alamSYS: Development of Stock Market Price Forecasting System Using Dynamic Mode Decomposition, Long Short-Term Memory with Arnaud Legoux Moving Average Convergence-Divergence Integration

A Special Problem
Presented to
the Faculty of the Division of Physical Sciences and Mathematics
College of Arts and Sciences
University of the Philippines Visayas
Miag-ao, Iloilo

In Partial Fulfillment
of the Requirements for the Degree of
Bachelor of Science in Computer Science by

OLARTE, John Markton M.

Nilo C. Araneta
Adviser

June 2023

Contents

List of Figures

List of Tables

Chapter 1

Review of Related Works and Literature

One of the challenges facing investors in the Philippine Stock Market is the limited availability of resources and tools for making market decisions. In contrast, other countries have begun implementing machine learning techniques for stock market prediction and analysis, which allows for more accurate decision-making and reduces the risk of poor investment outcomes. As a result, these countries are likely to experience better returns on their investments.

In this literature review, the following general topics are reviewed, discussed, and synthesized: (a) Integration of Machine Learning based Trading Algorithms; and (b) Utilization of Dynamic Mode Decomposition on the Financial markets.

1.1 Integration of Machine Learning based Trading Algorithms

Stock market analysis is crucial for effective risk management. This involves using various methods, such as technical and fundamental analysis, to make informed decisions for investors and traders. In recent years, the growth of computing power and resources has led to the increasing use of machine learning techniques for stock market prediction and analysis. These advances help companies better predict upcoming market trends and make more informed decisions.

The integration of machine learning algorithms in the stock market is growing, as investors and traders increasingly rely on fast and accurate market information to reduce potential risks and make better decisions. These algorithms allow for more efficient analysis of market data, leading to more informed decisions and improved investment outcomes (Obthong, Tantisantiwong, Jeamwatthanachai, & Wills, 2020).

1.1.1 Comparison of Machine Learning and Deep Learning Models in Stock Market Predictions

To have a better grasp in the accuracy of the different models used in algorithmic trading it is essential that different models are compared against each other.

Combination of Computational Efficient Functional Link Artificial Neural Network (CEFLANN) and Traditional Technical Analysis

This hybrid model combines a classification-based model: CEFLANN and the traditional technical analysis to create a stock trading framework Dash and Dash (2016), which the results show a profit of 24.29%.

Deep Long Short-Term Neural Network (LSTM) with Embedded Layer

In one of the models developed by Pang, Zhou, Wang, Lin, and Chang (2020), it shows that by adding an embedded layer to the LSTM it yields to a stock market price prediction accuracy of 57.2%. However, its accuracy dips to 52.4% when the model is applied to individual stocks.

LSTM with Automatic Encoder

As part of the second model developed by Pang et al. (2020), this model shows a slightly inaccurate stock market prediction, by only having a measured accuracy of 56.9%. However, compared to the first model developed by the group this is 0.1% more effective for individual stocks.

Optimal Deep Learning (ODL)

In the study conducted by Agrawal, Khan, and Shukla (2019) they have created a stock price prediction model using an Optimal Deep Learning (ODL) which combine the concepts of Correlation-Tensor and an Optimal LSTM algorithm. Whereas their results show a mean and highest accuracy of the model as 59.24% and 65.64%.

NMC-BERT-LSTM-DQN-X Algorithm

More recently, a team have applied a combination of three models for forecasting the market trends. Namely, (1) Non-stationary Markov Chain (NMC), (2) Bidirectional Encoder Representations from Transformers (BERT), (3) Long Short-Term Memory (LSTM). Wherein their model shows an accuracy of 61.77%. Furthermore, the team also mentioned that the model produces 29.25% annual return on investment, with a maximum losses rating of -8.29% (Liu, Yan, Guo, & Guo, 2022).

1.2 Utilization of Dynamic Mode Decomposition (DMD) on the Financial Markets

Dynamic Mode Decomposition (DMD) as an emerging data-driven technique which allows spatial-temporal pattern recognition from a complex set of data and was first introduced in the field of fluid mechanics by (SCHMID, 2010).

1.2.1 Chronological Utilization of DMD in the Financial Markets

In (2015) Mann and Kutz proved that DMD can be used as data-driven analytics on the financial market data. Wherein, DMD allows a predictive assessment of the market dynamics, which helps in the capitalization of stock market strategies and decisions to be applied.

Utilization of DMD for Determining the Cyclic Behavior in the Stock Market (2016)

By utilizing the reproducible Koopman modes it made it possible to have extracted four cyclic variations (also reproducible modes) in the stock market, which were previously unknown and have persisted since the 1870s' global economic crisis (Hua, Roy, McCauley, & Gunaratne, 2016; Williamson, 2015).

Utilization of DMD as part of an Algorithmic Trading Strategies for the Turkish Stock Market (2015 and 2017)

The study of Mann and Kutz (2015) in the utilization of DMD for financial stock market prediction has become the foundation of the study by Savaș (2017) on the algorithmic trading strategies with Dynamic Mode Decomposition for the Turkish Stock Market. Wherein, based on their results they found out that the timing of DMD analysis was not significantly accurate, as such they have used a simple moving average with genetic algorithm to improve the market timing of DMD, which prevents 80% of the false trade signals.

Furthermore, this also shows that DMD is an effective alpha model that is easy to implement and use for any algorithmic trading strategy, and the addition of technical analysis tools can further improve its capabilities, especially on the predictive temporal side of the data.

Utilization of DMD-based Trading Strategy in the Chinese Stock Market (2016)

In the study by Cui and Long (2016), they have found that DMD was able to capture the dynamic patterns of the Chinese Stock Market, especially in a sideway trending market.

Their study also shows that the predictive ability of DMD can effectively model the behavior of the Chinese Stock Market, even if there are no clear trends that can be observed.

Utilization of Adaptive Elastic DMD to Improve Momentum Strategies (2021)

A study by Uchiyama and Nakagawa (2021), using Adaptive Elastic Dynamic Mode Decomposition (AEDMD) shows that they were able to estimate the market trend, and were able to demonstrate that the approach is better than existing momentum strategy which are only based on simple past trends.

1.3 Synthesis

Fast and accurate market information is an essential tool for stock market participants. In recent years, the development of machine learning models for the financial markets, such as stocks, has proven to be increasingly effective in predicting future stock prices and trends. The use of Dynamic Mode Decomposition (DMD) in the stock market has also been shown to be effective in predicting stock price trends. The simplicity and elegance of the Koopman Decomposition Operator make it an ideal basis for the development of a Stock Market Price Trend Forecasting System.

Hence, these studies are crucial for the development of the alamSYS. As it can provide investors with fast and accurate information about which stocks are likely to go up or down, allowing them to make more informed decisions about buying or selling those stocks.

In addition to the potential benefits for investors and traders, the implementation of machine learning techniques in the stock market can also help improve market efficiency and reduce the risk of market manipulation. By providing a more accurate and comprehensive view of market trends, these techniques can help ensure that prices reflect the true value of stocks and other assets, leading to more stable and fair market conditions.

Chapter 2

Results and Discussions

This chapter presents results and discussions from this special problem. Its goal is to provide a comprehensive analysis and interpretation of the data collected for alamSYS's internal and external components. As a result, this chapter is divided into the following sections:

(a) Documentation for alamSYS

(b) DMD-LSTM Results and Discussions

(c) ALMACD Results and Discussions

(d) alamSYS System Tests Results and Discussions

(e) Results and Discussions for the Real World Application of alamSYS

2.1 alamSYS Documentation

The goal of this section is to thoroughly document the current state of the alamSYS in order to facilitate meaningful discussions.

2.1.1 Documentation for alamAPI and Database

xxx

2.1.2 Documentation for alamSYS Preprocessor

xxx

2.1.3 Documentation for alamAPP

xxx

2.1.4 Build and Deployment Guide

xxx

2.2 DMD-LSTM Model Results and Discussions

This section presents and discusses the Deep Learning Model's training, testing, and cross-validation results.

In Table 2.1 the training error metrics are shown for each of the window sizes tested.

Table 2.1: DMD-LSTM Training Error Metrics Scores for Different Window Sizes

Error Metrics	*Window Sizes*			
	5	**10**	**15**	**20**
MSE	0.000037	0.787877	0.006917	0.057851
RMSE	0.006106	0.887624	0.083166	0.240522

Table 2.1 continued from previous page

Error Metrics	*Window Sizes*			
	5	**10**	**15**	**20**
MAE	0.004175	0.755407	0.067645	0.202746
MAPE	**0.000001**	0.000194	0.000017	0.000053

Where it is observed that the best performing model based on having the lowest MAPE score is the DMD-LSTM with a window size of 5. Morever, we can see the differences from each MAPE score for each window size in the Figure 2.1 shown below.

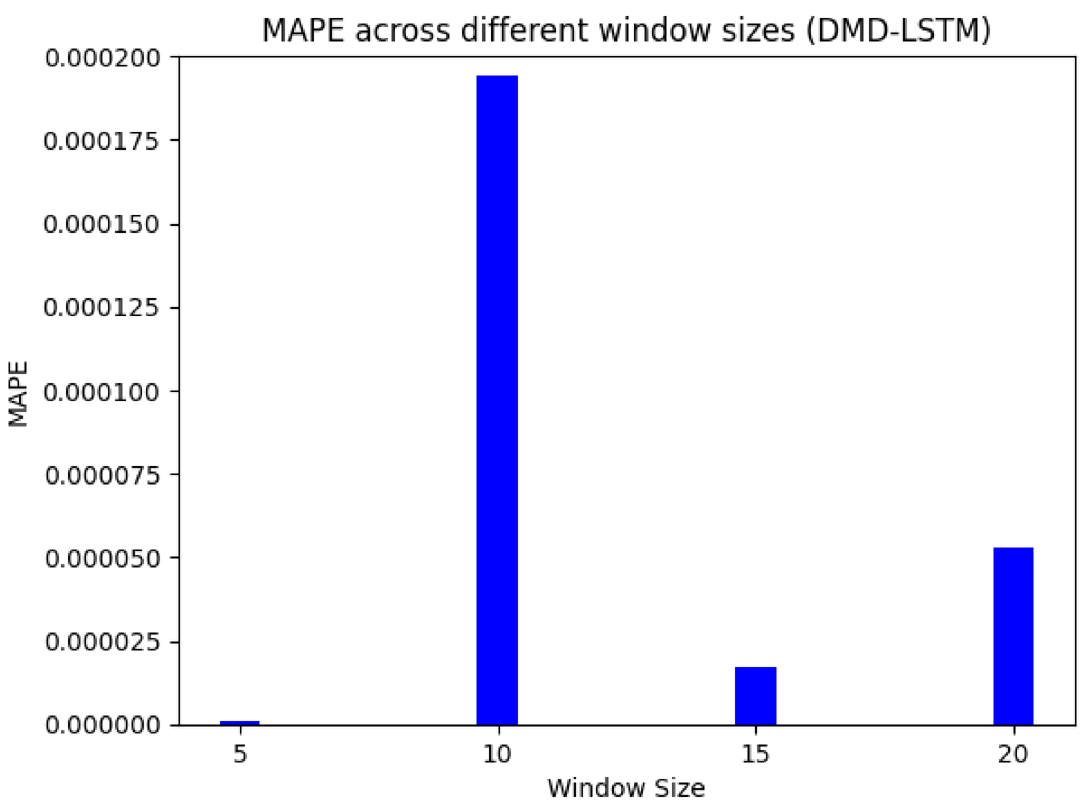

Figure 2.1: Comparison of MAPE Scores for DMD-LSTM Model Training Across Different Window Sizes

The figure above also shows that the MAPE score for window sizes 15 and 20 is higher than the MAPE score for window size 10. MAPE score increases from window size 15 to size 20, indicating that increasing window size may result in a lower performing model.

Furthermore, as previously stated, the window size of 5 results in the best MAPE score being the lowest. Where it outperforms the worst performing model (DMD-LSTM with window size 10) by 0.000193 units. As illustrated clearly in Figure 2.1.

Knowing that the DMD-LSTM model performs as expected based on the training data scores, it is critical that we also examine the training data results from a baseline LSTM. The baseline LSTM is, as the name implies, a simple LSTM model lacking the DMD component. The table below shows the results of the baseline LSTM training.

Table 2.2: Baseline LSTM Training Error Metrics Scores for Different Window Sizes

Error Metrics	*Window Sizes*			
	5	**10**	**15**	**20**
MSE	2912.840703	191.935882	1118.183283	706.136814
RMSE	53.970739	13.854093	33.439248	26.573235
MAE	35.301888	9.480864	22.099720	18.285352
MAPE	0.009618	**0.002527**	0.006024	0.005004

According to the table above, the baseline LSTM with window size 10 performs the best, with the lowest MAPE score of 0.002527 when compared to the other baseline LSTM models.

However, the DMD-LSTM model with window size 5 outperforms it by 0.002526. As a result, the alamSYS makes use of the DMD-LSTM model, specifically the one with a window size of 5. Where from now on, the DMD-LSTM model refers to the DMD-LSTM model with a window size of 5.

Nonetheless, the DMD-LSTM model's performance is limited to the training dataset from PSEI, and it must be cross-validated using data from other stocks, which includes the PSEI validation dataset. The results of this cross-validation is presented in Table 2.3. It should also be noted that cross-validation uses logarithmic normalization as a data preprocessing technique to make the dataset more normal, which aids in analyzing the model's performance with the given dataset. Normalization techniques, in particular, allow for closer variation within the forecasted data. (S.Gopal Krishna Patro, 2015).

Table 2.3: DMD-LSTM Cross-Validation Error Metrics Scores

Stocks	MSE	RMSE	MAE	MAPE
PSEI	0.00002	0.00419	0.00328	1.510000e-03
AC	0.00236	0.04856	0.03414	6.110000e-03
ALI	0.00255	0.05054	0.03645	1.597000e-02
AP	0.00129	0.03596	0.02515	9.220000e-03
BDO	0.00160	0.03999	0.02799	7.250000e-03
BLOOM	0.01883	0.13721	0.06901	1.052898e+12
FGEN	0.00224	0.04733	0.03265	1.197000e-02
GLO	0.00211	0.04595	0.03149	4.680000e-03
ICT	0.00335	0.05785	0.03731	3.005818e+11
JGS	0.00331	0.05752	0.03992	2.009923e+11
LTG	0.01567	0.12518	0.05858	3.583335e+12
MEG	0.00431	0.06565	0.04422	1.393042e+11
MER	0.00326	0.05708	0.03770	9.170000e-03
MPI	0.00273	0.05230	0.03390	2.497000e-02
PGOLD	0.00149	0.03865	0.02818	7.880000e-03
RLC	0.00338	0.05817	0.03978	6.922000e-02
RRHI	0.00131	0.03618	0.02699	6.390000e-03
SMC	0.00137	0.03702	0.02317	5.690000e-03
TEL	0.00178	0.04214	0.03002	4.240000e-03
URC	0.00297	0.05447	0.03742	1.798000e-02

As shown in the table above, the chosen DMD-LSTM model performs well across all other stocks, demonstrating that the model is not overfitted to the training dataset. This score additionally suggests that the model works with non-training data.

The figures below show a 100-day worth of predicted prices versus actual prices to better visualize the performance of the DMD-LSTM model for each stock.

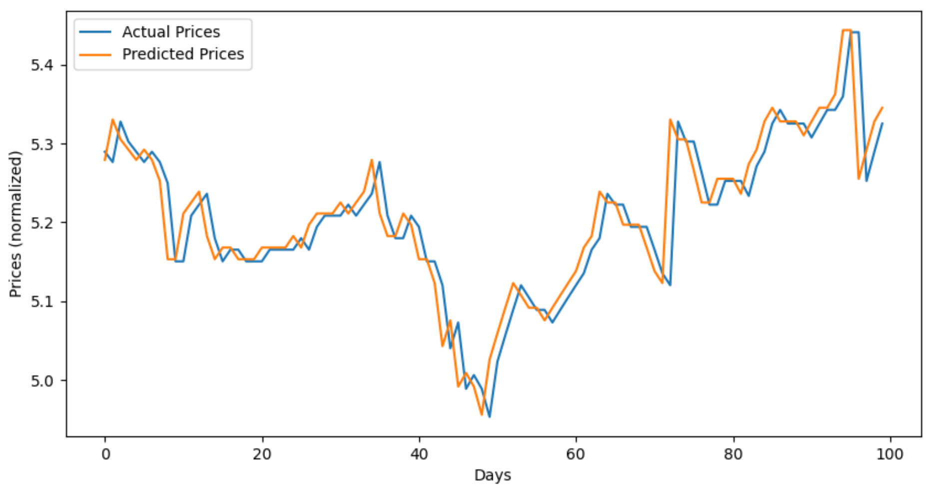

Figure 2.2: Actual vs Predicted Prices on AC for 100 days

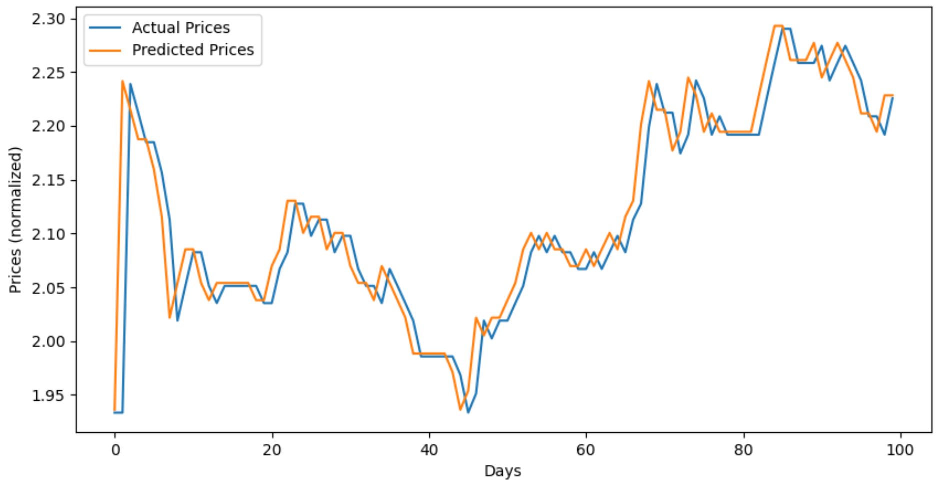

Figure 2.3: Actual vs Predicted Prices for ALI over 100 days

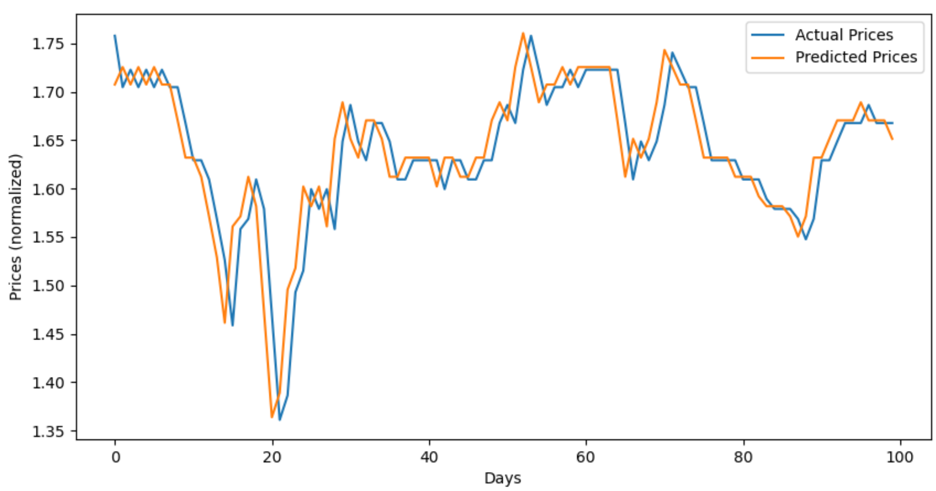

Figure 2.4: Actual vs Predicted Prices for AP over 100 days

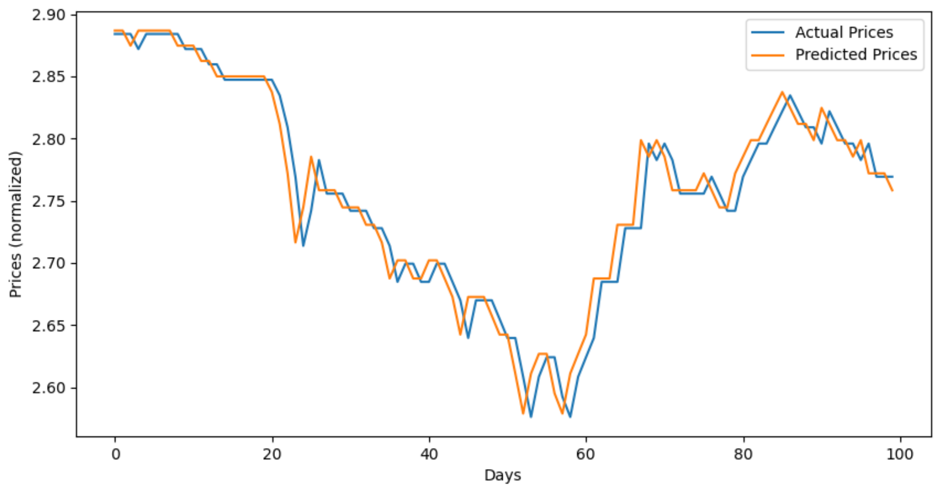

Figure 2.5: Actual vs Predicted Prices for BDO over 100 days

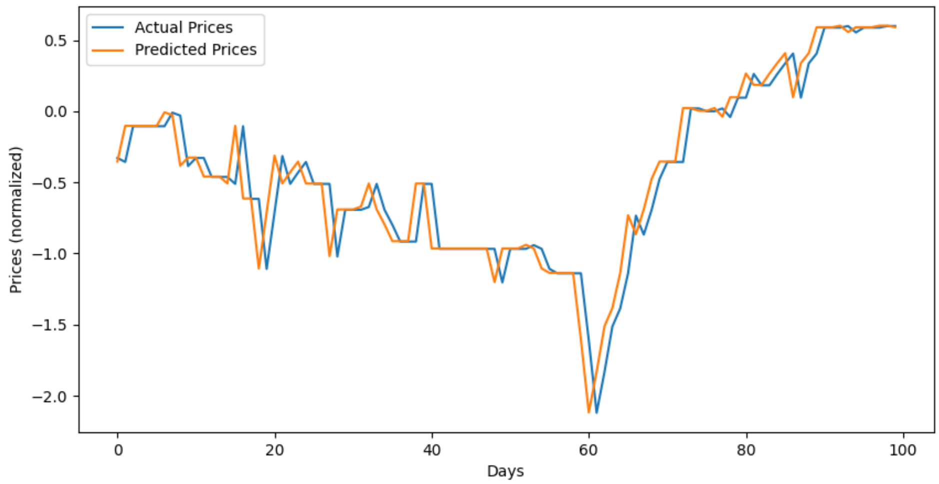

Figure 2.6: Actual vs Predicted Prices for BLOOM over 100 days

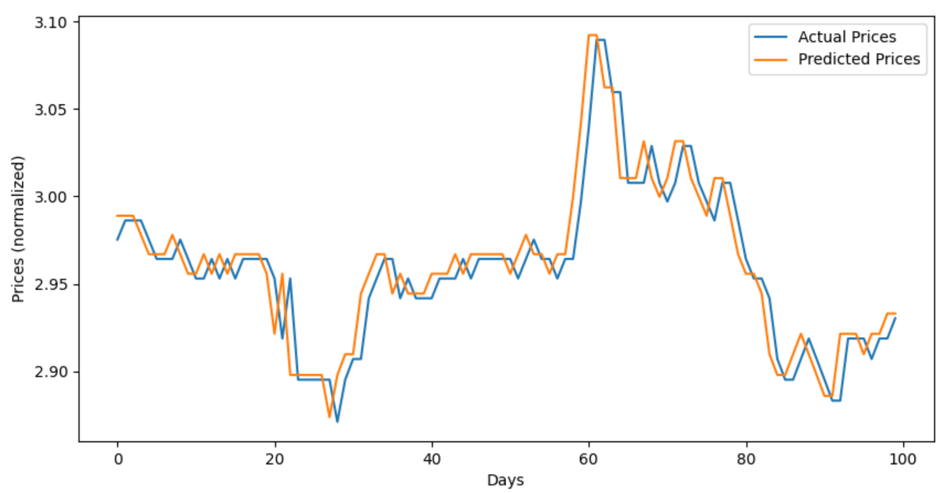

Figure 2.7: Actual vs Predicted Prices for FGEN over 100 days

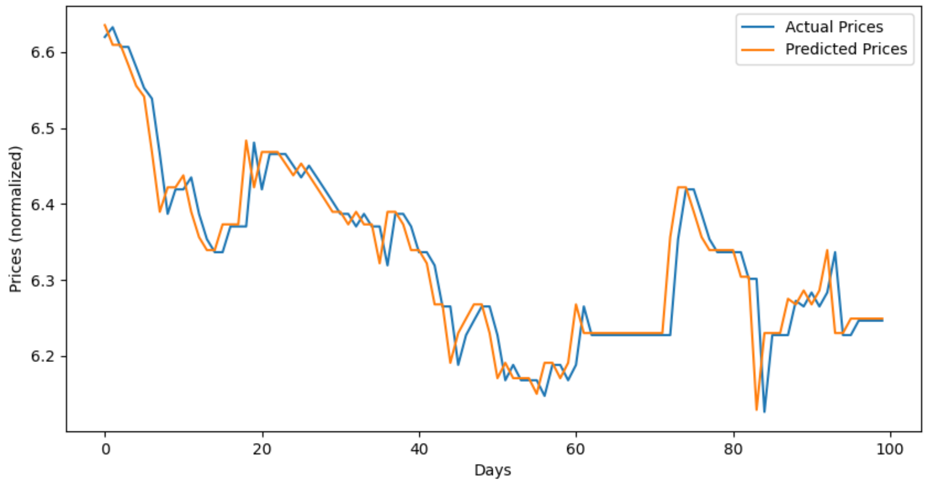

Figure 2.8: Actual vs Predicted Prices for GLO over 100 days

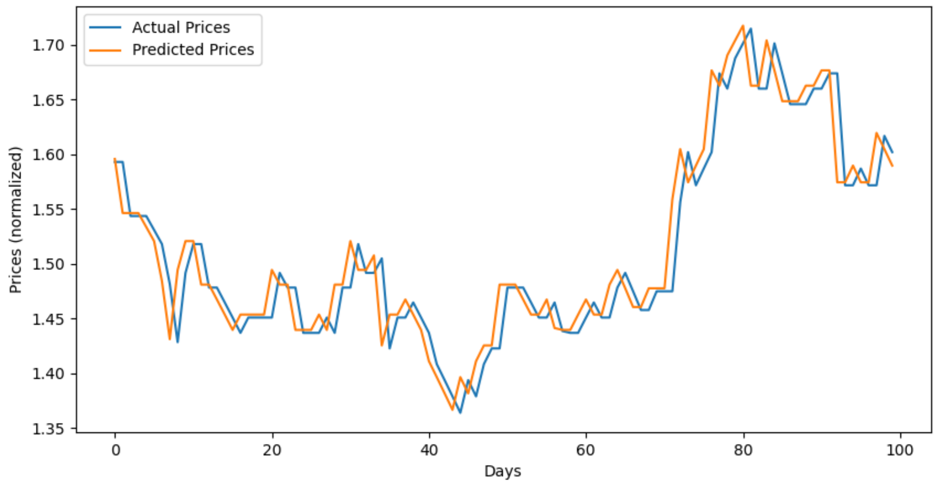

Figure 2.9: Actual vs Predicted Prices for ICT over 100 days

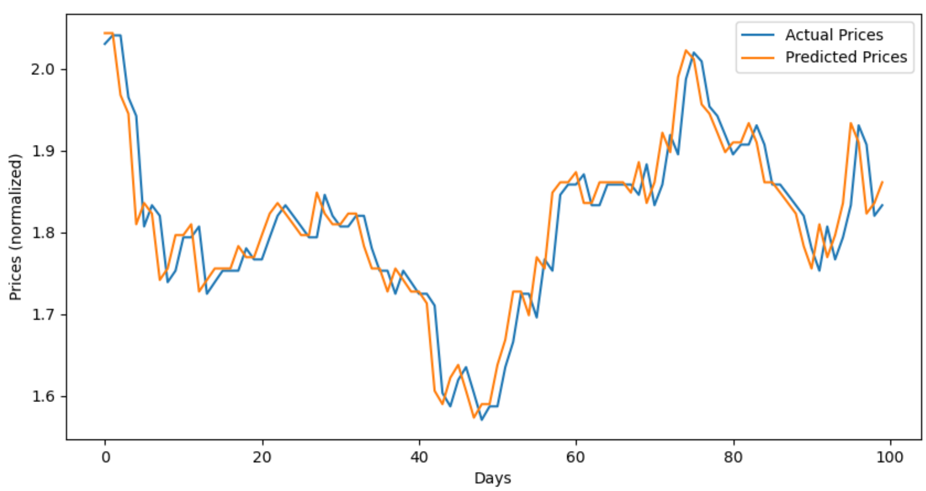

Figure 2.10: Actual vs Predicted Prices on JGS for 100 days

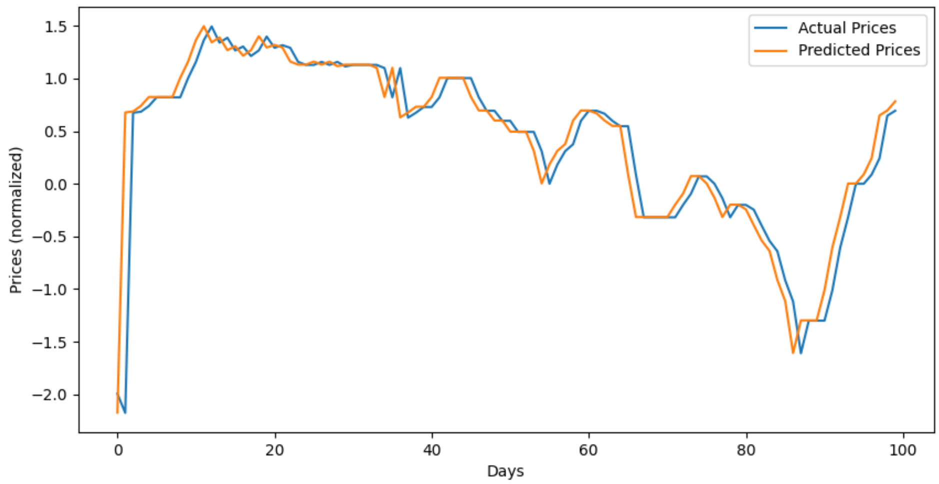

Figure 2.11: Actual vs Predicted Prices on LTG for 100 days

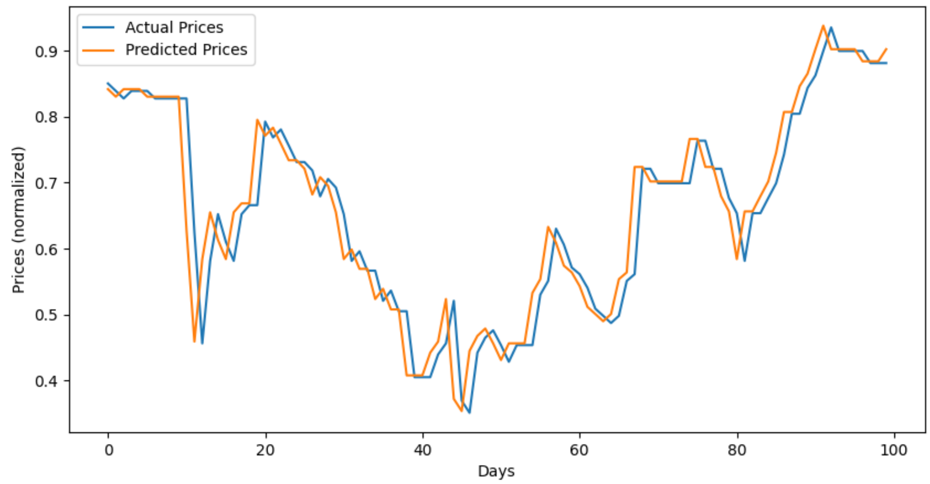

Figure 2.12: Actual vs Predicted Prices on MEG for 100 days

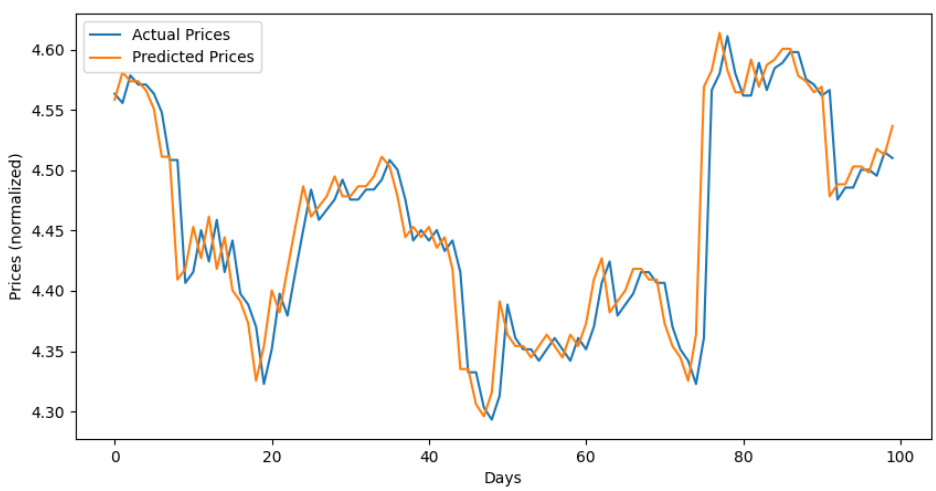

Figure 2.13: Actual vs Predicted Prices on MER for 100 days

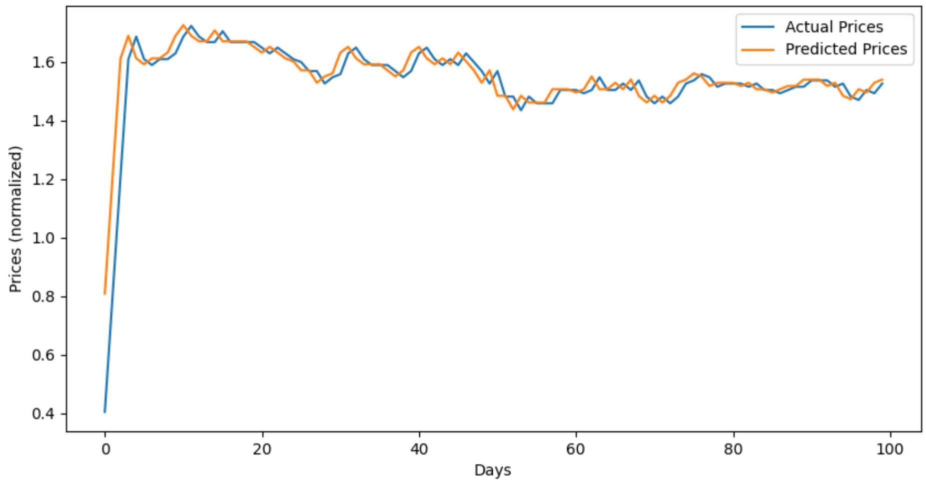

Figure 2.14: Actual vs Predicted Prices on MPI for 100 days

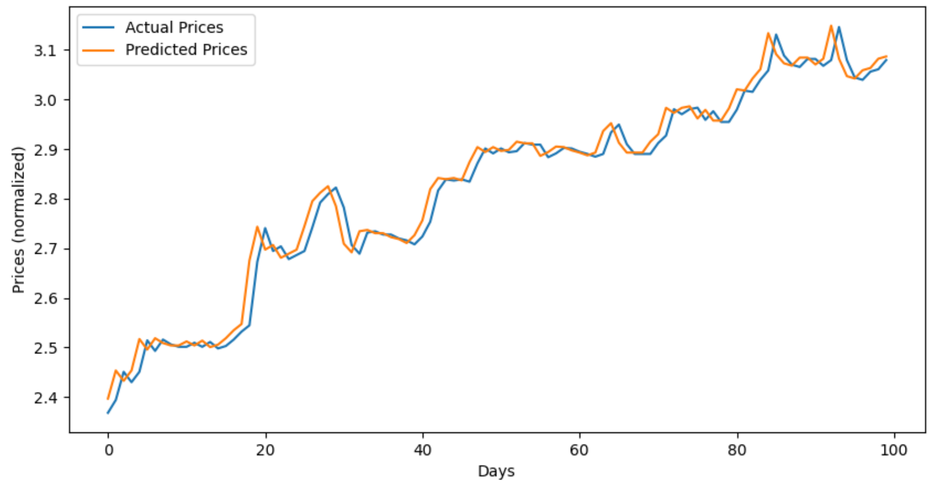

Figure 2.15: Actual vs Predicted Prices on PGOLD for 100 days

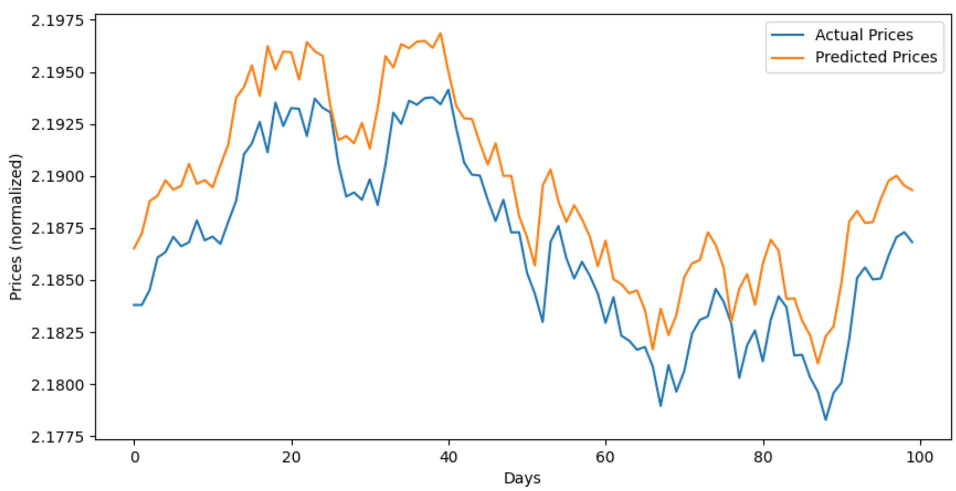

Figure 2.16: Actual vs Predicted Prices on PSEI for 100 days

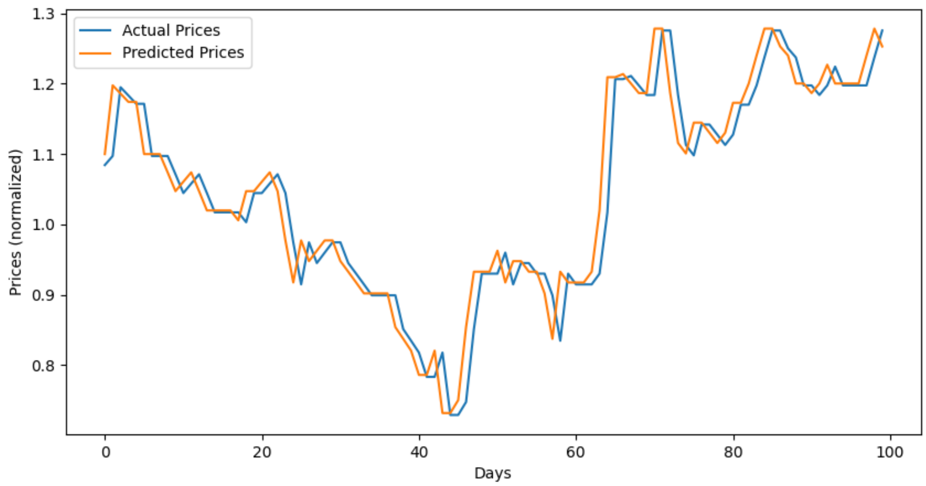

Figure 2.17: Actual vs Predicted Prices on RLC for 100 days

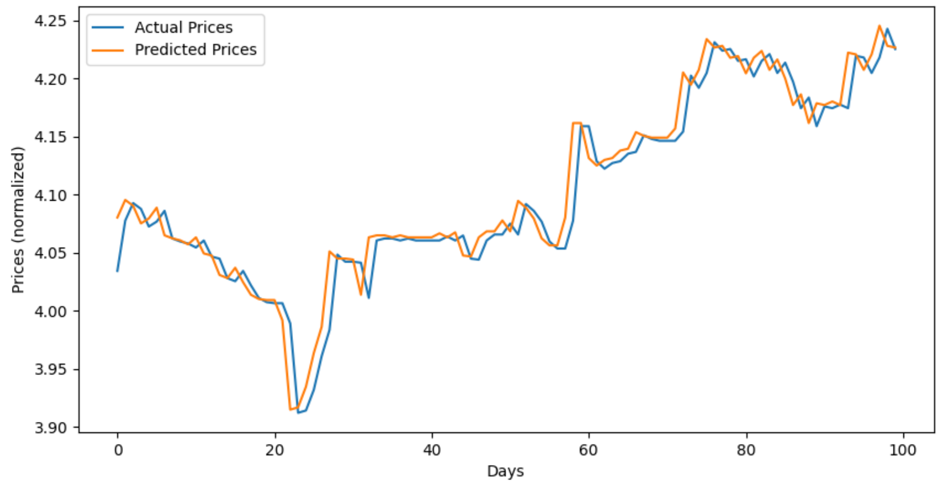

Figure 2.18: Actual vs Predicted Prices on RRHI for 100 days

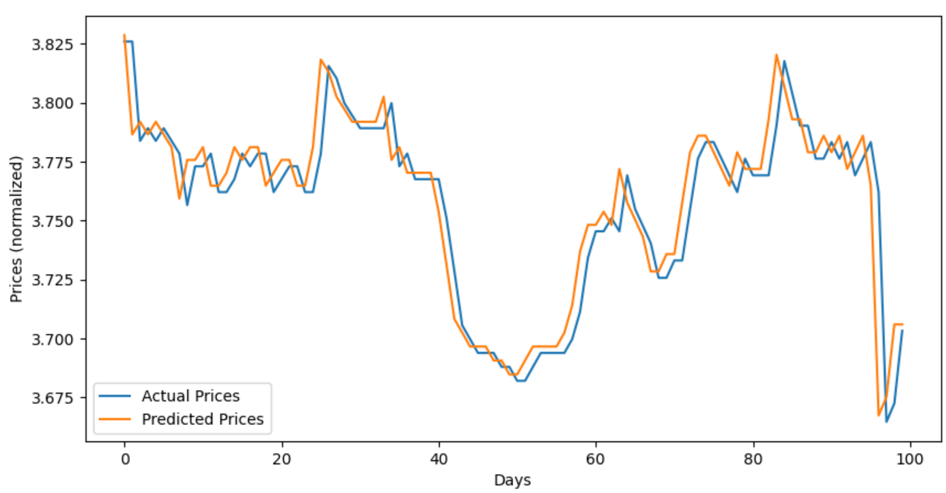

Figure 2.19: Actual vs Predicted Prices on SMC for 100 days

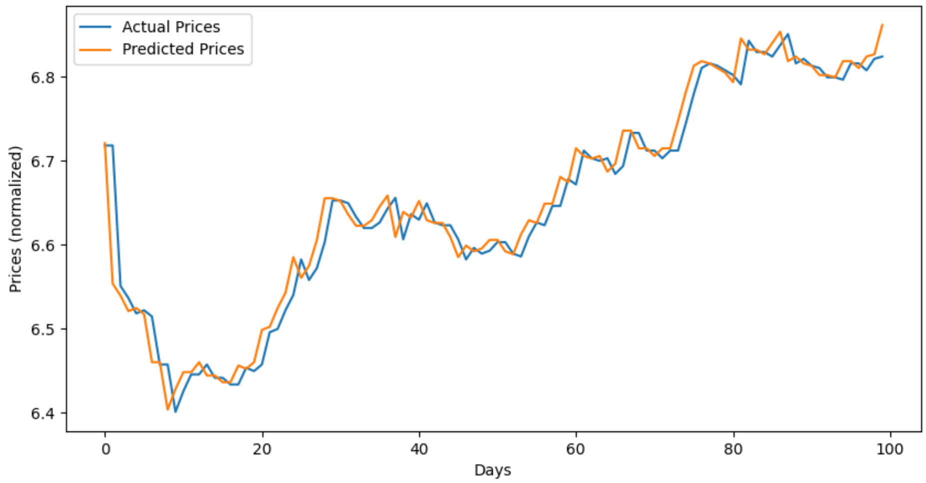

Figure 2.20: Actual vs Predicted Prices on TEL for 100 days

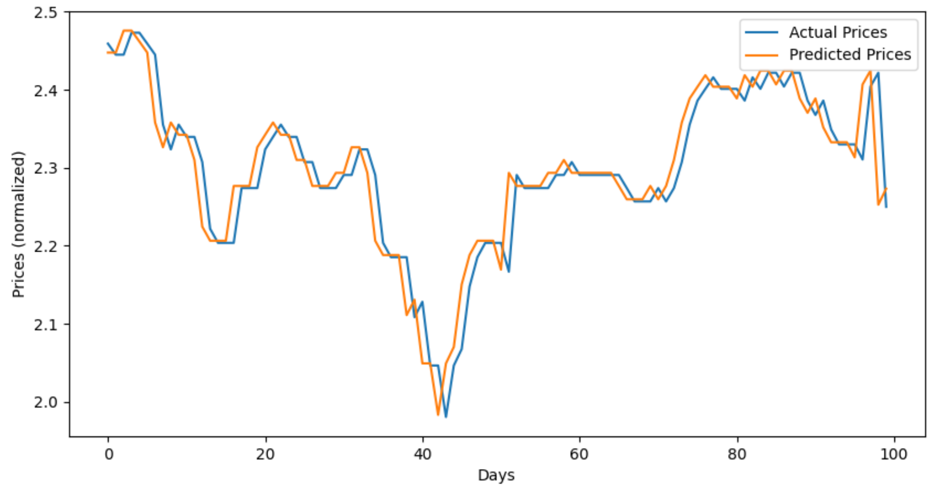

Figure 2.21: Actual vs Predicted Prices on URC for 100 days

The figures above show that the predicted prices follow the actual price trend. In addition, the discrepancy between predicted and actual prices is relatively small, as evidenced by the error metrics scores shown in Table 2.3.

However, the MAPE scores for BLOOM, ICT, JGS, LTG, and MEG range from ten billion to hundred billion. This outlier in the data is, fortunately, just the result of the applied logarithmic normalization, where some of the data in the datasets of the aforementioned stocks are in the negative range, that influence the calculation of the MAPE scores using the scikit-learn library. Because this library handles the calculation of the MAPE scores, there is no way to fix this bug. Moreover, if we take a look at the graphs of the 100 days prediction versus the actual for the aforementioned stocks in Figures 2.6, 2.9, 2.10, 2.11, and 2.12, respectively, it can still be observed that the model performs well on these stocks.

Not to mention that the other error metrics used show the same performance levels across the different stocks when the DMD-LSTM model is utilized. Meanwhile when the data normalization is removed, the MAPE scores for BLOOM, ICT, JGS, LTG, and MEG become 0.068108, 0.037207, 0.039754, 0.057332, and 0.044411 units, respectively.

Another observation from the graphs comparing actual and predicted prices over 100 days is that the predicted values appear to be higher than the actual prices. This indicates the possibility of loss because the model overestimates its prediction.

The successive predictions for the following day and up to ten days were tested using the price data from PSEI in order to make the system's predictions more useful for actual utilization. Table 2.4 shows the MAPE scores for the successive predictions of the DMD-LSTM for each days.

Table 2.4: DMD-LSTM Successive Predictions

Successive Days Predicted	**Actual and Predicted Data Ratio**	**MAPE Score**
1	100%	0.00973
2	80%	0.13403
3	60%	0.15782
4	40%	0.15646
5	20%	0.13910
6	0%	0.12494
7	-20%	0.11283
8	-40%	0.10014
9	-60%	0.08914
10	-100%	0.08976

From the table above it must be noted that the ratio values highlighted in red is to demonstrate that, despite the fact that negative ratio values shouldn't exist, doing so simply indicates that the data used to forecast the subsequent price data was overlapping by 2 to 5 times, depending on the ratio, and no longer used any actual data.

Moreover, in the integration of the DMD-LSTM model to the alamSYS, the

5 days successive predictions was utilized. Where it is shown from the Table 2.4 that it still performs well, even if the actual and predicted data ratio is only at 20%. This is also to limit the effect of stock market volatility that might affect the accuracy of the successive predictions of the model.

However, it can also be observed that the MAPE scores for successive days with zero to negative actual and predicted data ratio outperforms the MAPE scores from successive days 2 to 5 as illustrated in Figure 2.22, shown below.

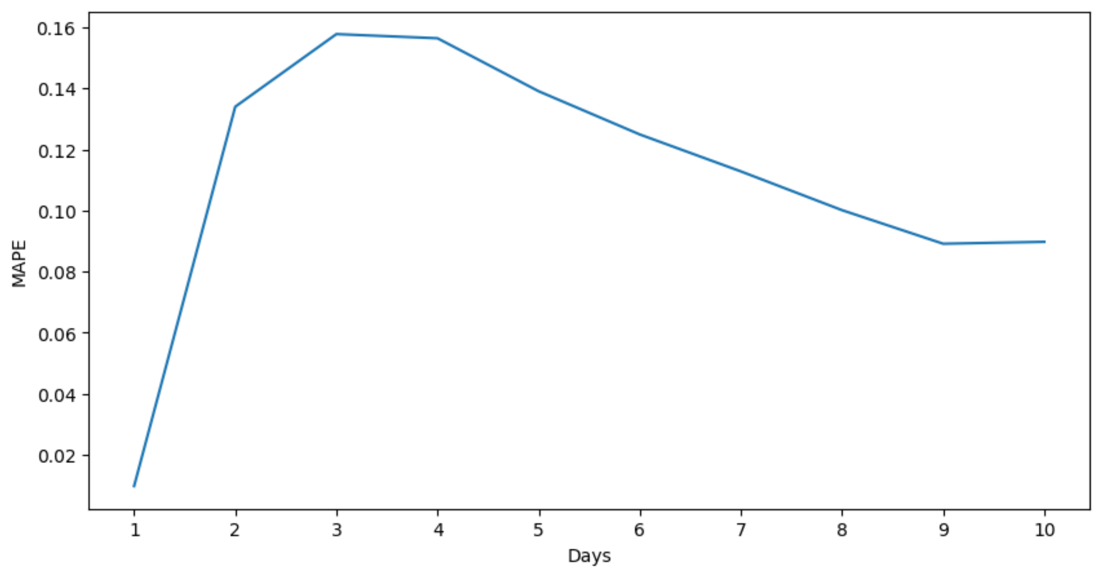

Figure 2.22: MAPE Scores for 1 to 10 (Days) Successive Predictions

Yet, since doing so might result in a poor generalization of data, they were not chosen to be the maximum consecutive days of predictions to be integrated in the alamSYS. As a matter of fact, it could be argued that these data's MAPE scores are overfitted, rendering them unreliable. On the contrary, it might also imply that the model maintains its accuracy for a longer time, even if the majority of the data used are those produced by the model itself. This could be a good thing, and may be attributed to the use of the dynamic modes, as first suggested in the study of Mann and Kutz (2015). In light of these considerations, additional testing is required to establish which of the two claims is true.

Overall, the results from the model training, evaluation, and cross-validation shows that the DMD-LSTM model developed in this special problem performs on par with the other studies that utilizes dynamic modes, as mentioned in Chapter 1 of this paper.

2.3 ALMACD Results and Discussions

The ability to predict consecutive days in the stock market is useless without a trading strategy - which allows risk mitigation and increases the probability of positive returns over time. Trading strategies, in particular, are based on a pre-defined set of rules and criteria that are used to determine when to buy and sell stocks. (Hayes, 2022).

A variety of algorithmic trading, on the other hand, refers to the use of mathematical and computational techniques to determine the best position to take for a specific set of stocks. Additionally, the possibility of loss due to the influence of human emotion is eliminated. (WallStreetMojo, n.d.).

Whereas, the author used the Arnaud Legoux Moving Average Convergence and Divergence (ALMACD) trading strategy in this special problem and integrated it into the alamSYS as the system's internal trading algorithm. ALAMCD uses predicted prices for the next 5 days, as well as 200 days of actual stock price data, to track the signals and output a simple flag indicating whether to buy or sell that stock at that time.

The compounded expected return after return backtesting is provided for each stock in Table 2.5 using the optimized parameters for the fast and slow ALMA. This was done to validate the potential returns for all stocks, not just the PSEI, from which the best ALMA parameters were derived.

Table 2.5: Optimal Alma Parameters Validation Results

Stock	Compounded Expected Return
PSEI	113966.8500
AC	20893.1914
ALI	1072.1418
AP	690.7100
BDO	2541.9970
BLOOM	495.4600
FGEN	581.0804
GLO	60538.0035
ICT	2815.6103
JGS	1569.8650
LTG	397.2854
MEG	149.2233
MER	8586.0306
MPI	146.0200
PGOLD	721.2700
RLC	649.4767
RRHI	1050.7000
SMC	2557.0770
TEL	72070.5000
URC	3207.5394

Based on the table of expected returns above, all stocks are expected to return a positive yield over time when these optimal ALMA parameters are used. It is also worth noting that the expected return is calculated for each unit of stock, which means that if we use the expected compounded return value of MPI at PHP 146.02, which appears to be the lowest - the actual return could be at least PHP 146,020, assuming the minimum board lot required for the stocks is 1000 shares (Pesobility, n.d.).

However, despite the high potential returns, investors should proceed with caution for two reasons. First, the expected return is based on historical price data, which may not follow the trend of future price data, potentially rendering the trading algorithm obsolete (Quantified Strategies, 2023). Second, the return calculation does not account for and compensate for the additional fees associated with buying and selling the stock, which can affect the overall actual returns. Moreover, the author investigated the potential for returns by following the alamSYS predictions, as discussed further in Section 2.5.

2.4 alamSYS System Tests Results and Discussions

xxx

Table 2.6: Idle System Average Resource Usage Statistics

	alamAPI	**alamDB**	**alamPREPROCESSOR**
CPU Utilization (%)	0.168125	0.254313	0.009769
Memory Utilization (MiB)	45.718311	166.775377	312.798300

xxx

Table 2.7: Internal Load Average Resource Usage Statistics

	Data Collector	**Data Processor**	**alamSYS PREPROCESSOR (Data Collector & Data Processor)**
Failure Rate (%)	0	0	0
Success Rate (%)	100	100	100

Table 2.7 continued from previous page

	Data Collector	Data Processor	alamSYS PREPROCESSOR (Data Collector & Data Processor)
Average Runtime (s)	41.72398	8.38061	48.30466
Average CPU Utilization (%)	11.40659	92.71117	20.03138
Average Memory Utilization (MiB)	3.64200	57.09545	794.29436
Average Network Utilization (Mb)	232.73640	154	77.27655

xxx

Table 2.8: Deployment Load Test Results (Buy Requests)

	Number or Requests		
	10	**100**	**1000**
Success Rate (%)	100	100	100
Average Processing Time (s)	11.905222	139.618550	1159.773569

xxx

Table 2.9: Deployment Load Test Results (Sell Requests)

	Number or Requests		
	10	**100**	**1000**
Success Rate (%)	100	100	100

Table 2.9 continued from previous page

	Number or Requests		
Average Processing Time (s)	13.384126	130.119867	1642.995011

xxx

2.5 Results and Discussions for the Real World Application of alamSYS

xxx

Table 2.10: Return Performance Comparison Between alamSYS and PSEI

	Realized Profit (PHP)	Realized Gain (%)
alamSYS	7,839.75	1.51
PSEI	-22,788.90	-13.810

xxx

Chapter 3

Conclusions and Future Work

xxx

References

Agrawal, M., Khan, D. A. U., & Shukla, D. P. K. (2019, 7). Stock price prediction using technical indicators: A predictive model using optimal deep learning. *International Journal of Recent Technology and Engineering (IJRTE)*, *8*, 2297-2305. doi: 10.35940/ijrteB3048.078219

Cui, L. X., & Long, W. (2016, 11). Trading strategy based on dynamic mode decomposition: Tested in chinese stock market. *Physica A: Statistical Mechanics and its Applications*, *461*, 498-508. doi: 10.1016/j.physa.2016.06.046

Dash, R., & Dash, P. K. (2016, 3). A hybrid stock trading framework integrating technical analysis with machine learning techniques. *The Journal of Finance and Data Science*, *2*, 42-57. doi: 10.1016/j.jfds.2016.03.002

Hayes, A. (2022). *What is a trading strategy? how to develop one.* Retrieved April 22, 2023, from `https://www.investopedia.com/terms/t/trading-strategy.asp`

Hua, J.-C., Roy, S., McCauley, J. L., & Gunaratne, G. H. (2016, 4). Using dynamic mode decomposition to extract cyclic behavior in the stock market. *Physica A: Statistical Mechanics and its Applications*, *448*, 172-180. doi: 10.1016/j.physa.2015.12.059

Liu, C., Yan, J., Guo, F., & Guo, M. (2022, 8). Forecasting the market with machine learning algorithms: An application of nmc-bert-lstm-dqn-x algorithm in quantitative trading. *ACM Transactions on Knowledge Discovery from Data*, *16*, 1-22. doi: 10.1145/3488378

Mann, J., & Kutz, J. N. (2015, 8). Dynamic mode decomposition for financial trading strategies.

Obthong, M., Tantisantiwong, N., Jeamwatthanachai, W., & Wills, G. (2020). A survey on machine learning for stock price prediction: Algorithms and techniques. In (p. 63-71). SCITEPRESS - Science and Technology Publications.

doi: 10.5220/0009340700630071

Pang, X., Zhou, Y., Wang, P., Lin, W., & Chang, V. (2020, 3). An innovative neural network approach for stock market prediction. *The Journal of Supercomputing*, *76*, 2098-2118. doi: 10.1007/s11227-017-2228-y

Pesobility. (n.d.). *Board lot of philippine stock exchange.* Retrieved April 22, 2023, from https://www.pesobility.com/ref/boardlot

Quantified Strategies. (2023). *Why trading strategies are not working (how to know when trading systems stopped performing?).* Retrieved April 22, 2023, from https://www.quantifiedstrategies.com/why-trading-strategies-are-not-working/

Savaș, M. C. (2017). *Algorithmic trading strategies using dynamic mode decomposition: Applied to turkish stock market.* Retrieved from https://etd.lib.metu.edu.tr/upload/12621107/index.pdf

SCHMID, P. J. (2010, 8). Dynamic mode decomposition of numerical and experimental data. *Journal of Fluid Mechanics*, *656*, 5-28. doi: 10.1017/S0022112010001217

S.Gopal Krishna Patro, K. K. s. (2015). Normalization: A preprocessing stage. *International Journal of Advanced Research in Science, Engineering and Technology*, *205*. Retrieved from 10.17148/IARJSET.2015.2305

Uchiyama, Y., & Nakagawa, K. (2021, 7). Improving momentum strategies using adaptive elastic dynamic mode decomposition. In (p. 388-393). IEEE. doi: 10.1109/IIAI-AAI53430.2021.00068

WallStreetMojo. (n.d.). *Algorithmic trading.* Retrieved April 22, 2023, from https://www.wallstreetmojo.com/algorithmic-trading/

Williamson, J. G. (2015). *Handbook of the economics of international migration.* Retrieved December 15, 2022, from https://www.sciencedirect.com/topics/economics-econometrics-and-finance/global-economic-crisis