Question 1:
(c)
For the first RNN model (with one hidden layer):
ClippedGradientDescentOptimizer does the best job. Both train loss and validation loss gained by ClippedGradientDescentOptimizer are smaller than that gained by MomentumOptimizer, though the difference between these two results are really small. Therefore, training and validation performance are monotonically related.

(d)
For the LSTM model:
Both ClippedGradientDescentOptimizer and MomentumOptimizer get totally terrible results this time. Their train loss and validation loss diverge at EPOCH 3 and EPOCH 2 respectively. Before divergence, ClippedGradientDescentOptimizer gets smaller train loss and validation loss than MomentumOptimizer, so training and validation performance are monotonically related.

(e)
For the second RNN model (with two hidden layers)
ClippedGradientDescentOptimizer does the best job. Both train loss and validation loss gained by ClippedGradientDescentOptimizer are smaller than that gained by MomentumOptimizer, though the difference between these two results are really small. Therefore, training and validation performance are monotonically related.

(f)
When compare these three models, I
found ClippedGradientDescentOptimizer totally
beats MomentumOptimizer no matter which models they use. The second phenomenon is that RNN model with 2 hidden layers is the best among these three models. All RNN (with 2 hidden layers) models with different optimizers get better results (lower error) in validation and test data compared to other models, even the worst model in RNN (with 2 hidden layers) models can beat the best model in the other models. RNN models with one hidden layer get not bad result, their performance are just a little bit worse than RNN models with two hidden layers. LSTM models get totally terrible results this time. Their train loss and validation loss diverge at EPOCH 3 and EPOCH 2 respectively.