at kind of adjustments can I make to the model
to the actual graph?

e

actual graph. One approach could be to experiment with the hyperparameters of your model, such as the number of layers, the number of LSTM units, and the sequence length. You can try different combinations of these hyperparameters to see if they improve the performance of your model.

Another approach could be to try different data preprocessing techniques. For example, you could try normalizing or scaling your input data to see if that improves the performance of your model. Additionally, you could try using different feature engineering techniques to extract more relevant information from your input data.

It's important to keep in mind that improving the performance of a machine learning model is an iterative process and may require some trial and error. 😊

5 of 20

Ok here are the elements of the models that are different. All other code is exactly the same. Model 7 has 4 layers, 350 lstm cells and a sequence length of 60. Model 11 has 4 layers, 350 lstm cells and sequence length of 60. Keeping this in mind can you re answer my previous question?

Based on the information you provided, it seems like models 7 and 11 have the same architecture with 4 layers and 350 LSTM cells. However, you mentioned that model 11 has a sequence length of 120 in your earlier message, while in this message you mentioned that it has a sequence length of 60. Could you please clarify the sequence length of model 11?

In general, the sequence length is an important hyperparameter that determines how many time steps of the input data the model takes into account when making predictions. A longer sequence length means that the model takes into account more historical data when making predictions. You could try experimenting with different sequence lengths to see if that improves the performance of your models. 😊

6 of 20

Model 11 has sequence length of 120.

Thank you for clarifying the sequence length of model 11. Since model 11 has a longer sequence length than model 7 (120 vs 60), it takes into account more historical data when making predictions. This could potentially improve the performance of the model, but it could also make the model more computationally expensive to train.

You could try experimenting with different sequence lengths to see if that improves the performance of your models. You could also try other approaches I mentioned earlier, such as experimenting with the number of layers and LSTM units, trying different data preprocessing techniques, and using feature engineering to extract more relevant information from your input data. 😊

7 of 20

So I noticed that a lower sequence length seems to move the predictions graph forward on the x axis. So should I keep lowering the sequence length till it gets to the correct x axis position relative to the actual graph

⚠ Sorry, looks like something went wrong. Let's start over.

Ask me anything...

Feedback

at kind of adjustments can I make to the model
to the actual graph?

e

actual graph. One approach could be to experiment with the hyperparameters of your model, such as the number of layers, the number of LSTM units, and the sequence length. You can try different combinations of these hyperparameters to see if they improve the performance of your model.

Another approach could be to try different data preprocessing techniques. For example, you could try normalizing or scaling your input data to see if that improves the performance of your model. Additionally, you could try using different feature engineering techniques to extract more relevant information from your input data.

It's important to keep in mind that improving the performance of a machine learning model is an iterative process and may require some trial and error. 😊

5 of 20

Ok here are the elements of the models that are different. All other code is exactly the same. Model 7 has 4 layers, 350 lstm cells and a sequence length of 60. Model 11 has 4 layers, 350 lstm cells and sequence length of 60. Keeping this in mind can you re answer my previous question?

Based on the information you provided, it seems like models 7 and 11 have the same architecture with 4 layers and 350 LSTM cells. However, you mentioned that model 11 has a sequence length of 120 in your earlier message, while in this message you mentioned that it has a sequence length of 60. Could you please clarify the sequence length of model 11?

In general, the sequence length is an important hyperparameter that determines how many time steps of the input data the model takes into account when making predictions. A longer sequence length means that the model takes into account more historical data when making predictions. You could try experimenting with different sequence lengths to see if that improves the performance of your models. 😊

6 of 20

Model 11 has sequence length of 120.

Thank you for clarifying the sequence length of model 11. Since model 11 has a longer sequence length than model 7 (120 vs 60), it takes into account more historical data when making predictions. This could potentially improve the performance of the model, but it could also make the model more computationally expensive to train.

You could try experimenting with different sequence lengths to see if that improves the performance of your models. You could also try other approaches I mentioned earlier, such as experimenting with the number of layers and LSTM units, trying different data preprocessing techniques, and using feature engineering to extract more relevant information from your input data. 😊

7 of 20

So I noticed that a lower sequence length seems to move the predictions graph forward on the x axis. So should I keep lowering the sequence length till it gets to the correct x axis position relative to the actual graph

⚠ Sorry, looks like something went wrong.

at kind of adjustments can I make to the model
to the actual graph?

e
actual graph. One approach could be to experiment with the hyperparameters of your model, such as the number of layers, the number of LSTM units, and the sequence length. You can try different combinations of these hyperparameters to see if they improve the performance of your model.

Another approach could be to try different data preprocessing techniques. For example, you could try normalizing or scaling your input data to see if that improves the performance of your model. Additionally, you could try using different feature engineering techniques to extract more relevant information from your input data.

It's important to keep in mind that improving the performance of a machine learning model is an iterative process and may require some trial and error. 😊

5 of 20

Ok here are the elements of the models that are different. All other code is exactly the same. Model 7 has 4 layers, 350 lstm cells and a sequence length of 60. Model 11 has 4 layers, 350 lstm cells and sequence length of 60. Keeping this in mind can you re answer my previous question?

Based on the information you provided, it seems like models 7 and 11 have the same architecture with 4 layers and 350 LSTM cells. However, you mentioned that model 11 has a sequence length of 120 in your earlier message, while in this message you mentioned that it has a sequence length of 60. Could you please clarify the sequence length of model 11?

In general, the sequence length is an important hyperparameter that determines how many time steps of the input data the model takes into account when making predictions. A longer sequence length means that the model takes into account more historical data when making predictions. You could try experimenting with different sequence lengths to see if that improves the performance of your models. 😊

6 of 20

Model 11 has sequence length of 120.

Thank you for clarifying the sequence length of model 11. Since model 11 has a longer sequence length than model 7 (120 vs 60), it takes into account more historical data when making predictions. This could potentially improve the performance of the model, but it could also make the model more computationally expensive to train.

You could try experimenting with different sequence lengths to see if that improves the performance of your models. You could also try other approaches I mentioned earlier, such as experimenting with the number of layers and LSTM units, trying different data preprocessing techniques, and using feature engineering to extract more relevant information from your input data. 😊

7 of 20

So I noticed that a lower sequence length seems to move the predictions graph forward on the x axis. So should I keep lowering the sequence length till it gets to the correct x axis position relative to the actual graph

⚠ Sorry, looks like something went wrong.