Model Documentation

Model Introduction:

Long Short Term Memory networks – usually just called “LSTMs” – are a special kind of RNN, capable of learning long-term dependencies. They were introduced by Hochreiter & Schmidhuber (1997), and were refined and popularized by many people in following work. They work tremendously well on a large variety of problems, and are now widely used.

LSTM networks are very good at holding long term memories. Or in other words, the prediction of nth sample in a sequence of test samples can be influenced by an input that was given many time steps before. The memory may or may not be retained by the network depending on the data. Preserving the long term dependencies in the network is done by its Gating mechanisms. The network can store or release memory on the go through the gating mechanism.LSTM is a good choice for such sequences which have long term dependencies in it.

And since we have big data base, LSTMs are the best choice.

Model Architecture:

For our LSTM model we used the following architecture:

No: of parameters = 11

"model":

"loss": "mse",

"optimizer": "adam"

Total layers : 6

Layer 1

"type": "lstm",

"neurons": 100,

"input_timesteps": 49, (time window)

"input_dim": 11,

Layer 2{

"type": "dropout",

"rate": 0.2

},

Layer 3{

"type": "lstm",

"neurons": 150,

"return_seq": true

},

Layer 4{

"type": "lstm",

"neurons": 100,

"return_seq": false

},

Layer 5{
"type": "dropout",
"rate": 0.2
},
Layer 6{
"type": "dense",
"neurons": 1,
"activation": "linear"
}

Parameters used

"columns":
["bid","ask","bsize","asize","spread","price","size","arrival_rate","Trade","Buy","mid"]

Trade : it signifies whether trade happened or not
Buy : it signifies whether the trade that happened was a buy order or sell order
Arrival_rate: Rate of trade