Brief Recurrent Neural Networks Study

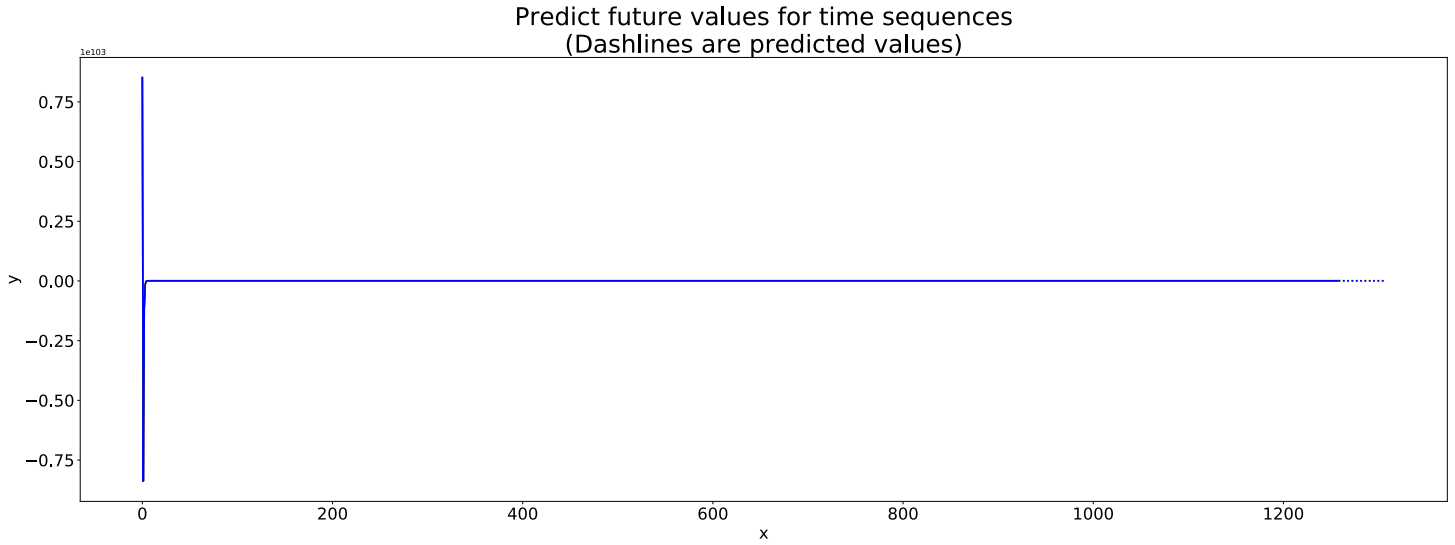
Brief overview of why gated units are necessary in recurrent neural networks: vanishing and exploding gradients.

Spencer Bertsch – Winter 2019
Dartmouth College

Training 3 layer RNN

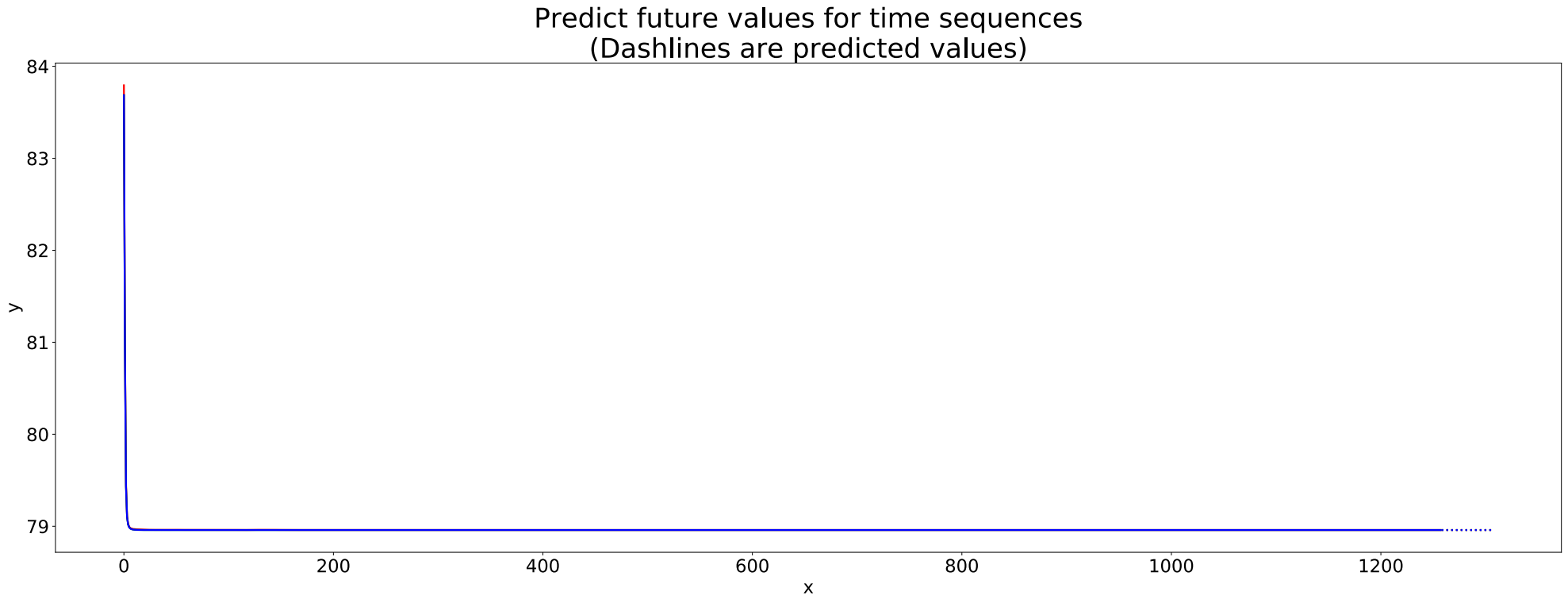
Predicting stock using a gated RNN.

Detecting patterns in the stock market is an extremely difficult task so my hopes for the success of this simple proof of concept project were low. Still, it was a good exercise in order to get experience developing recurrent networks with LSTM units to mitigate the effects of vanishing gradients.



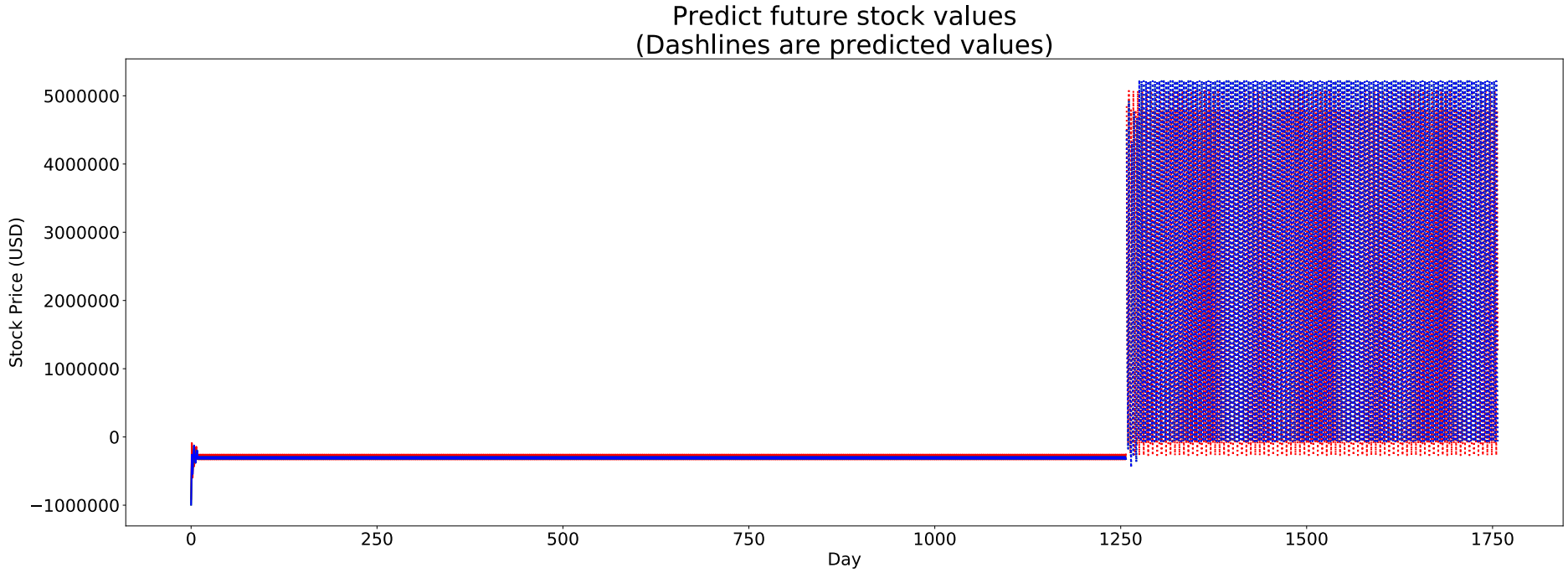
Vanishing gradients

Using only 20 LSTM units in the hidden layer, the important information stored in the earlier (lower t value) features were getting lost in the training process, leaving the resulting model unable to generate a functional representation of market movement.



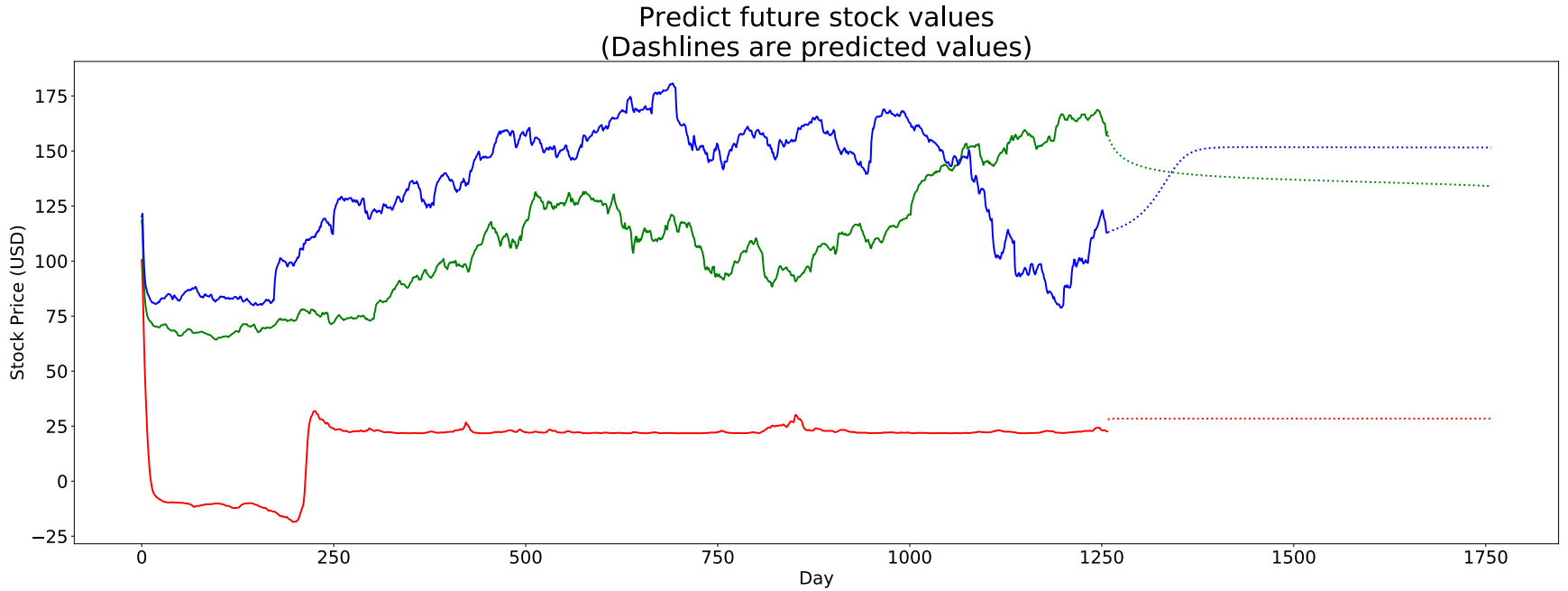
Exploding gradients

After the number of units was increased to 300, the gradients computed during backpropagation "exploded" rendering them useless and producing a useless model.



Limited success

When the number of gated units in the hidden layer was adjusted and the number of epochs was increased, the model was able to learn well enough to model the three stocks in question. However, when the model was asked to predict into the future, the predictions immediately reached steady state. If I were to continue studying this area, I would include some type of gradient clipping to reduce exploding gradients and experiments to determine the optimal network topology for this application.



Final notes

Instead of using gated RNNs for finance applications, I will shift focus over to NLP applications, but continue studying optimized encoder-decoder or sequence to sequence models. Still, further successes could be realized in the area of finance by employing other more sophisticated techniques other than simple gated RNNs such as using attention. In addition, the task of predicting movement in equity markets is a particularly challenging one; shifting focus to a more easily modeled finance application would yield better results. (But what's the fun in that really?)