Stock Index Prediction with Machine Learning and Deep Learning Models

Hong Thanh Hoai, Mai Hoang Lan, Nguyen Thi Hong Phuc

Vingroup Big Data Institute

December 28, 2020

Table of Contents

Problem Overview

Why Predict Stock?

- Maximize profits
- Predict the economy
- Implement suitable economic policies

Challenges

- Stochastic nature
- Multiple factors

Project Objectives

What are the Goals?

- Build a working ARIMA (Autoregressive Moving Average) model
- Build a working LSTM model
- Build a working CNN model
- Build a working feature fusion LSTM - CNN model
- Outputs: predicted daily closing for Dow Jones Industrial Average (DJIA)

$$DJIA = \frac{\sum stock\ price}{d}; \text{ Dow divisor: } d \approx 0.152$$

Data Overview - Dow Jones 2009-2017

	Date	Open	High	Low	Close	Adj Close	Volume
0	2009-01-02	8772.250000	9065.280273	8760.780273	9034.690430	9034.690430	213700000
1	2009-01-05	9027.129883	9034.370117	8892.360352	8952.889648	8952.889648	233760000
2	2009-01-06	8954.570313	9088.059570	8940.950195	9015.099609	9015.099609	215410000
3	2009-01-07	8996.940430	8996.940430	8719.919922	8769.700195	8769.700195	266710000
4	2009-01-08	8769.940430	8770.019531	8651.190430	8742.459961	8742.459961	226620000

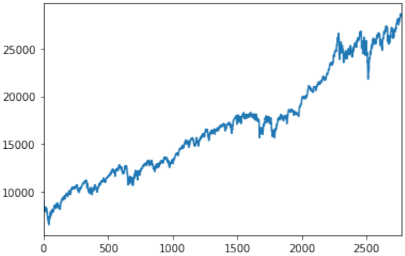
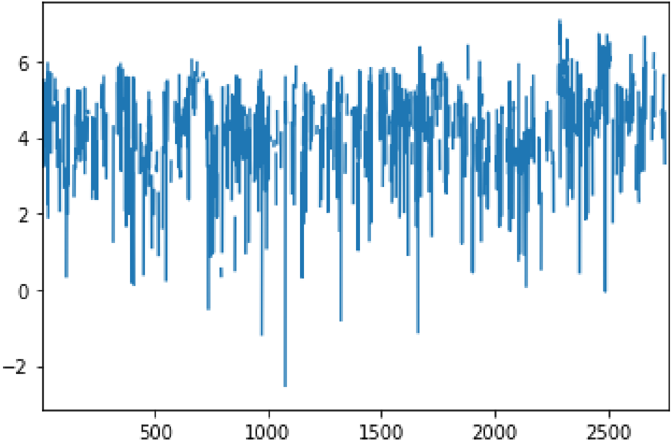
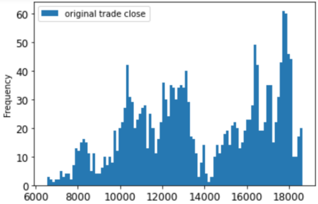


Figure: 2767 days in total. (Train set: 1660 — Test set: 553)

Data Overview - Dow Jones 2009-2017



ARIMA Model Results



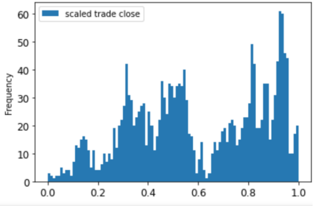


Figure: Original Trade Close and Scaled Trade Close

ARIMA Model Results

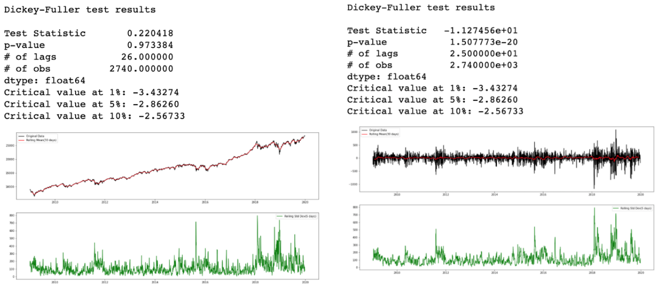


Figure: Test for Data Stationarity

ARIMA Model Results

Hyper-parameters estimation:

- Differencing (d): make time series stationary, avoding ovr differenced series.
- Auto-Regression AR (p): Investigating Partial Auto-correction (PACF) for defining p
- Moving Average MA (q): Investigating Auto-correlation (ACF) for estimating q

ARIMA Model Results

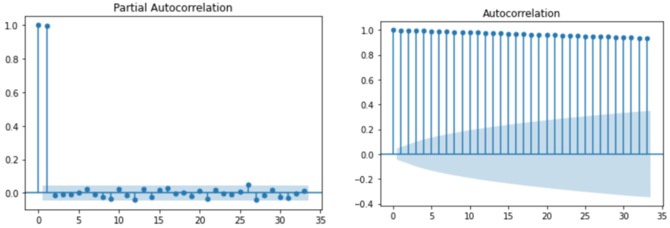


Figure: PACF and ACF plot

ARIMA Model Results

```
                               SARIMAX Results
==============================================================================
Dep. Variable:                  Close   No. Observations:                 1936
Model:                 ARIMA(1, 1, 1)   Log Likelihood                6068.774
Date:                Sun, 27 Dec 2020   AIC                         -12131.548
Time:                        02:43:23   BIC                         -12114.844
Sample:                             0   HQIC                        -12125.404
                               - 1936
Covariance Type:                  opg
==============================================================================
                 coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------
ar.L1         -0.5159      0.254     -2.033      0.042      -1.013      -0.018
ma.L1          0.4718      0.262      1.797      0.072      -0.043       0.986
sigma2         0.0001    2.5e-06     44.166      0.000       0.000       0.000
===================================================================================
Ljung-Box (L1) (Q):                   0.01   Jarque-Bera (JB):               409.21
Prob(Q):                              0.94   Prob(JB):                         0.00
Heteroskedasticity (H):               1.73   Skew:                            -0.32
Prob(H) (two-sided):                  0.00   Kurtosis:                         5.16
===================================================================================
```

Figure: Results for ARIMA Model(1,1,1)

ARIMA Model Results

	timestamp	h	prediction	actual
1	8/10/15	t+1	17370.500507	17615.16992
2	8/11/15	t+1	17685.151502	17402.83984
3	8/12/15	t+1	17361.905542	17402.50977
4	8/13/15	t+1	17412.633924	17408.25000
5	8/14/15	t+1	17401.151941	17477.40039

Figure: Predictions from ARIMA

ARIMA Model Results

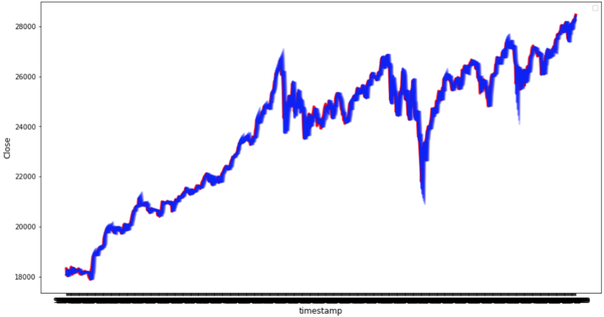


Figure: Plot of Actual and Predicted Values

RMSE=274.9319, MAE=182.287

LSTM Model Results

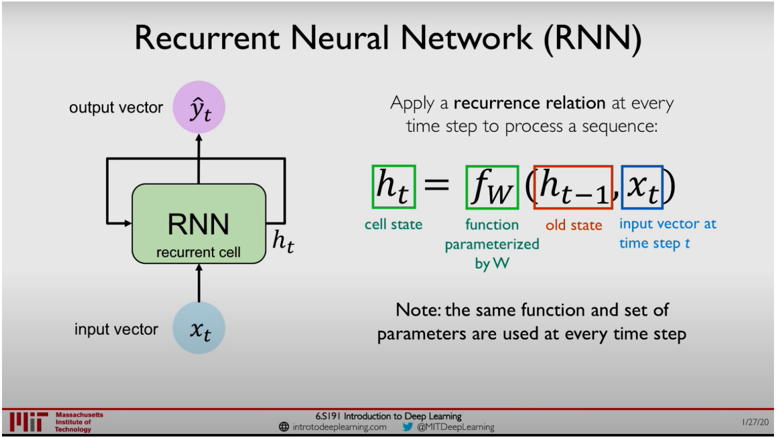


Figure: Simple RNN diagram (Courtesy of MIT) [4]

LSTM Model Results

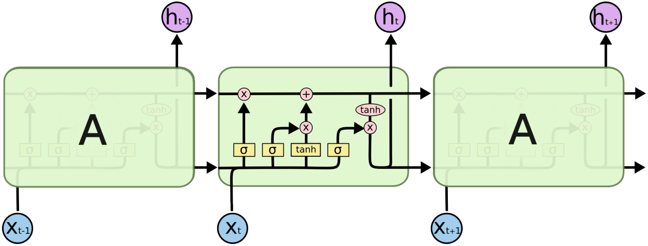


Figure: LSTM architecture [3]

Input: $\log \frac{Close_t}{Close_{t-1}}, \log \frac{Close_{t+1}}{Close_t}, \log \frac{Close_{t+2}}{Close_{t+1}}, \log \frac{Close_{t+3}}{Close_{t+2}} \ldots$

Output: $\log \frac{Close_{t+33}}{Close_{t+28}}$

LSTM Model Results (Training)

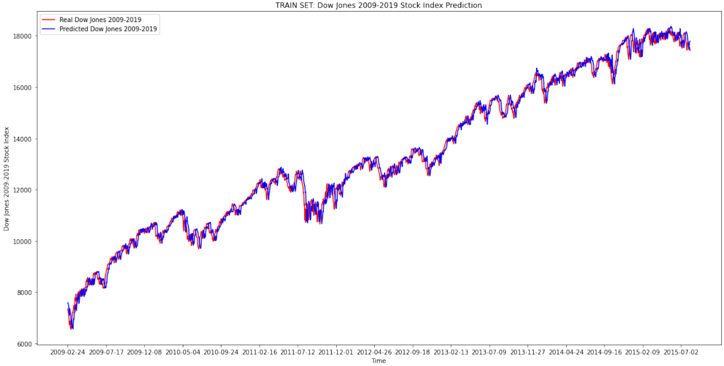


Figure: LSTM Model on Training Set

RMSE = 263.2288, MAE = 208.7557

LSTM Model Results (Testing)

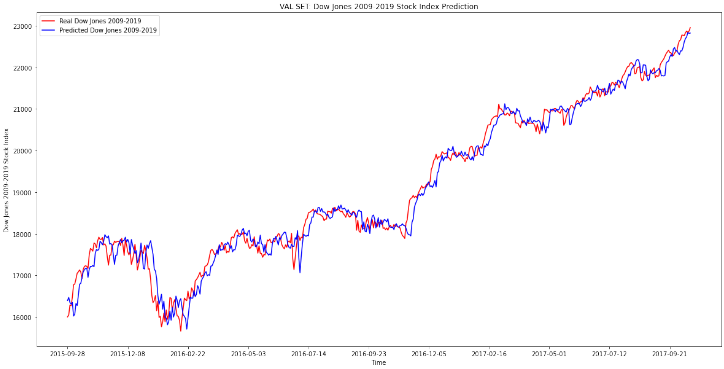
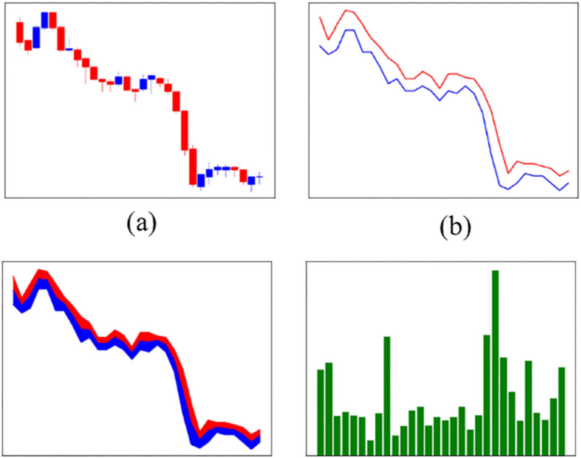


Figure: LSTM Model on Test Set

RMSE = 296.7456, MAE = 228.3258

CNN Model Results



(a) (b) (c) (d)

CNN Model Results (Training)

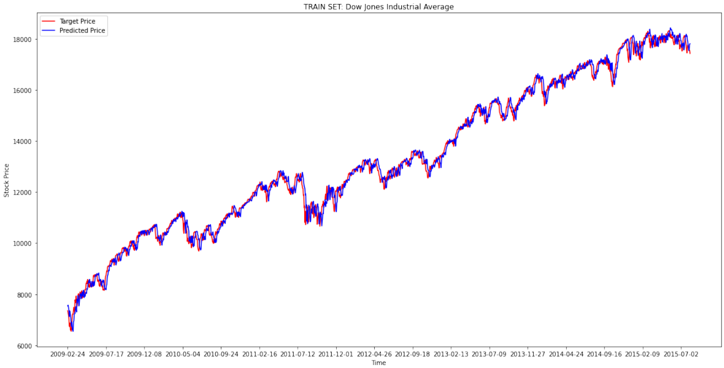


Figure: CNN Model on Training Set

RMSE = 250.6945, MAE = 191.6990

CNN Model Results (Testing)

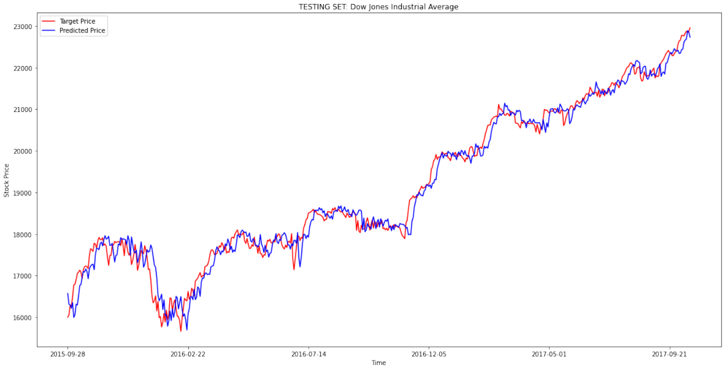
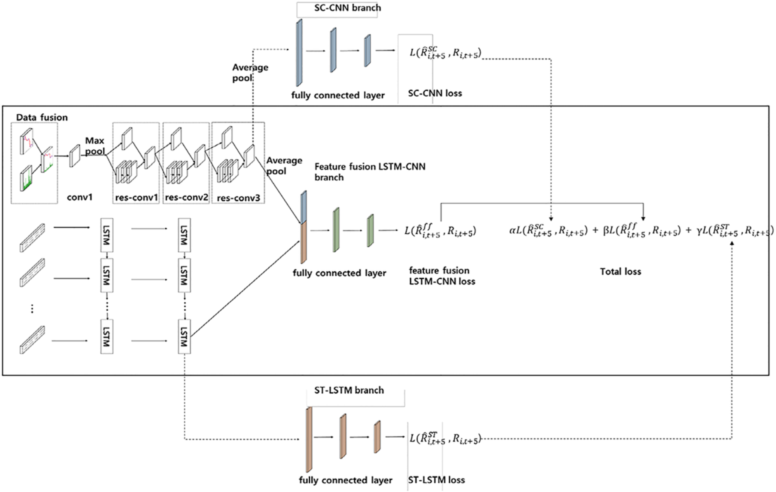


Figure: CNN Model on Test Set

RMSE = 270.6161, MAE = 201.3691

LSTM-CNN Model Results



LSTM-CNN Model Results (Training)

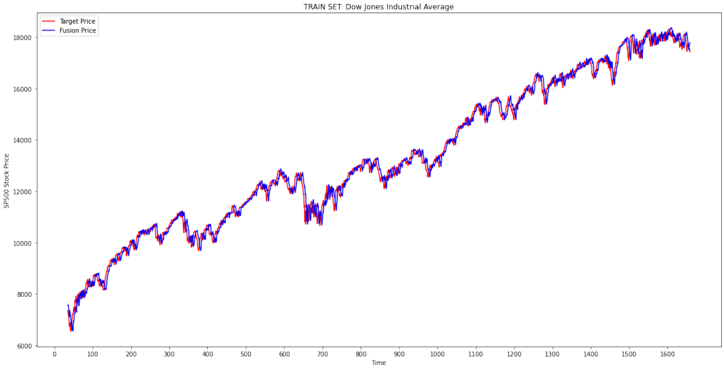


Figure: LSTM-CNN Model on Training Set

RMSE = 251.2169, MAE = 191.4447

LSTM-CNN Model Results (Testing)

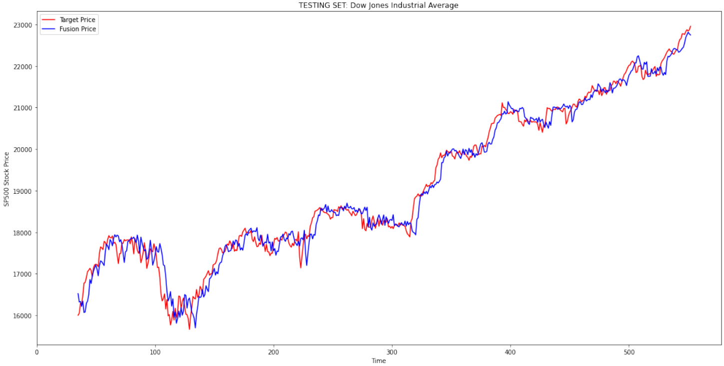


Figure: LSTM-CNN Model on Test Set

RMSE = 267.5648, MAE = 198.2916

Model Comparison

Table 1: Result on Dow Jones Industrial Average (DJI) (Test Set)

	RMSE	MAE
ARIMA	274.9319	182.2872
LSTM	296.7456	228.3258
CNN	270.6161	201.3691
LSTM-CNN	269.0664	198.8619

Figure: Model Comparison

Future Work

- Implement sentiment analysis to extract relevant stock news.
- Implement Generative Adversarial Network (GAN) with LSTM.
- Use Deep Reinforcement Learning (DRL) for deciding GAN's hyper-parameters.

Reference

H.Q.Thang. *Vietnam Stock Index Trend Prediction using Gaussian Process Regression and Autoregressive Moving Average Model*. Research and Development on Information and Communication Technology, HUST, 2018.

Kim T, Kim HY. *Forecasting stock prices with a feature fusion LSTM-CNN model using different representations of the same data*. PLoS ONE 14(2): e0212320. `https://doi.org/10.1371/journal.pone.0212320`, 2019.

Hao Y, Gao Q. *Predicting the Trend of Stock Market Index Using the Hybrid Neural Network Based on Multiple Time Scale Machine Learning*. MDPI Appl. Sci. 2020, 10(11), 3961. `https://doi.org/10.3390/app10113961`, 2020.

- CS231n. *Convolutional Neural Networks (CNNs / ConvNets)*. https://cs231n.github.io/convolutional-networks/.
- Aston Zhang, Zachary C. Lipton. *Dive into Deep Learning*.
- *Understanding LSTM Network*. https://colah.github.io/posts/2015-08-Understanding-LSTMs/, 2015.
- *Recurrent Neural Network*. MIT Deep Learning Bootcamp 6.S191. http://introtodeeplearning.com/, 2020.