Avinash Chouhan
CISC 484 - HW5

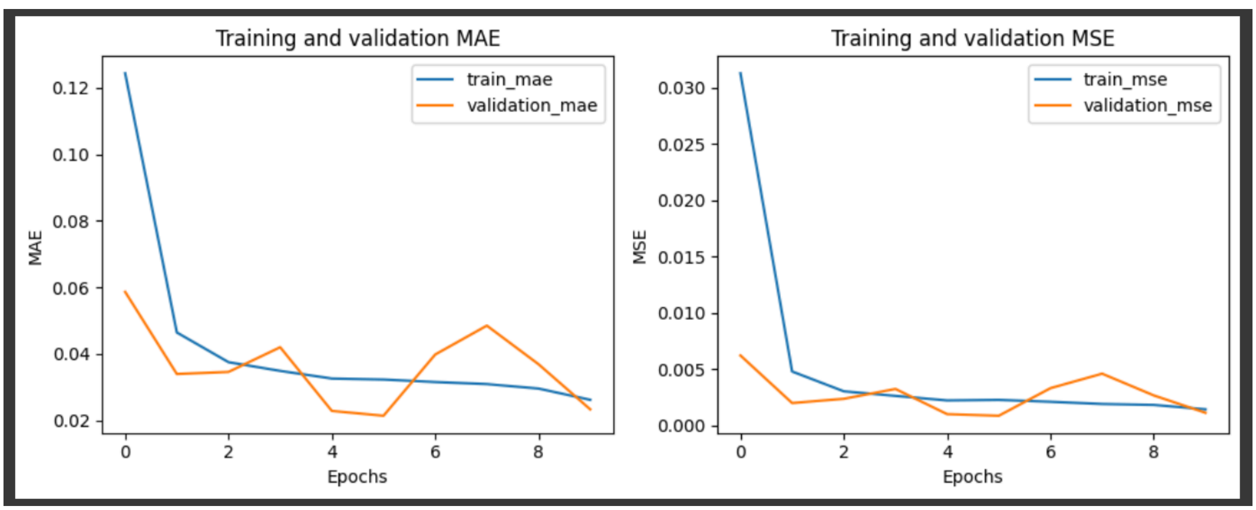
[*GPT-4o*] Summary: Among the three optimizers, **Adam** performed the best due to its low MAE and MSE values and fast convergence, minimizing loss drastically by epoch 1. However, it showed instability and overfitting, with large gaps between training and validation errors. **SGD** demonstrated average performance, with stable error curves and gradual convergence by epoch 10, but also suffered from overfitting. **RMSprop** performed the worst, with high error values, unstable learning curves, and significant overfitting, making it unsuitable for this task.

Adam (Adaptive Moment Estimation)

Model: "sequential_4"

Layer (type)	Output Shape	Param #
lstm_14 (LSTM)	(None, 60, 50)	10,400
lstm_15 (LSTM)	(None, 60, 50)	20,200
lstm_16 (LSTM)	(None, 60, 50)	20,200
lstm_17 (LSTM)	(None, 50)	20,200
dropout_4 (Dropout)	(None, 50)	0
dense_11 (Dense)	(None, 50)	2,550
dense_12 (Dense)	(None, 25)	1,275
dense_13 (Dense)	(None, 1)	26

Total params: 224,555 (877.17 KB)
Trainable params: 74,851 (292.39 KB)
Non-trainable params: 0 (0.00 B)
Optimizer params: 149,704 (584.79 KB)



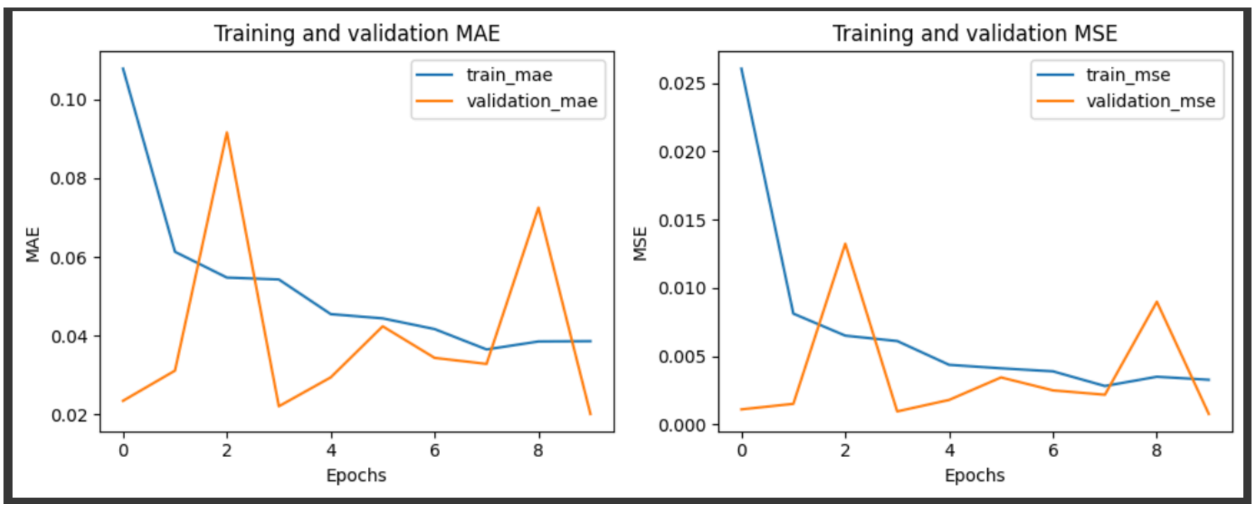
The ADAM optimizer has shown to be the best out of the three due to its low MAE and MSE values. The convergence for this optimizer is also very fast as it minimizes loss drastically by epoch 1 and continues to minimize more gradually by epoch 10. However, this is unstable and does not fit the data perfectly. The error curves are not very smooth and gaps between the validation and training data are large showing overfitting in the model.

RMSprop

Model: "sequential_5"

Layer (type)	Output Shape	Param #
lstm_18 (LSTM)	(None, 60, 50)	10,400
lstm_19 (LSTM)	(None, 60, 50)	20,200
lstm_20 (LSTM)	(None, 60, 50)	20,200
lstm_21 (LSTM)	(None, 50)	20,200
dropout_5 (Dropout)	(None, 50)	0
dense_14 (Dense)	(None, 50)	2,550
dense_15 (Dense)	(None, 25)	1,275
dense_16 (Dense)	(None, 1)	26

Total params: 149,704 (584.79 KB)
Trainable params: 74,851 (292.39 KB)
Non-trainable params: 0 (0.00 B)
Optimizer params: 74,853 (292.40 KB)



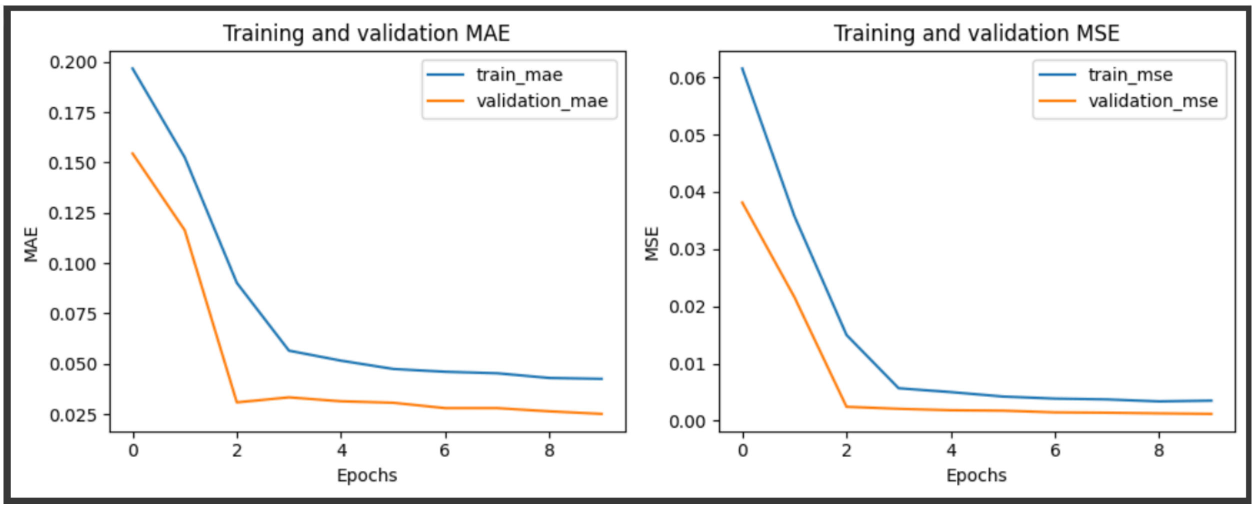
The RMSprop optimizer has shown to be the worst out of the three due to its high MAE and MSE values. The convergence for this optimizer is also very gradual as it minimizes loss to a smaller magnitude by epoch 1 in comparison to other optimizers. This is also very unstable and does not fit the data perfectly. The error curves are not smooth and gaps between the validation and training data are large showing overfitting in the model.

SGD (Stochastic Gradient Descent)

Model: "sequential_6"

Layer (type)	Output Shape	Param #
lstm_22 (LSTM)	(None, 60, 50)	10,400
lstm_23 (LSTM)	(None, 60, 50)	20,200
lstm_24 (LSTM)	(None, 60, 50)	20,200
lstm_25 (LSTM)	(None, 50)	20,200
dropout_6 (Dropout)	(None, 50)	0
dense_17 (Dense)	(None, 50)	2,550
dense_18 (Dense)	(None, 25)	1,275
dense_19 (Dense)	(None, 1)	26

Total params: 149,704 (584.79 KB)
Trainable params: 74,851 (292.39 KB)
Non-trainable params: 0 (0.00 B)
Optimizer params: 74,853 (292.40 KB)



The SGD optimizer is the average out of the three. The convergence for this optimizer is average as it minimizes loss drastically by epoch 2 rather than epoch 1, like in the ADAM optimizer, and continues to minimize more gradually by epoch 10. However, this is stable, due to its smooth error curves. But the gaps between the validation and training data are large showing consistent overfitting in the model.