't seen during training to assess generalization performance.

When you make these changes, track the changes meticulously and test one change at a time to understand its impact. Use a systematic approach like cross-validation to evaluate the performance of different models and hyperparameters.

Top of Form

2nd I will increase the epoch from 40 to 100 to see what will happen:

A graph with a line

Description automatically generated

A graph showing the temperature of a station

Description automatically generated

3rd The main model had some issue (didn’t include the surface temp in the training past data), fixed that:

100 Epoc:

A graph with a line

Description automatically generated

A graph with green and blue lines

Description automatically generatedA graph with lines and numbers

Description automatically generated with medium confidenceA graph with green line and red line

Description automatically generated

**Added learning rate: 0.001**

model = Sequential()  
model.add(LSTM(50, return\_sequences=True, input\_shape=sample\_input\_shape))  
model.add(Dropout(0.2))  
model.add(LSTM(50))  
model.add(Dropout(0.2))  
model.add(Dense(12))  
optimizer = tf.keras.optimizers.Adam(learning\_rate=0.001)  
model.compile(optimizer=optimizer, loss='mean\_squared\_error', metrics=['mse', 'mae'])

A graph with a line

Description automatically generated

A graph with green lines and red and blue lines

Description automatically generated

A graph with lines and numbers

Description automatically generated with medium confidence

A graph with lines and numbers

Description automatically generated with medium confidence

**learning rate: 0.0001, Epoc 100**

**A graph with a line

Description automatically generated** **A graph with green line and red line

Description automatically generated**

**A graph with different colored lines

Description automatically generated**

**A graph with lines and numbers

Description automatically generated with medium confidence**

**A graph with different colored lines

Description automatically generated** **A graph with lines and numbers

Description automatically generated with medium confidence**

**A graph showing the growth of a stock market

Description automatically generated**

**A graph with colored lines

Description automatically generated**

**A graph with green line and red line

Description automatically generated**

**learning rate: 0.0001, Epoc 200**

**A graph of a function

Description automatically generated**

A graph with a line

Description automatically generated with medium confidence

A graph with lines and numbers

Description automatically generated with medium confidence

A graph showing the growth of a stock market

Description automatically generated with medium confidence

**learning rate: 0.0005, Epoc 250**

**A graph of a graph

Description automatically generated with medium confidence**

**A graph with lines and numbers

Description automatically generated with medium confidence**

**A graph with green line and red line

Description automatically generated**

**A graph showing a line

Description automatically generated with medium confidence**

**A graph with green lines and red and blue lines

Description automatically generated**

**A graph with green line and red line

Description automatically generated**

**learning rate: 0.0001, Epoc 250**

**A graph of a function

Description automatically generated**

**A graph with colored lines

Description automatically generated** **A graph with lines and numbers

Description automatically generated with medium confidence**

A graph showing the growth of a stock market

Description automatically generated A graph with green lines and red and blue lines

Description automatically generated

A graph with lines and numbers

Description automatically generated with medium confidence A graph with green line and red line

Description automatically generated

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 134 | 100 | 200 | 250 | 300 |
| 0.001 |  |  |  |  |
| 0.0005 |  |  | **A graph with lines and numbers  Description automatically generated with medium confidence** |  |
| 0.0001 | **A graph with different colored lines  Description automatically generated** | A graph with a line  Description automatically generated with medium confidence | **A graph with colored lines  Description automatically generated** |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 453 | 100 | 200 | 250 | 300 |
| 0.001 | **A graph with green line and red line  Description automatically generated** |  |  |  |
| 0.0005 |  |  | **A graph with green line and red line  Description automatically generated** |  |
| 0.0001 |  |  | A graph with lines and numbers  Description automatically generated with medium confidence |  |

|  |  |  |  |
| --- | --- | --- | --- |
| 462 | 100 | 200 | 250 |
| 0.001 | **A graph with lines and numbers  Description automatically generated with medium confidence** |  |  |
| 0.0005 |  |  | **A graph with green line and red line  Description automatically generated** |
| 0.0001 |  | A graph with lines and numbers  Description automatically generated with medium confidence | **A graph with lines and numbers  Description automatically generated with medium confidence** |

|  |  |  |  |
| --- | --- | --- | --- |
| 529 | 100 | 200 | 250 |
| 0.001 | **A graph with colored lines  Description automatically generated** |  |  |
| 0.0005 |  |  | **A graph with green lines and red and blue lines  Description automatically generated** |
| 0.0001 |  |  | A graph with green lines and red and blue lines  Description automatically generated |

# This version just included the past info and the weather forecast to predict surface temperature. (model 1)

model.add(LSTM(50, return\_sequences=True, input\_shape=sample\_input\_shape))  
model.add(Dropout(0.2))  
model.add(LSTM(50))  
model.add(Dropout(0.2))  
model.add(Dense(12))

Epoc 100 learning rate: 0.0005

A graph with blue and orange lines

Description automatically generated

A graph with a line

Description automatically generated with medium confidenceA graph with lines and numbers

Description automatically generated with medium confidence

A graph showing a line

Description automatically generated with medium confidenceA graph with green lines and red and blue lines

Description automatically generated

A graph with a line and a red x

Description automatically generated with medium confidenceA graph with lines and numbers

Description automatically generated with medium confidence

Epoc 250 learning rate: 0.0005

A graph of a graph

Description automatically generated with medium confidence

A graph with colored lines

Description automatically generatedA graph with lines and numbers

Description automatically generated with medium confidence

A graph showing the growth of a stock market

Description automatically generated A graph with green line and red line

Description automatically generated

A graph with a line and a cross

Description automatically generated with medium confidence A graph with green line and red line

Description automatically generated

Epoc 250 learning rate: 0.0001

A graph of a function

Description automatically generated

A graph with a line

Description automatically generated with medium confidence A graph with lines and a line

Description automatically generated with medium confidence

A graph showing the growth of a stock market

Description automatically generated A graph with green line and red line

Description automatically generated

A graph with lines and a cross

Description automatically generated with medium confidence A graph with lines and numbers

Description automatically generated with medium confidence

# This version just included the past info and the weather forecast to predict surface temperature. (model 2)

model. Add(Bidirectional(LSTM(100, return\_sequences=True), input\_shape=sample\_input\_shape))  
model.add(Dropout(0.3))  
model.add(BatchNormalization())  
  
model.add(Bidirectional(LSTM(100, return\_sequences=True)))  
model.add(Dropout(0.3))  
model.add(BatchNormalization())  
  
model.add(Bidirectional(LSTM(100)))  
model.add(Dropout(0.3))  
model.add(BatchNormalization())  
  
model.add(Dense(50, activation='relu'))  
model.add(Dropout(0.3))  
  
model.add(Dense(12)) # Assuming your output size is 12

Epoch: 250 and LR = 0.0001

A graph of a graph

Description automatically generated with medium confidence

A graph with colored lines

Description automatically generated A graph with lines and a line

Description automatically generated with medium confidence

A graph showing a line

Description automatically generated with medium confidence A graph with green line and red line

Description automatically generated

A graph with lines and a line

Description automatically generated with medium confidence A graph with different colored lines

Description automatically generated

Epoch: 500 and LR = 0.0001

A graph with numbers and lines

Description automatically generated

A graph with lines and numbers

Description automatically generated with medium confidence A graph with lines and a line

Description automatically generated with medium confidence

A graph showing the growth of a stock market

Description automatically generated with medium confidence A graph with lines and a red and blue line

Description automatically generated

A graph with a line and a line

Description automatically generated with medium confidence A graph with lines and numbers

Description automatically generated with medium confidence