Neural Network-LSTM model for Multivariate Time Series Data Prediction

Megdam Ahmed Chowdhury and Tahsin Shahnewaz

Department of Mathematics, Texas A&M University-Commerce

Mentor: Dr. Nikolay M. Sirakov

The title of our project is **Neural Network-LSTM model for Multivariate Time Series Data Prediction**. I, Tahsin Shahnewaz and my partner Megdam Ahmed Chowdhury from the department of mathematics, TAMUC is presenting the poster under the supervision of our mentor Dr. Nikolay M. Sirakov.

In our project, we used multivariate time series stock price data that has a mixture of long and short-term data. After processing it, we used LSTM layers for sequential analysis to predict the outcome of opening prices for future 90 days. We validated our Neural Network using multi attribute stock price historical data. Traditional approaches may fail in this process due to time delay and gradient disappearance. LSTM model has numerous applications such as Time Series Prediction, Image Captioning, Speech Recognition etc.

In our background section, we have shown an algorithm that has been used for LSTM model for stock price prediction. First, we analyzed the historical stock price data and converted our data frame into numpy array. Then we normalized the dataset and built our LSTM model in Keras. After training, predicting and visualizing the result, we have come up with the graph of the historical stock price along with our predicted future 90 days' opening prices.

Now I'm going to explain the internal structure of our model.

An LSTM model is built of 3 gates.

(1)The Forgotten Gate is the first step in this process. It is used to determine the information that is discarded. σ refer to the activation function of Sigmoid, w refers to the weight, and b refers to the offset. Sigmoid outputs a value between 0 and 1.

(2) The next step involves the New Memory Network and the Input Gate. The goal of this step is to determine what new information should be added to the networks long-term memory (cell state). i_t refers to how much information needs to be updated by Sigmoid, 0 means no information should be updated, and 1 means completely updated.

(3) The last step is the Output Gate which determines the output information. It decides the new hidden state. t_o refers to how much information needs to be output, and 0 means no information is output, and 1 means to output all the information.

To get the results,

We trained the LSTM on the data using Keras in Python. We used 2 LSTM layers for sequential analysis. The model is applied for 50 training epochs with a batch size of 16 for comparison. After testing for over 59 minutes, the validation loss came up to 8.9% and the accuracy of the

model is 91.1%. The graphical representation show the trend of the previous prices along with the trend of the opening price for future 90 days.

To conclude we can say,

In this paper, we have shown how a time series data can be transformed to a format that is usable to LSTM time series forecasting. The network learns features from the previous data for predicting the future value. We have come up with a better fitting degree and improved accuracy of the prediction results. The paper can be further developed by adding predictions related to stock news to enhance the stability and accuracy.