주가 데이터분석을 위한 머신러닝 리포트 *

서정호

July 2, 2023

1 정의

1.1 사용된 Dataset

- Apple (AAPL) Historical Stock Data: 이 데이터셋은 2010년부터 2020년까지의 애플 (AAPL) 주식 데이터를 담고 있습니다.
- Tesla(TSLA) Tesla Stock Data : 이 데이터셋은 2010년부터 2023년까지의 최근 주식 데이터를 담고 있습니다.

1.2 RMSE (Root Mean Squared Error) 정의

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(y_{\text{pred }i} - y_{\text{actual }i}\right)^2}$$

1. RMSE는 예측값과 실제값 간의 차이를 제공하며

1.3 MAPE (Mean Absolute Percentage Error %) 정의

$$\text{MAPE} = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{y_{\text{actual }i} - y_{\text{pred }i}}{y_{\text{actual }i}}\right| \times 100$$

1. MAPE(%)는 이 차이를 실제값에 대한 상대적인 비율로 측정

2. MAPE 값이 12%인 경우, 예측된 주식 가격과 실제 주식 가격 간의 평균 차이가 12%라는 것을 나타냅니다.

1.4 주식 가격 예측을 위한 이동평균(Moving Average, MA)

$$\text{SMA} = \frac{\text{Sum of Data Points}}{\text{Number of Data Points}}$$

1. 일반적으로 짧은 기간, 중간 기간 및 장기 투자에 대해 각각 20일, 50일 및 200일 MA가 사용됩니다.

*Funding information or credit goes here.

1.5 EMA

$$\text{EMA} = (\text{Current Data Point} \times \text{Smoothing Factor}) + (\text{Previous EMA} \times (1 - \text{Smoothing Factor}))$$

1.6 LSTM

$$\text{Forget Gate: } f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f)$$

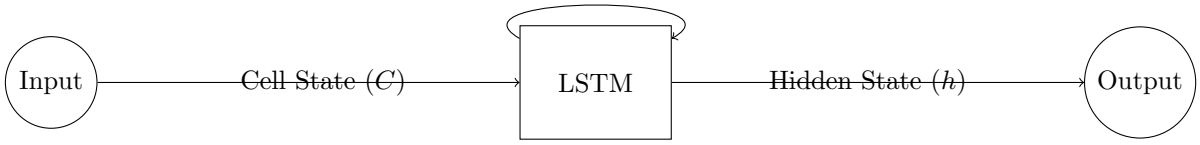
$$\text{Input Gate: } i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i)$$

$$\text{Output Gate: } o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o)$$

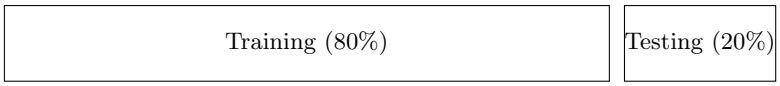
$$\text{Cell Gate: } \tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C)$$

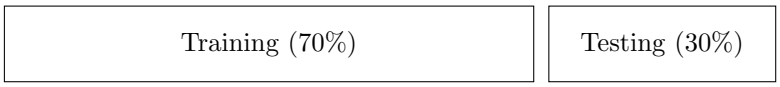
$$\text{Cell State: } C_t = f_t \cdot C_{t-1} + i_t \cdot \tilde{C}_t$$

$$\text{Output: } h_t = o_t \cdot \tanh(C_t)$$



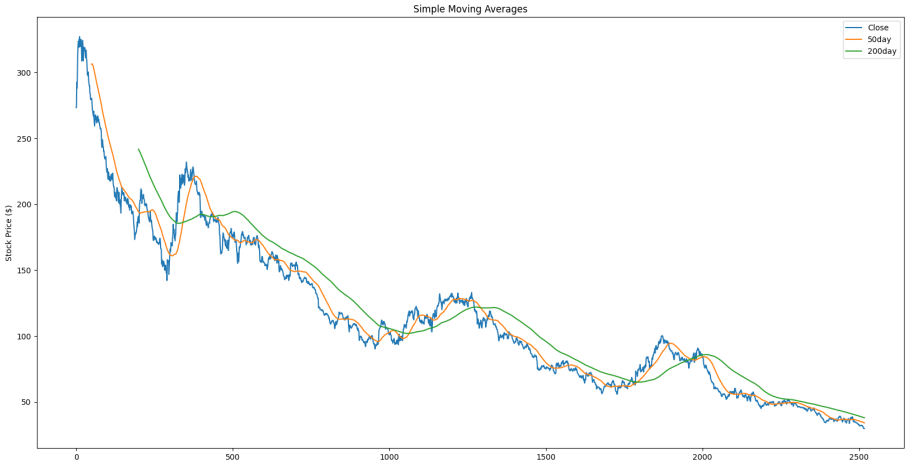
1.6.1 Scaling





2 Moving Average vs Exponential Moving Average For AAPL Stock Data

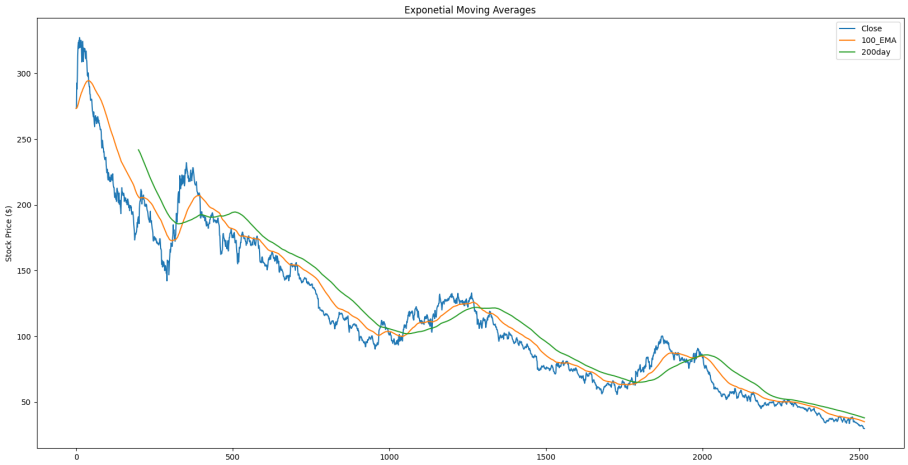
2.1 Moving Average For AAPL Stock Data



2.2 Exponential Moving Average For AAPL Stock Data

Data	Data Partition	RMSE_SMA	MAPE_SMA (%)
AAPL	2014 : 503	4.42	6.25

Table 1: RMSE and MAPE Results for SMA on AAPL Data

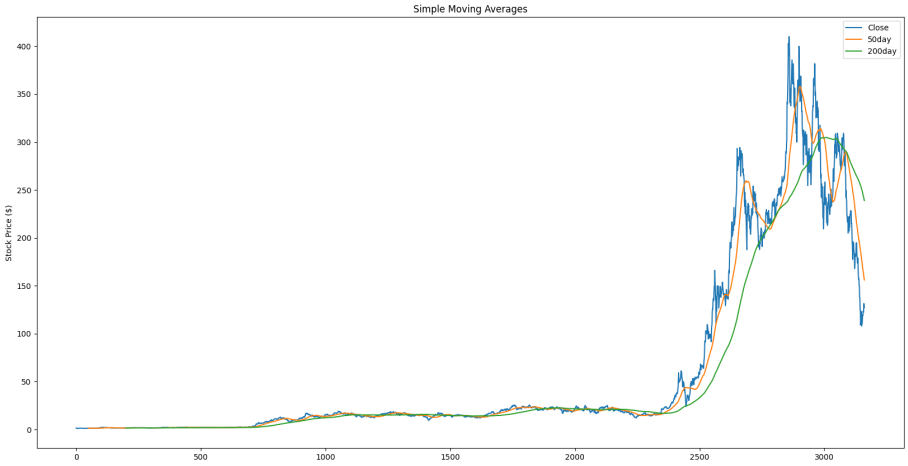


Data	Data Partition	RMSE_SMA	MAPE_SMA (%)
AAPL	2014 : 503	3.86	5.65

Table 2: RMSE and MAPE Results for EMA on AAPL Data

3 Moving Average vs Exponential Moving Average For TSLA Stock Data

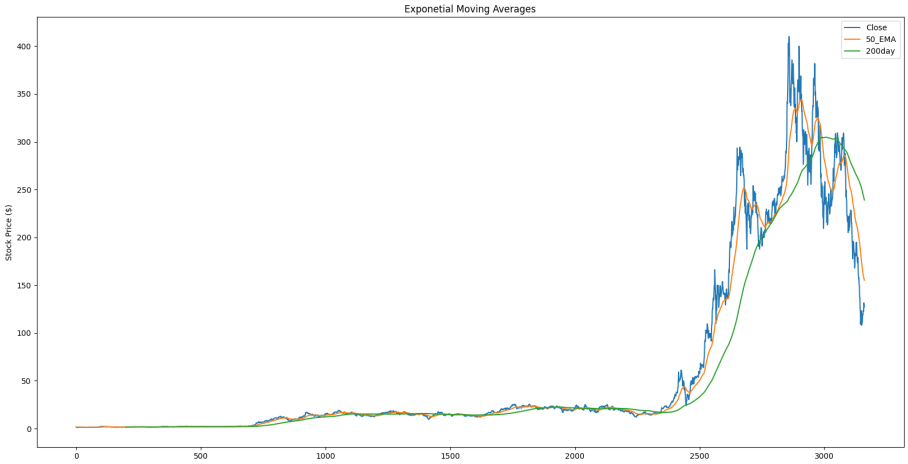
3.1 Moving Average For TSLA Data



Data	Data Partition	RMSE_SMA	MAPE_SMA (%)
TSLA	2529 : 632	39.92	14.69

Table 3: RMSE and MAPE Results for SMA on TSLA Data

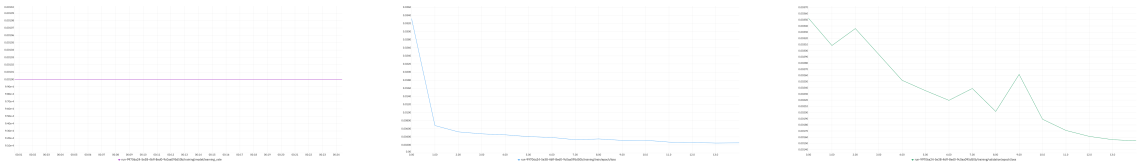
3.2 Exponential Moving Average For TSLA Data



Data	Data Partition	RMSE_EMA	MAPE_EMA (%)
TSLA	2529 : 632	33.78	12.42

Table 4: RMSE and MAPE Results for EMA on TSLA Data

3.3 Optimal Performance of LSTM Model for AAPL Data: Achieving Perfect Predictive Results



(a) Learning rate analysis for AAPL stock model training

(b) Training set epoch loss for AAPL stock

(c) Validation set epoch loss for AAPL stock

Figure 1: Performance Analysis on AAPL Data Training

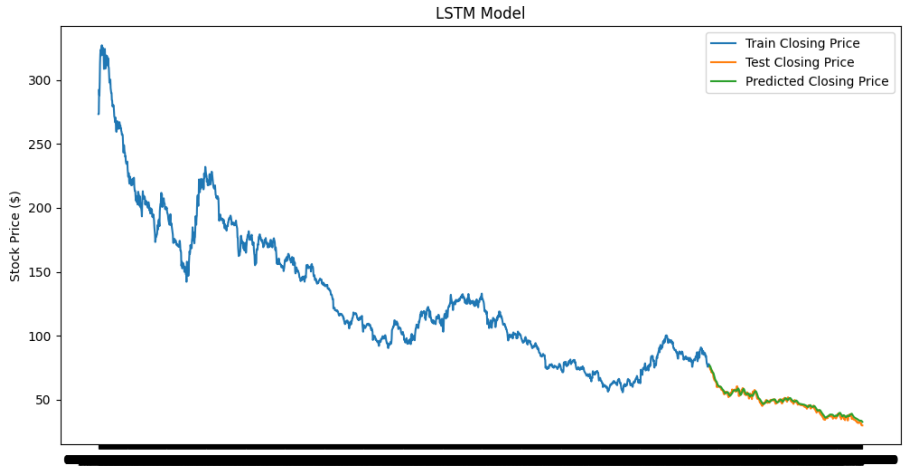
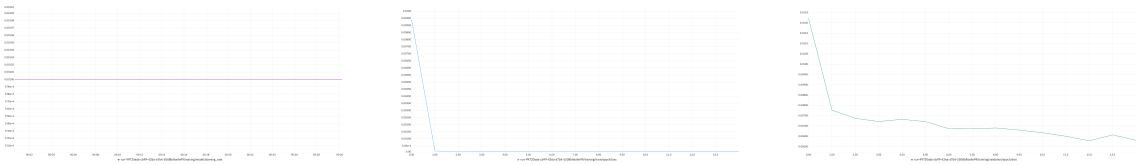


Figure 2: Evaluating the Performance of the LSTM Model with 15 Epochs and 15 Batch Size For AAPL Data

Data	Parameters	Data Partition	RMSE	MAPE (%)
AAPL	{'epochs': 15, 'batch_size': 15}	2014 : 503	1.89	3.63

3.4 Untuned LSTM Model Parameters and Performance Metrics for TSLA Stock Prediction.



(a) Learning rate analysis for TSLA stock model training

(b) Training set epoch loss for TSLA stock

(c) Validation set epoch loss for TSLA stock

Figure 3: Performance Analysis of TSLA Stock Training without tuned hyperparameter

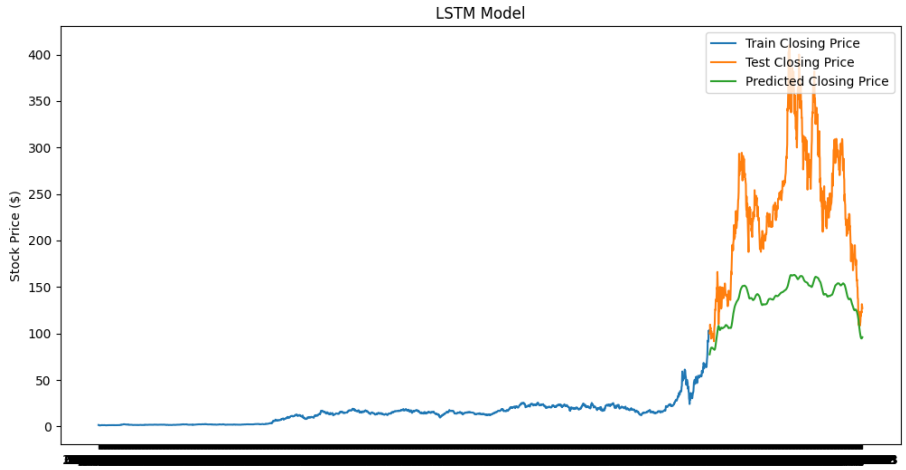


Figure 4: Evaluating the Performance of the LSTM Model with 15 Epochs and 15 Batch Size.

Data	Parameters	Data Partition	RMSE	MAPE (%)
TSLA	{'epochs': 15, 'batch_size': 15}	2529 : 632	113.47	39.26

3.5 Algorithmic explanation for hyperparameter tuning of an LSTM model

Algorithm 1 Hyperparameter Tuning for LSTM

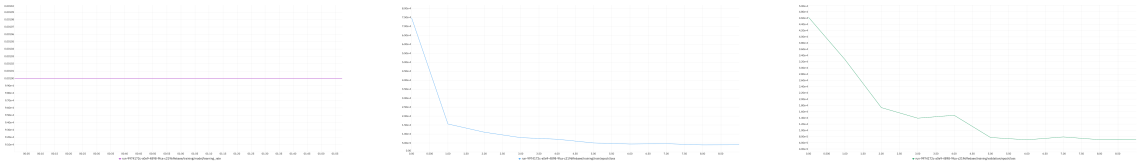
```
procedure HyperparameterTuning(Hyperparameters)
    Initialize best_rmse ← ∞, best_mape ← ∞, best_params ← {}
    for epochs in hyperparameters['epochs'] do
        for batch_size in hyperparameters['batch_size'] do
            Train and evaluate LSTM model
            rmse_lstm, mape_lstm ← train_and_evaluate_lstm(
            stockprices=df,
            train_size=train_size,
            window_size=window_size,
            cur_epochs=epochs,
            cur_batch_size=batch_size,
            test=test,
            run=run )
            if rmse_lstm < best_rmse then
                best_rmse ← rmse_lstm
                best_mape ← mape_lstm
                best_params ← {'epochs': epochs, 'batch_size': batch_size}
            end if
        end for
    end for
    Print Best Parameters: best_params
    Print Best RMSE: best_rmse
    Print Best MAPE: best_mape
end procedure
```

3.6 Console-Log

```
2023/06/21 15:36:06   1771/1946 [=========================>...] - ETA: 1s - loss: 3.5005e-05
2023/06/21 15:36:07   1779/1946 [=========================>...] - ETA: 1s - loss: 3.5006e-05
2023/06/21 15:36:07   1787/1946 [=========================>...] - ETA: 1s - loss: 3.5209e-05
2023/06/21 15:36:07   1795/1946 [=========================>...] - ETA: 0s - loss: 3.5332e-05
2023/06/21 15:36:07   1803/1946 [=========================>...] - ETA: 0s - loss: 3.5255e-05
2023/06/21 15:36:07   1811/1946 [=========================>...] - ETA: 0s - loss: 3.5293e-05
2023/06/21 15:36:07   1819/1946 [=========================>..] - ETA: 0s - loss: 3.5189e-05
2023/06/21 15:36:07   1827/1946 [=========================>..] - ETA: 0s - loss: 3.5343e-05
2023/06/21 15:36:07   1835/1946 [=========================>..] - ETA: 0s - loss: 3.5198e-05
2023/06/21 15:36:07   1843/1946 [=========================>..] - ETA: 0s - loss: 3.5292e-05
2023/06/21 15:36:07   1851/1946 [=========================>..] - ETA: 0s - loss: 3.5357e-05
2023/06/21 15:36:07   1859/1946 [=========================>..] - ETA: 0s - loss: 3.5613e-05
2023/06/21 15:36:07   1867/1946 [=========================>..] - ETA: 0s - loss: 3.5894e-05
2023/06/21 15:36:07   1875/1946 [=========================>..] - ETA: 0s - loss: 3.6100e-05
2023/06/21 15:36:07   1883/1946 [=========================>.] - ETA: 0s - loss: 3.6172e-05
2023/06/21 15:36:07   1891/1946 [=========================>.] - ETA: 0s - loss: 3.6308e-05
2023/06/21 15:36:07   1899/1946 [=========================>.] - ETA: 0s - loss: 3.6284e-05
2023/06/21 15:36:07   1907/1946 [=========================>.] - ETA: 0s - loss: 3.6147e-05
2023/06/21 15:36:07   1915/1946 [=========================>.] - ETA: 0s - loss: 3.6159e-05
2023/06/21 15:36:07   1923/1946 [=========================>.] - ETA: 0s - loss: 3.6227e-05
2023/06/21 15:36:07   1931/1946 [=========================>.] - ETA: 0s - loss: 3.6301e-05
2023/06/21 15:36:08   1939/1946 [=========================>.] - ETA: 0s - loss: 3.6488e-05
2023/06/21 15:36:08   1946/1946 [==============================] - 13s 7ms/step - loss: 3.6492e-05 - val_loss: 9.9261e-05
2023/06/21 15:36:08  1/30 [>.............................] - ETA: 7s
2023/06/21 15:36:08   14/30 [=============>................] - ETA: 0s
2023/06/21 15:36:08   26/30 [=========================>....] - ETA: 0s
2023/06/21 15:36:08   30/30 [==============================] - 0s 4ms/step
2023/06/21 15:36:08  RMSE LSTM: 61.39451184617583
2023/06/21 15:36:08
2023/06/21 15:36:08  MAPE LSTM: 18.428115632957827
2023/06/21 15:36:08
2023/06/21 15:36:13  Shutting down background jobs, please wait a moment...
```

Figure 5: Displaying the Standard Output of the LSTM Model with running 10 Epochs and a Batch Size of 1.

3.7 Execution Results of LSTM Model with 10 Epochs and Batch Size of 1



(a) Learning rate analysis for TSLA stock model training with tuned hyperparameters

(b) Training set epoch loss for TSLA stock with tuned hyperparameter

(c) Validation set epoch loss for TSLA stock with tuned hyperparameter

Figure 6: Performance Analysis of TSLA Stock Training with tuned hyperparameter

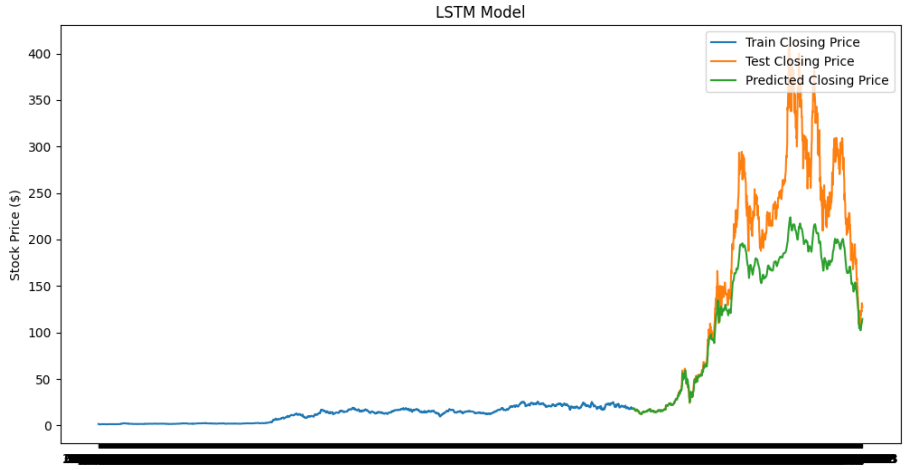


Figure 7: Executing the Performance of LSTM Model with 10 Epochs and 1 Batch Size

Trial	Parameters	Data Partition	RMSE	MAPE (%)
5	{'epochs': 10, 'batch_size': 1}	2213 : 948	61.00	18.46

3.8 결과 요약

Figure 8: Displaying the Sequential Process Plots of Tuning Results

Trials	Parameters	Data Partition	RMSE	MAPE (%)
STOC-94	{'epochs': 15, 'batch_size': 15}	2529 : 632	113.47	39.26
STOC-95	{'epochs': 15, 'batch_size': 15}	2213 : 948	92.68	28.96
STOC-96	{'epochs': 10, 'batch_size': 10}	2213 : 948	86.59	26.98
STOC-97	{'epochs': 9, 'batch_size': 1}	2213 : 948	74.61	21.74
STOC-98	{'epochs': 10, 'batch_size': 1}	2213 : 948	61.00	18.46
STOC-99	{'epochs': 7, 'batch_size': 1}	2213 : 948	54.21	15.27
STOC-100	{'epochs': 8, 'batch_size': 1}	2213 : 948	52.96	14.74
STOC-101	{'epochs': 5, 'batch_size': 1}	2213 : 948	47.83	13.52
STOC-102	{'epochs': 6, 'batch_size': 1}	2213 : 948	44.79	12.34
STOC-103	{'epochs': 7, 'batch_size': 1}	2213 : 948	42.61	11.29
STOC-104	{'epochs': 8, 'batch_size': 1}	2213 : 948	40.96	10.37
STOC-105	{'epochs': 9, 'batch_size': 1}	2213 : 948	38.74	9.57

Table 5: Summary of Hyperparameter Tuning Results

3.9 Conclusion

- Table 1와 2를 참고하면, AAPL의 SMA RMSE 값은 4.42이고, MAPE 값은 6.25입니다. EMA RMSE 값은 3.86이며 MAPE 값은 5.65입니다. SMA 모델에 비해 EMA 모델이 더 정확하고 정밀한 결과를 보여주었으며, 각각 12.67%와 9.6%의 감소를 보였습니다.
- Table 3와 4를 참고하면, TSLA의 SMA RMSE 값은 39.92이고, MAPE 값은 14.69입니다. EMA RMSE 값은 33.78이며 MAPE 값은 12.42입니다. 마찬가지로, MA 모델에 비해 EMA 모델이 더 낮은 오차율과 RMSE 결과를 보여주었으며, 각각 15.37%와 15.43%의 감소를 보였습니다.
- Section 3.3을 참고하면, LSTM 모델은 다른 두 가지 방법에 비해 뛰어난 예측 능력을 보여준 것을 확인할 수 있습니다. RMSE와 MAPE 값은 각각 1.89와 3.64%입니다. Figure 2 그래프를 보면 Testing 값과 Prediction이 거의 일치함을 확인할 수 있습니다. MAPE 백분율의 큰 감소는 LSTM 모델이 MA 및 EMA 모델에 비해 달성한 정확성과 정밀성의 더 높은 수준을 나타내는걸 확인할 수 있습니다.
- Section 3.7 결과를 보면, 앞서 사용된 AAPL LSTM 모델과는 달리 높은 RMSE와 MAPE 값을 나타냈습니다. 또한, 해당 수치를 시각화한 그래프도 Testing 값과 Prediction 값 사이에 큰 오차를 보여주고 있습니다

- Section 3.5의 Algorithm 1을 보면, Hyperparameter tuning에 필요한 요소들이 담겨 있습니다. 해당 알고리즘은 cur_batch_size와 cur_epochs 배열 값을 받아 하나씩 실행하여 brute-force 방식으로 hyperparameter tuning을 수행합니다.
- Section 3.7에서 하이퍼파라미터 튜닝을 통해 LSTM 모델의 성능이 상당히 향상되었음을 확인할 수 있습니다. MSE로 측정된 모델의 성능은 초기 값과 비교하여 46.27%라는 현저한 감소를 보였습니다. 이러한 감소는 예측된 종가 값이 실제 테스트 종가 값과 더욱 정확하게 일치하도록 개선된 것을 의미합니다. 또한, MAPE 값 역시 52.91%의 인상적인 개선을 보였습니다. Figure 6를 참고한 결과.
- 추가적인 하이퍼파라미터 조정을 통해 RMSE가 38.74이고 MAPE가 9.57%로 눈에 띄게 감소했습니다. Table 5의 지표를 확인하면 RMSE 값과 MAPE의 요약 결과를 확인할 수 있습니다. 이는 기존 값과 비교하여 75.61%의 성능 향상을 보여주어 좋은 결과를 얻을 수 있었습니다.

3.10 Evaluation

- 최적화를 진행하는 중간에 비교적 낮은 에포크와 배치 크기 매개 변수가 눈에 띕니다. 이는 그들이 모델의 성능에 미치는 영향을 더 조사하고 탐색할 필요성을 제안합니다.
- 이 영역에서의 지식이나 전문성을 바탕으로 추가적인 통찰력이나 발견을 포함하는 것은 분석을 더욱 향상시킬 것입니다.
- 전반적으로, LSTM 모델은 예측 능력에서 우월성을 보여주며, 추가적인 개선 가능성이 있을 수는 있지만, 이 단계에서 실험은 종료되었습니다.
- 그러나, 낮은 배치 크기와 낮은 에포크 수 선택은 모델의 성능에 영향을 미칠 수 있다는 점을 주목해야 합니다. 낮은 배치 크기는 모델이 각 반복에서 데이터의 작은 부분 집합으로 훈련되는 것을 의미하며, 이는 기울기 추정에서의 노이즈 증가와 수렴 속도 감소를 초래할 수 있습니다.
- 마찬가지로, 낮은 에포크 수는 모델이 짧은 시간 동안 훈련되는 것을 의미하므로, 데이터의 복잡한 패턴과 관계를 포착하는 능력을 제한할 수 있습니다.
- 이러한 요소들은 모델 성능의 부족과 예측 정확도의 저하를 초래할 수 있습니다.