Effective Crude Oil Trading Techniques Using Long Short-Term Memory and Convolution Neural Networks

Wisaroot Lertthaweedech, Pittipol Kantavat, and Boonserm Kijsirikul
Department of Computer Engineering, Faculty of Engineering, Chulalongkorn University, Bangkok, Thailand
Email: 6370267221@student.chula.ac.th, {pittipol.k, boonserm.k}@chula.ac.th

***Abstract*—Crude oil plays a vital role in the global economy and forecasting crude oil prices is crucial for both government and private sectors. However, the crude oil price is high volatility, influenced by various factors and challenging to predict. Thus, various machine learning techniques have been proposed to predict crude oil prices for decades. In this study, we propose an Artificial Neural Network (ANN) with different combinations of Convolutional Neural Networks (CNN) and Long Short-Term Memory (LSTM) to improve the trend forecasting of crude oil prices for better trading signals compared to traditional strategies. As the crude oil price is a time series data, it is appropriate to apply CNN and LSTM for forecasting. The concept of our model is that CNN could detect features or patterns in different locations of time series data, while LSTM could maintain both short-term and long-term memory along with time series data. The collaboration of their abilities could help the neural network model understand complex relationships of historical data and trends of crude oil prices. Our study found that the combination of CNN and LSTM could significantly enhance trading performance in the long run.**

***Index Terms*—crude oil trading, machine learning, deep learning, trading signal, technical analysis, artificial intelligent**

I. Introduction

Crude oil is a commodity that significantly impacts the world economy because 30% of the overall energy supply in the world uses crude oil as the source of energy [1] Products of refined crude oil are used in various economic activities such as power generation, raw material for the petrochemical industry, and transportation by vehicles, ships, and airplanes. Therefore, crude oil price possesses direct influences on many industries that are related to these economic activities. In addition, the crude oil price is a vital factor in the global economy for inflation forecast, monetary policy, and fiscal policy by the government sector [2]. However, the crude oil price is very volatile. It depends on the dynamic condition of demand and supply, including the growth of economic activities, technology, alternative energy like natural gas, coal, renewable energy, black-swan event like the coronavirus disease of 2019 (COVID-19).

There are several research on crude oil price prediction models for decades, such as Support Vector Machine (SVM) [3], [4], Autoregressive Integrated Moving Average (ARIMA) [3]-[5], Random Walk [3], Genetic Algorithm (GA) [6], Generalized Autoregressive Conditional Heteroskedasticity (GARCH) [7], Vector Autoregressive (VAR) [8], Error Correction Model (ECM) [7], [9], Thanks to the development of computing technology, machine learning models with complex algorithms, such as deep learning, are more popular and have performed higher performance in recent years [10].

Different types of deep learning models were proposed to implement various asset price forecasting models [11]. Convolutional Neural Network (CNN) was one of the most popular types of Artificial Neural Network (ANN) layer for stock price prediction. CNN could learn for feature selection automatically [12]. Gudelek *et al.* [13] used 2D CNN to classify types of stock price movement from the history of stock price and technical indicator data. They proposed 2-layers of CNN followed using a 3×3 filter size for both layers. The number of filters was 32 and 64 for each CNN layer, respectively. The performance evaluation results indicated that their proposed CNN models could predict stock prices movement with high accuracy and outperform Buy & Hold strategy. Tsantekidis *et al.* [14] proposed multi-layers of CNN to predict stock price movement using high-frequency time series derived from the order book. Their CNNs model was composed of 2 sets of convolution and pooling layers followed by two dense layers. They compared model performance with the linear SVM model and MLP model. They found that the proposed CNN model could predict stock price with higher accuracy than other models. Lee *et al.* [15] applied CNN with Deep Q-Network to predict various stock prices and perform the trading test in many stock markets globally. The proposed model utilized stock chart images as input for CNN as a function approximator. Then, CNN created feature maps as a representation for action. The portfolio from their backtest performed well in many stocks market over 12 years of testing.

The Long-Short Term Memory model (LSTM) was another popular type of ANN layer for stock price

Manuscript received March 10, 2022; revised May 17, 2022; accepted May 26, 2022.

doi: 10.12720/jait.13.6.645-651

prediction [16] because LSTM was designed to have a cell state that could maintain memory inside the neural network for both long and short sequences of data. LSTM was more appropriate for time series data with long data sequences than other types of neural networks [17]. Li *et al.* [18] proposed two layers of the LSTM model to predict stocks in China and compared the performance with SVM. They found that LSTM has higher accuracy than SVM for the low-variance stock price. Cheng *et al.* [19] proposed Attention-based LSTM to predict types of stock price movement from features extracted from daily News. Pang *et al.* [20] proposed an LSTM model consisting of an embedding layer and three layers of LSTM. They employed an embedding layer to convert high-dimensional data into low-dimensional data. Their proposed LSTM model can improve accuracy and mean square error from the multilayer perceptron model. Fischer and Krauss [21] deployed a 1-layer LSTM with a dropout layer to predict stock in S&P 500 market. The proposed LSTM models could extract meaningful information from noisy financial time series data. The trading performance of this model was also outperformed other techniques in their study, including random forest and logistics regression.

In this study, we propose combinations of LSTM and CNN models to predict crude oil price trends. We use the WTI crude oil prices during the period 2015 to 2020 from Yahoo finance [22] for experimental data. Prediction error and trading performance are used as evaluation metrics. We summarize that our proposed models could significantly enhance trading performance, as discussed in the experimental result section.

The remaining parts of this paper are structured as follows: Section II provides basic knowledge of the neural network used in this study, including CNN and LSTM. Section III is a literature review. Section IV illustrates our proposed models. Section V describes the source of the experimental dataset and data preprocessing for the experiment. Section VI provides details of data preprocessing and labeling. Section VII analyzes and discusses the experimental results. Lastly, Section VIII concludes the paper.

II. Related Theory

A. Convolutional Neural Network (CNN) [23]

The Convolutional Neural Network (CNN) is a neural network that is widely used for computer vision tasks. In contrast to the original neural network that fully connects all pairs of nodes between layers, CNN decreases the number of edges between the previous layer and its layer to simplify the network as shown in Fig. 1. Each node in the CNN layer will only connect some nodes of the previous layer, and these edges are called filters. The weights in the filters can be learned through the backpropagation process. Because the same edges' weight will be applied to all the connected nodes in the CNN layer, CNN can discover patterns in different locations of time series data.

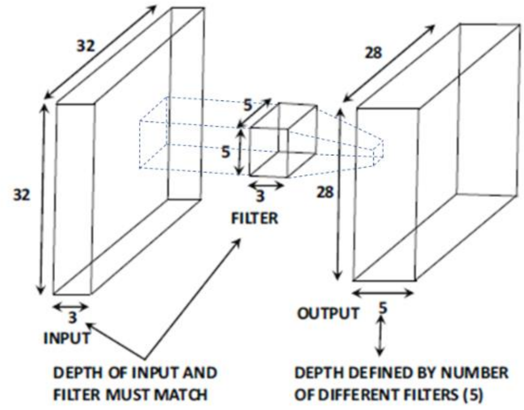

Figure 1. The architecture of CNN [23].

B. Long Short-Term Memory (LSTM)

Long Short-Term Memory (LSTM) is proposed by Hochreiter and Schmidhuber [17] to solve a problem in a recurrent neural network, called the vanishing gradient. It could happen when the data sequence is too long, leading to a vanishingly small gradient to update the weight. LSTM learns long data sequences using four gates, as shown in Fig. 2. A forget gate f_t controls the memory of the previous data sequence as in (1). Input gates i_t and $\tilde{C}_t$ control new information as in (2) and (3). An output gate o_t controls the output of hidden states h_t as in (4) and (5).

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

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \tag{2}$$

$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C) \tag{3}$$

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \tag{4}$$

$$h_t = o_t * \tanh(C_t) \tag{5}$$

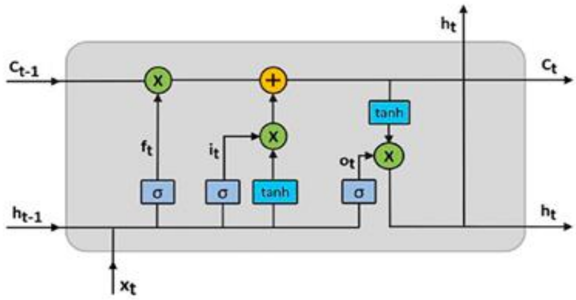

Figure 2. The architecture of LSTM [12].

III. Literature Review

This part reviewed related literature that applied similar techniques used in this study. Firstly, we reviewed literature that applied CNN and LSTM to various financial time series data. In addition, we reviewed the survey paper to identify potential input features that could enhance the prediction capability of the model.

Rezaei *et al.* [12] proposed a CNN-LSTM model to predict stock indexes globally, as illustrated in Fig. 3. Original stock indexes data were decomposed by Empirical Mode Decomposition (EMD) and Complete Ensemble Empirical Mode Decomposition (CEEMD) algorithms into different frequency spectra. After decomposition, the data were normalized before feeding to CNN model in order to process the pattern of each decomposed spectra separately. Then, it was transferred to LSTM model to process data with the previous time step. Finally, they summed up the processed frequency spectra for prediction results. The model could predict stock indexes with lower RMSE than that of vanilla LSTM, Support Vector Regression, and Decision Tree Regression across all markets.

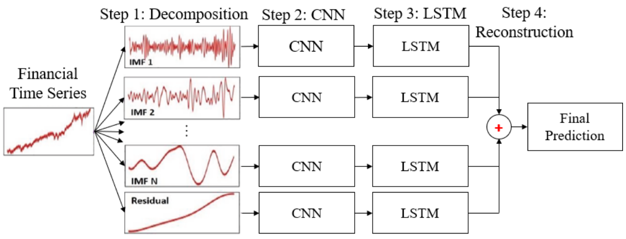

Figure 3. The architecture of CNN-LSTM proposed by Rezaei *et al.* [12].

TABLE I. LIST OF TECHNICAL INDICATORS USED AS INPUT FOR THE MODEL

Trend Indicators	**Volatility Indicators**
Simple Moving Averages (SMA)	Bollinger Bands
Moving Average Convergence Divergence (MACD)	Average True Range (ATR)
Average Directional Movement Index (ADX)	Ulcer Index (UI)
Commodity Channel Index (CCI)	**Volume indicators**
Momentum Indicators	Accumulation/Distribution Index (ADI)
Rate of Change (ROC)	On-balance volume (OBV)
Relative Strength Index (RSI))	Chaikin Money Flow (CFI)
True Strength Index (TSI)	Force Index (FI)
Stochastic RSI %K (%K)	Money Flow Index (MFI)
Stochastic RSI %D (%D)	Volume-price trend (VPT)
Williams %R (%R)	Volume Weighted Average Price (VWAP)

Kumar *et al.* [11] conducted a survey paper related to stock market forecasting using computational intelligence. They found that over 50% of their reviewed research applied technical indicators as input features and technical indicators have led to outstanding results for stock market prediction. Kumar *et al.* [24] also suggested technical indicators listed in Table I to construct a feature vector for stock prediction using ANN. These indicators were categorized based on their inbuilt capability. Trend indicators, such as Simple Moving Averages (SMA), Moving Average Convergence Divergence (MACD), Average Directional movement Index (ADX) and Commodity Channel Index (CCI), were favorable for stock trend analysis. Momentum indicators, such as Rate of Change (ROC), Relative Strength Index (RSI) and True Strength Index (TSI), were preferable to measure momentum, i.e., speed of price changes in a certain period. Stochastic oscillators like Stochastic RSI %K, Stochastic RSI %D and Williams %R were also momentum indicators used to identify overbought and oversold signals. Volatility Indicators like Bollinger bands and Average True Range (ATR) were used to measure stock price volatility. Lastly, volume indicators, such as Accumulation/Distribution Index (ADI), On-Balance Volume (OBV) and Chaikin Money Flow (CMF), were used to analyze sentiment pressure in the market.

IV. PROPOSED METHOD

We proposed two combinations of CNN and LSTM, including 1. CNN-LSTM and 2. LSTM -CNN as shown in Fig. 4 and Fig. 5, respectively. The models are inspired by CNN-LSTM by Rezaei *et al.* [12]. While their works focus on frequency decomposition of the input time series. Our works focus more on effect of combination and order of CNN and LSTM layers.

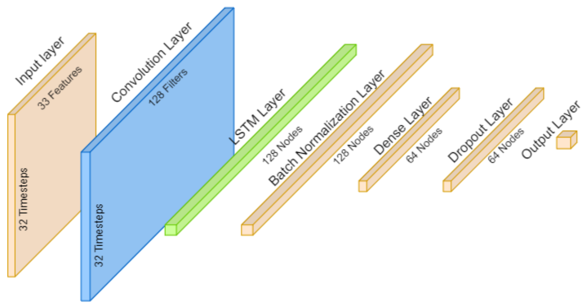

Figure 4. CNN-LSTM architecture.

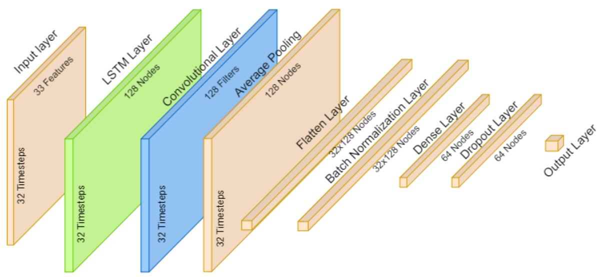

Figure 5. LSTM-CNN architecture.

We also run experiments for single LSTM and single CNN to compare our proposed models' performance. After input layers, the inputs are fed into CNN and LSTM, which are varied based on a combination of CNN and LSTM. For CNN-LSTM model, the inputs are fed to CNN layers first and transferred to LSTM layer.

In contrast, for LSTM-CNN model, the inputs are fed to LSTM layers first and transferred to CNN layer. After the CNN layer, we applied average pooling and flatten layer for dimension reduction. Afterward, all models are followed by batch normalization layer, dense layer, dropout layer, and finally output layer.

CNN layer could extract critical information across time series data, while LSTM could enhance the output accuracy by maintaining state across long-term and short-term sequences of data [12]. Batch normalization is used for normalized vectors after a combination of CNN and LSTM layers to accelerate convergence and reduce the required epoch of training [25]. The dense layer is used to process information from the earlier layers before passing it to the output. The dropout layer is applied to reduce overfitting issues.

V. Experimental Dataset

We use the daily West Texas Intermediate (WTI) crude oil price during the period 2005 to 2020 from Yahoo Finance [22]. WTI crude oil is a specific grade of crude oil and one of the world's primary benchmarks for the crude oil price. The daily historical data include open, high, low, close, and volume. Then, we calculate those pricing data into technical indicators following Kumar *et al.* [24], as shown in Table I.

Finally, we separate data into six groups by sliding windows. We set the name of each group corresponding to the trading backtest year from 2015 to 2020. Each group is split into training data (9-year), validation data (1-year after training data) and test data (1-year after validation data), as demonstrated in Table II.

TABLE II. Separation of Data for the Experiment

Group	2005	2006	2007	2008	2009	2010	2011	2012	2013	2014	2015	2016	2017	2018	2019	2020
2015																
2016																
2017																
2018																
2019																
2020																

Training Validation Test

VI. Data Preprocessing and Labeling

We transform data using two techniques to increase stationary in the time series data. Firstly, we transform open, high, low, SMA, Bollinger Bands, ATR, and VWAP to relative value compared to the closed prices by (6). Secondly, we transform data by applying relative change compared to the previous time-step for close, volume, ADI, and OBV by (7). In addition, we utilize quantile transformation to re-distribute features with highly skewed distribution, including volume, ADI, OBV, and VPT. After that, all data are normalized in the range [0, 1] before training neural network models by (8).

$$X_{t,\ Relative\ Value} = \frac{X_t - C_t}{C_t} \tag{6}$$

$$X_{t,\ Relative\ Change} = \frac{X_t - X_{t-1}}{X_t} \tag{7}$$

$$X_{t,\ Normalized} = \left(\frac{X_t - X_{min}}{X_{max} - X_{min}}\right) \tag{8}$$

Labeling consists of 2 steps, including 1. smoothing close prices by Savitzky-Golay filter [26] and 2. transforming the smooth close price to daily return. The results are demonstrated in blue line in Fig. 6.

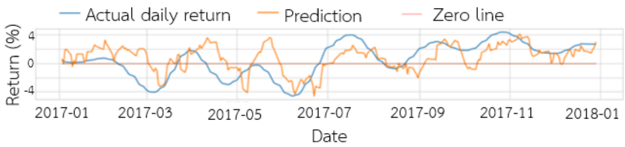

Figure 6. Example of the daily return of the smooth prices by Savitzky-Golay filter.

VII. Experimental Results

This section discussed two different evaluation metrics for different purposes, including 1. prediction error and 2. trading performance. The prediction error investigates the overall prediction capability of the model, while the trading performance evaluates profitability by backtest the portfolio constructed from the trading signal from the model. These performances are assessed and compared to traditional strategies such as buy and hold, RSI, Stochastics, MACD, and SMA.

A. *Prediction Error*

The models are trained using training data and evaluated using Mean Square Error (MSE) with configuration in Table III. We perform a grid search for the number of filters between 64 and 128 in the CNN layer and the number of nodes between 64 and 128 in LSTM layer using validation data. Then, we select the grid configuration with the lowest MSE for trading performance evaluation using test data. We process the experiment three times with different random seeds and average them as experimental results.

The MSEs of the test data for each data group are shown in Table IV. The bold texts indicate the lowest MSE of each year. The combinations of CNN and LSTM, CNN-LSTM and LSTM-CNN, significantly outperform single CNN or single LSTM, yielding the same level of MSEs. Remark that the MSEs of all models in 2020 are higher than other years because the crude oil price is very volatile by the COVID-19 situation.

TABLE III. Model Training Configurations

Parameters	**Configurations**
Batch size	128
Optimizer	Adam
Learning rate	1×10^{-5} with decay factor 0.95 and patience = 10 epochs
Stopping algorithms	500 epochs or patience = 30 epochs

TABLE IV. AVERAGE MSE OF THE TEST SET FOR EACH DATA GROUP

	CNN	LSTM	CNN-LSTM	LSTM-CNN
2015	2.5	2.2	1.7	**1.7**
2016	2.3	**1.4**	1.8	1.6
2017	1.1	**0.6**	0.7	0.8
2018	1.5	1.3	**0.8**	1.2
2019	1.2	1.1	0.9	**0.8**
2020	29.3	31.0	**25.1**	25.1
Avg.	6.3	6.3	**5.2**	5.2

a. The units of MSE on the table are 10^{-2} $\%^2$.
b. The bold texts indicate the lowest MSE for each year.

B. *Trading Performance*

We deploy the trained models, including CNN, LSTM, CNN-LSTM, and LSTM-CNN, to perform trading backtest. Backtest is trading simulation based on buying signal and selling signal. The trading signal is triggered from the predicted daily return from the model, as shown in the orange line in Fig. 6. We set zero as a threshold. If the smooth daily return breaks threshold up, it is triggered for a buying signal. On the other hand, if the smooth daily return breaks threshold down, it is triggered for a selling signal.

The performance was evaluated by two trading evaluation metrics, including 1. Return on Investment (ROI) and 2. Sharpe ratio. The ROI measure profitability of an investment calculated by (9). We set the transaction fee at 0.1%. The Sharpe ratio is a widely used matric for risk-adjusted return of portfolios, calculated by the ratio of return and risk as shown in (10).

$$ROI = \frac{Portfolio\ Value_i - Portfolio\ Value_{i-1}}{Portfolio\ Value_{i-1}} \quad (9)$$

$$Sharpe\ Ratio = \frac{R_p - R_f}{\sigma_p} \quad (10)$$

ROI for each group is shown in Table V. The bold texts indicate the highest ROI for each year. The green cells indicate positive ROI and the red cells indicate negative ROI. The average ROI of all groups is shown in Fig. 7. The average annualized ROI of LSTM-CNN at 17.8% is the highest among all strategies, while CNN-LSTM provided a slightly lower annualized ROI at 16.6%. LSTM-CNN and CNN-LSTM is the only two strategies that provide positive ROI for all years.

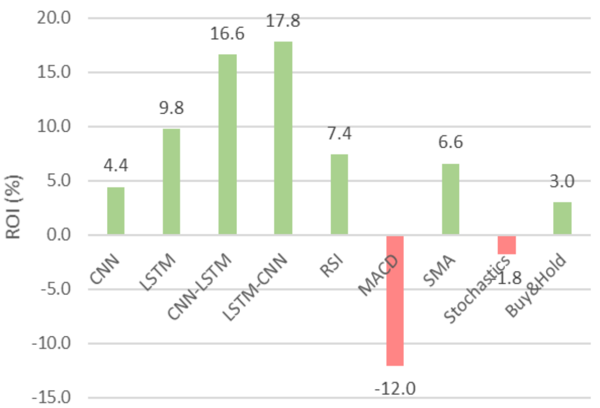

Figure 7. Average ROI across all groups during 2015 - 2020 for all trading strategies.

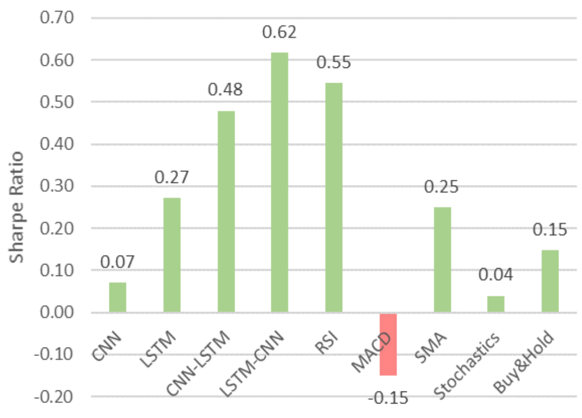

Figure 8. Average Sharpe ratio across all groups during 2015 - 2020 for all trading strategies.

TABLE V. ROI OF ALL TRADING STRATEGIES

	CNN	LSTM	CNN-LSTM	LSTM-CNN	RSI	MACD	SMA	Stochastics	Buy & Hold
2015	-15.7	-5.9	0.4	**3.8**	-20.0	-1.3	-10.7	-30.4	-29.8
2016	7.8	26.7	49.2	51.1	**67.5**	54.8	21.7	9.8	46.7
2017	3.8	10.9	4.6	25.3	**27.8**	9.1	10.4	25.0	15.6
2018	-6.9	4.0	**4.0**	0.9	-18.3	-13.4	-9.4	-12.1	-25.0
2019	-0.4	-3.7	1.0	7.3	1.0	-21.4	-8.6	0.9	**31.3**
2020	37.8	26.5	**40.5**	18.3	-13.5	-100.0	36.1	-4.2	-20.8

a. The bold texts indicate the highest ROI for each year.
b. The green cells indicate positive ROI and the red cells indicate negative ROI.

TABLE VI. SHARPE RATIO OF ALL TRADING STRATEGIES

	CNN	LSTM	CNN-LSTM	LSTM-CNN	RSI	MACD	SMA	Stochastics	Buy & Hold
2015	-0.24	-0.13	0.06	**0.13**	-0.47	-0.04	-0.43	-0.95	-0.63
2016	0.31	0.83	1.85	1.61	**2.10**	1.79	0.71	0.68	0.98
2017	0.19	0.57	0.24	1.40	**2.44**	0.58	0.66	1.60	0.62
2018	-0.37	0.24	**0.27**	0.06	-0.71	-0.74	-0.55	-0.53	-0.80
2019	0.01	-0.17	0.02	0.27	0.04	-0.99	-0.41	0.05	**0.92**
2020	0.51	0.30	0.44	0.23	-0.13	-1.50	**1.52**	-0.61	-0.20

a. The bold texts indicate the highest Sharpe ratio for each year.
b. The green cells indicate positive Sharpe ratio and the red cells indicate negative Sharpe ratio.

The average annualized ROIs from the single CNN and single LSTM models underperform both LSTM-CNN and CNN-LSTM with an annualized ROI of 4.4%

and 9.8%, respectively. The traditional strategies also underperform both LSTM-CNN and CNN-LSTM.

The Sharpe ratio of each group is shown in Table VI and the average Sharpe ratio of all groups is shown in Fig. 8. LSTM-CNN yields the highest Sharpe ratio at 0.62. Subsequently, RSI and CNN-LSTM provide Sharpe ratios at 0.55 and 0.48, respectively.

According to Table V and Table VI, a combination of both techniques, CNN and LSTM, can perform positive ROI and Sharpe ratio every year in the experiment. In addition, they typically provide higher ROI and shape ratio than average value of each year. This indicates that the combination of CNN and LSTM layers could enhance trading performance from both single models and the traditional strategies.

LSTM-CNN leads to the best ROI and Sharpe ratio among all strategies. Although the performance between CNN-LSTM and LSTM-CNN are not distinguishable by MSE metrics, LSTM-CNN provides better ROI and Sharpe ratio than CNN-LSTM on average. When comparing year by year, LSTM-CNN typically performs better than or similar to CNN-LSTM. The only year that CNN-LSTM performs better than LSTM-CNN is 2020, which is when the COVID-19 event occurs. In 2020, the oil price had a strong downtrend and a strong uptrend in the same year. Although the MSE of CNN-LSTM and LSTM-CNN are not much different in 2020, a slight difference in the trading signals from the model could lead to a large difference in ROI because of the strong trend during this year.

The trading performances are quite correlated to prediction error. The performance of the combinations of CNN and LSTM layers in either order could enhance performance from single CNN model or single LSTM model for both prediction error and trading performance. This is because the combination model has more complexity in the model and the model could learn to understand the more complex relationship between features and output.

VIII. Conclusion

Various combinations based on CNN and LSTM are proposed in this paper to perform WTI crude oil price trend prediction. We also conduct portfolio backtest during 2015 - 2020. The performances of the models are evaluated in two dimensions, including prediction error and trading performance.

From the experimental results, LSTM-CNN yields the highest performance among all the models because it contains both CNN and LSTM components with the proper ordering and can learn complex relationships from the training data better than the other techniques.

Conflict of Interest

The authors declare no conflict of interest.

Author Contributions

Wisaroot Lertthaweedech proposed the model architectures, conducted the research, and wrote the paper; under the supervision of Pittipol Kantavat and Boonserm Kijsirikul. All authors had reviewed and approved the final version.

Acknowledgment

This research was supported by Chevron Thailand Exploration and Production Company Limited for the first author scholarship studying master's degree.

References

[1] International Energy Agency. Key world energy statistics 2020. [Online]. Available: https://www.iea.org/reports/key-world-energy-statistics-2020

[2] F. Kunze, M. Spiwoks, K. Bizer, *et al.*, "The usefulness of oil price forecasts-Evidence from survey predictions," *Managerial and Decision Economics*, vol. 39, no. 4, pp. 427-446, 2018.

[3] L. Yu, X. Zhang, and S. Wang, "Assessing potentiality of support vector machine method in crude oil price forecasting," *EURASIA Journal of Mathematics, Science and Technology Education*, vol. 13, no. 12, 2017.

[4] W. Xie, L. Yu, S. Xu, *et al.*, "New method for crude oil price forecasting based on support vector machines," *Computational Science*, vol. 3474, pp. 444-451, 2006.

[5] C. Zhao and B. Wang, "Forecasting crude oil price with an Autoregressive Integrated Moving Average (ARIMA) model," in *Fuzzy Information & Engineering and Operations Research & Management*, Springer, 2014, ch. 27, pp. 275-286.

[6] M. A. Kaboudan, "Compumetric forecasting of crude oil prices," in *Proc. the IEEE Conference on Evolutionary Computation*, 2001, vol. 1, pp. 283-287.

[7] A. Hou and S. Suardi, "A nonparametric GARCH model of crude oil price return volatility," *Energy Economics*, vol. 34, no. 2, pp. 618-626, 2012.

[8] S. Ramyar and F. Kianfar, "Forecasting crude oil prices: A comparison between artificial neural networks and vector autoregressive models," *Computational Economics*, vol. 53, no. 2, pp. 743-761, 2017.

[9] M. Brigida, "The switching relationship between natural gas and crude oil prices," *Energy Economics*, vol. 43, pp. 48-55, 2014.

[10] S. Sun, Y. Wei, and S. Wang, "AdaBoost-LSTM ensemble learning for financial time series forecasting," in *Proc. International Conference on Computational Science*, 2018, ch. 55, pp. 590-597.

[11] G. Kumar, S. Jain, and U. P. Singh, "Stock market forecasting using computational intelligence: A survey," *Archives of Computational Methods in Engineering*, vol. 28, no. 3, pp. 1069-1101, 2020.

[12] H. Rezaei, H. Faaljou, and G. Mansourfar, "Stock price prediction using deep learning and frequency decomposition," *Expert Systems with Applications*, vol. 169, 2021.

[13] M. U. Gudelek, S. A. Boluk, and A. M. Ozbayoglu, "A deep learning based stock trading model with 2-D CNN trend detection," in *Proc. IEEE Symposium Series on Computational Intelligence*, 2018, pp. 1-8.

[14] A. Tsantekidis, N. Passalis, A. Tefas, *et al.*, "Forecasting stock prices from the limit order book using convolutional neural networks," presented at the IEEE 19th Conference on Business Informatics, 2017.

[15] J. Lee, R. Kim, Y. Koh, *et al.*, "Global stock market prediction based on stock chart images using deep Q-network," *IEEE Access*, vol. 7, pp. 167260-167277, 2019.

[16] A. Thakkar and K. Chaudhari, "A comprehensive survey on deep neural networks for stock market: The need, challenges, and future directions," *Expert Systems with Applications*, vol. 177, 2021.

[17] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735-1780, 1997.

[18] Z. Li and V. Tam, "A comparative study of a recurrent neural network and support vector machine for predicting price movements of stocks of different volatilites," in *Proc. IEEE Symposium Series on Computational Intelligence*, 2017, pp. 1-8.

[19] L. C. Cheng, Y. H. Huang, and M. E. Wu, "Applied attention-based LSTM neural networks in stock prediction," in *Proc. IEEE International Conference on Big Data*, 2018, pp. 4716-4718.
[20] X. Pang, Y. Zhou, P. Wang, *et al.*, "An innovative neural network approach for stock market prediction," *The Journal of Supercomputing*, vol. 76, no. 3, pp. 2098-2118, 2018.
[21] T. Fischer and C. Krauss, "Deep learning with long short-term memory networks for financial market predictions," *European Journal of Operational Research*, vol. 270, no. 2, pp. 654-669, 2018.
[22] NY Mercantile. Crude Oil (WTI). [Online]. Available: https://finance.yahoo.com/quote/CL=F
[23] C. C. Aggarwal, *Neural Networks and Deep Learning*, Springer, 2018.
[24] G. Kumar, U. P. Singh, and S. Jain, "Hybrid evolutionary intelligent system and hybrid time series econometric model for stock price forecasting," *International Journal of Intelligent Systems*, vol. 36, no. 9, pp. 4902-4935, 2021.
[25] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proc. the 32nd International Conference on Machine Learning*, 2015.
[26] P. Virtanen, R. Gommers, T. E. Oliphant, *et al.*, "SciPy 1.0: Fundamental algorithms for scientific computing in Python," *Nature Methods*, vol. 17, pp. 261-272, 2020.

Wisaroot Lertthaweedech received the B.Sc. degree in Chemical Engineering, the M.Eng. degree in Petroleum Engineering and the M.Sc. degree in Computer Science from Chulalongkorn University in 2012, 2016 and 2022, respectively. He is currently working as a Petroleum Engineer at Chevron Thailand Exploration and Production Company Limited, Thailand. His research interests include artificial intelligence and machine learning.

Pittipol Kantavat received the B.Eng., M.Eng. and Ph.D. degrees in computer engineering from Chulalongkorn University, Thailand, in 2004, 2008 and 2018, respectively. He is currently a Lecturer at the Department of Computer Engineering, Chulalongkorn University, Thailand. His research interests include artificial intelligence, pattern recognition and machine learning.

Boonserm Kijsirikul received the B.Eng. degree in Electronic and Electrical Engineering, the M.Sc. degree in computer science, and the Ph.D. in computer science from Tokyo Institute of Technology, Japan, in 1986, 1990, and 1993, respectively. He is currently a Professor at the Department of Computer Engineering, Chulalongkorn University, Thailand. His current research interests include Machine Learning, Artificial Intelligence, Natural Language Processing, and Speech Recognition.