A naïve LSTM implementation, performance tested on

- ☑ image classification (MNIST)
- ☑ sentiment analysis (IMDB)
- ☑ language modelling (Penn Tree)

Experiment results

Image classification

We use the MNIST dataset to train our model for image classification. The overall model architecture is Embedding + LSTM + Average Pooling + Linear + Softmax + Cross Entropy Loss. The models are trained using the SGD optimizer, with the batch size of 64.

The TensorFlow model is trained with learning rate of 1.0 for 20 epochs, whereas my model is trained with initial learning rate of 1.0 and decay rate of 0.2 for 100 epochs.

Model	**test accuracy**	**train accuracy**	**train loss**
TensorFlow	96.8	99.4	0.023
my model	93.4	97.2	0.133

Sentiment analysis

We use the IMDB dataset for sentiment analysis (binary text classification). The model architecture is Embedding + LSTM + Average Pooling + Linear + Softmax + Cross Entropy Loss. The embedding lookup matrix is initialized with the Glove word vectors, without updating during training.

My model is trained with the SGD optimizer early stopped at 50 epochs where the gradients begin exploding. The hyperparameters settings are as follows:

- random seed = 42
- batch size=32
- hidden size=32
- initial learning rate = 1.0
- learning rate decay rate = 0.2
- weight initialization: W ~ 0.1* N(0, 1) for LSTM layer, W ~ N(0, 1) for linear layer
- gradient threshold = 5

For comparison, the model written in TensorFlow is trained at learning rate 1 for 20 epochs, with batch size=32, hidden size=32.

Model	**test accuracy**	**train accuracy**	**train loss**
TensorFlow	73.8	87.8	0.301
my model	73.0	81.7	0.460

Language modelling

For the language modelling task, we choose the Penn Tree Bank dataset. The model architecture is Embedding layer + LSTM + Linear Layer + Softmax Layer, the loss is cross entropy. The linear layer share parameters across time.

The TensorFlow model is trained with SGD optimizer with learning rate = 1.0 for almost 2 hours. My model is trained with SGD optimizer with learning rate = 1.0 for almost 7 hours.

Model	test loss / perplexity	train loss / perplexity
TensorFlow (SGD 50 epochs)	6.23 / 507.76	6.28 / 533.79
my model (SGD 50 epochs)	6.15 / 468.71	6.19 / 487.85
TensorFlow(Adam 50 epochs)	5.85	5.86
TensorFlow (Adam 100 epochs)	5.71	5.65