Applying dual attention deep learning model to predict oil futures prices

Hsin-Chun Yen
National Kaohsiung University of Science and Technology
Wen-Chen Huang (wenh@nkust.edu.tw)
National Kaohsiung University of Science and Technology https://orcid.org/0000-0001-9389-8865

Research Article

Keywords: Dual Attention, CNNBiLSTM, CNNBiGRU, RMSE, deep learning, sliding window, Prophet, TPA-LSTM, ARIMA

Posted Date: October 4th, 2022

DOI: https://doi.org/10.21203/rs.3.rs-1136379/v1

Applying dual attention deep learning model to predict oil futures prices

Hsin-Chun Yeh and Wen-Chen Huang

Sam0527@outlook.com and wenh@nkust.edu.tw

Department of Information Management

National Kaohsiung University of Science and Technology

Kaohsiung, Taiwan

Abstract

Due to the non-linear and complex characteristics of oil futures prices, they are easily affected by external factors. Therefore, predicting the price of oil futures is a very challenging topic in deep learning time series, but the existing literature lacks research on introducing relevant features. Therefore, the purpose of this paper is to use a two-way attention mechanism to predict oil futures prices and import gold prices as features.

The data source is taken from Yahoo Finance, providing 5146 items from 2000/02/28 to 2020/06/29. The paper uses CNNBiLSTM, CNNBiGRU models and Attention CNNBiLSTM, Attention CNNBiGRU, Dual Attention CNNBiGRU, Dual Attention CNNBiLSTM and TPA-LSTM, Prophet, ARIMA that increase the attention mechanism to conduct experiments to find the best model. Moreover, explore the sliding window review days to deal with two-dimensional time series data and the influence of the delay of the typical days to predict the experiment's impact.

The experimental results show that increasing the number of review days will decrease the prediction accuracy. The new model is imported using the two-dimensional time series method, and the attention mechanism is integrated. Importing CNNBiGRU with a dual attention mechanism from 80% of the training

items and 20% of the test items in the experimental data set can improve the overall prediction accuracy. For example, the model BiGRU has improved the RMSE from 2.79 to 1.85 by adding a dual attention mechanism. In addition, it is found that if the regression model has large fluctuations in the prediction data, it will increase the RMSE and MAE, especially in the experimental items of 96% training and 4% testing data sets.

Keywords: Dual Attention, CNNBiLSTM, CNNBiGRU, RMSE, deep learning, sliding window, Prophet, TPA-LSTM, ARIMA

I. Introduction

Facebook proposes Prophet [1], which uses supervised learning through complex neural networks to allow the model to learn influential learning trends quickly and predict short-term trends based on time, month, week, and a quarter. This research aims to use oil futures price as the forecast target and gold price setting as the feature. Process the data set through the sliding window, and explore whether the 1-20 review days can improve the accuracy of neural network learning. Google proposed the Encode Decode Attention [2] method, and the result was better than the previous neural network. In this study, the Attention mechanism is added to improve the neural network long and short-term memory (LSTM) and GRU models through a weighting method different from Google's approach.

Here are some goals:

1. Can the use of supervised learning improve the long-standing problem of prediction accuracy?

2. Will the delayed input of two-dimensional data affect the predicted value of the model?

3. Will the accuracy of using a two-dimensional model with an attention mechanism effectively improve the prediction value?

4. Will the number of days in the sliding window affect the performance of deep learning?

This research uses TPA-LSTM [4], Prophet, ARIMA [3] to conduct module testing. Furthermore, put forward the model test of CNN BiLSTM Attention, CNN BiGRU Attention, CNN BiGRU Dual Attention CNN BiLSTM Dual Attention.

The contributions of this research are as follows:

1. Use CNNBiLSTM and CNNBiGRU models to obtain 2.17 RMSE in 90% training and 10% testing data sets.

2. After importing two-dimensional data and delaying the number of days of the gold price, the RMSE dropped from 3.33 to 3.07.

3. Using a two-dimensional model with an attention mechanism, the RMSE decreased from 2.79 to 1.85.

4. The number of days in the sliding window affects the accuracy of learning prediction. As the number of days increases, the accuracy decreases.

After CNN BiGRU joins the attention mechanism, the model architecture diagram is as follows:

Figure 1 single attention CNNBIGRU

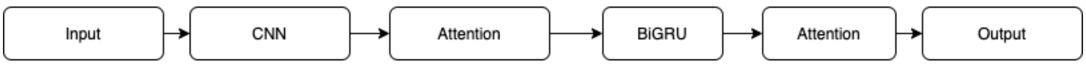

Figure 2 Dual Attention with CNNBiGRU

Figure 1 only uses an attention mechanism to enhance the final output. Figure 2 shows that after placing the attention mechanism on CNN, the short-term features extracted by CNN are enhanced again. After training through BiGRU, it is passed to the attention mechanism and output again. Subsequent experiments will be conducted with BiGRU/BiLSTM on accuracy, review days, and feature delays to find the best neural network model.

This study uses the data set provided by Yahoo, with a time series of 5146 oil futures and gold prices ranging from 2000/02/28 to 2020/06/29. The noise is not processed, and the response learning of the model to the shock is added, which is applied to TPA LSTM and four neural network models proposed in this research for testing.

2. Literature Review

The first section will briefly describe the regression model and the literature in recent years. The second section will discuss in depth the methods of various neural networks. Finally, the third section discusses the related literature on the relationship between gold and crude oil in two-dimensional data.

2.1 Regression model

Forecasting time series data is an essential topic in economics, business, and finance. The most famous models for linear univariate time series forecasting are Autoregressive Integrated Moving Average (ARIMA) [3], Autoregressive Regression (A.R.) [5], Moving Average (M.A.) [6], and Autoregressive Moving Average (ARMA) [7]. Kazem, Ahmad, et al. proposed a new direction using linear vector regression (SVR) [8]. The optimized SVR is used to predict stock market prices. SVR is more sensitive to parameter selection and is good at handling binary classification problems. However, most of these models are limited to linear processing. A new ANFIS method [9] is proposed to predict stock market prices, and it became one of the alternative methods used by most researchers. According to the test results in the literature, although the data results are pretty excellent, it still discards many relevant data that can be utilized as data features. If combining relevant elements, the accuracy of prediction may be affected.

2.2 Neural network

Using a neural network to predict stock price attributes can achieve good prediction results and outperform regression and other models [10,11]. They used standardized data and imported genetic algorithm (G.A.) to adjust the parameters, resulting in higher accuracy. However, the relevant features are discarded in the test, and the model is a single model, which may not be able to deal with different data effectively. Gilliland et al. [12] mentioned how neural networks are applied to business forecasts and solve actual cases. Then Prophet is proposed to build a large-scale prediction algorithm [13], which combines critical cycles such as weeks, months, and quarters. Under the neural network with fast learning ability, the actual used in Facebook advertising traffic analysis and forecasting. LSTM and Prophet are used to predict stock prices [14]. The performance of LSTM is significantly better than other regression techniques. Because Prophet algorithm's limitation, the prediction will gradually become inaccurate like a regression model for a long time.

Selvin et al. [15] use three different deep learning architectures. For comparison and evaluation of its performance, the sliding window method is used to predict future prices. It is evident from the results that the CNN architecture can identify trend changes, so CNN is considered the best model. The CNN model primarily uses the information at a specific moment to make predictions. However, as the data becomes complex, the model still has room for adjustment, such as adding other neural network layers. Althelaya et al. [16] compared the performance of BLSTM and SLSTM models with lightweight neural networks and unidirectional LSTM. The results show that both BLSTM and stacked LSTM networks have a better understanding of predicting short-term prices. Therefore, it is convinced that the deep learning method is better than the lightweight neural network.

In addition to the in-depth development of traditional neural networks, some scholars are still developing new models and combining other models to improve learning accuracy. Kim and Won [17] created a model. This model combines the following three models GARCH, EGARCH, and EWMA with LSTM, and calls this model GEW-LSTM, which provides our idea and practice of a hybrid neural network. Atsalakis and Valavanis [18] compares LSTM with ARIMA. The average error rate obtained by LSTM is reduced by 84-87%, which shows that LSTM is superior to ARIMA and exhibits genuinely random behavior. Still, it only imports the data and does not incorporate relevant data, such as (VAR) regression model characteristics. Sachdeva et al. [19] combined statistical methods, introduced LSTM to the volume-weighted average price, and developed a simplified transaction strategy. Nelson et al. [20] use LSTM and stock prices to make sequence predictions. Moreover, change the past to put the closing price and add technical indicators to improve the forecasting problem. The results show that when predicting whether the price of a particular stock will rise shortly, the average accuracy rate reaches 55.9%. The key to getting better prediction results is improving the input or modifying the new direction of the model. Xingjian et al. [21] use Convolutional LSTM for rainfall forecast training, import LSTM into CNN image recognition, and combine the advantages of LSTM. Shih et al. [22] cut the data into fragments, obtain the current segment and another component through a neural network to calculate variables, and use attention to select the most significant variable for learning. The difference from the model proposed in this research is that the proposed model uses the Softmax function to calculate the current time step and features to calculate the weight instead of extracting fragments.

2.3 The relationship between oil and gold

International crude oil prices have experienced drastic fluctuations. First, global crude oil prices continue to rise and reach record highs. Then, the subsequent encounter with the new coronavirus has led to a decrease in overall demand. In addition, due to the global epidemic, various industries have been shut down, and the total loss is difficult to estimate. As a result, U.S. oil prices have experienced a historic plunge. Since crude oil is a significant indicator of economic development, the high volatility of crude oil prices has brought a massive impact on economic growth, especially for those crude oil-importing countries. As a result, the crude oil pricing mechanism has become a hot topic in academia. In the research of [23], they observed and analyzed the relationship between gold and inflation and found that gold and oil can predict each other, and the two will affect each other.

Rising oil prices have led to an increase in inflation, which also means an increase in the price of gold. Chkili [24] uses the GARCH (FIGARCH) model to analyze the dynamic relationship between the gold and crude oil markets. Experiments prove that the dynamic relationship between the two markets will change over time. Ftiti et al. [25] analyzed and compared oil and gold in different needs. Under the preventive demand shock of oil, the response of gold price is lagging behind the fluctuation of oil price. However, the effect is short-lived, lasting only 16 to 24 days. Therefore, the standard linkage and causality between the two markets are more prominent. Previous research has shown that the price of oil affects the price of gold through multiple channels.

3. Research method and design

This research conducts experiments through deep learning predictions. First, analyze the two-dimensional sequence to import multiple models, add attention mechanism, feature days delay, and slide window to look back days to see if it can improve the accuracy. The practical steps are as follows:

- Design and test method: Use multiple models to compare with TPA-

  LSTM, Prophet, ARIMA.

- Data processing: Pre-processing the data sets and then normalizing of data sets.
- Model building: Establish a variety of models BiLSTM/BiGRU, add attention mechanism to compare.
- Research experiment items: Design five experiments and review their results.

3.1 Design and test method

By combining different models, such as Attention CNN BiLSTM, Attention CNN BiGRU, Dual Attention CNN BiLSTM, Dual Attention CNN BiGRU, enhancing a single model, compares these models with TPA-LSTM. Find out which model is the best performer, and look for the relevant influence of the number of review days.

3.2 Data set pre-processing and normalization

The data set selects oil futures (CL = F) and gold prices provided by Yahoo for processing and uniformly normalizes the data.

3.3 Model building

This model is built by combining CNN and two-way LSTM/GRU. CNN extracts short-term features, and two-way LSTM/GRU can obtain better learning through calculation than in the past one-way. Furthermore, integrate the attention mechanism to enhance the feature effect through weighted multiplication. The model could be designed to have one or two attention mechanisms.

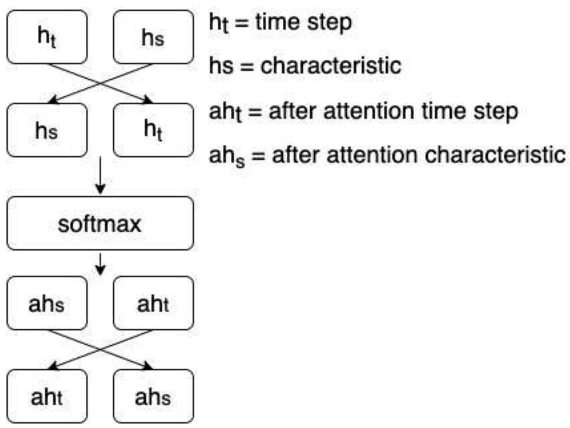

Figure 3 attention mechanism

Figure 3 uses CNN to extract short-term characteristic neurons and pass them to the lower layer to exchange the current h_t and h_s dimensions. Then through the lower layer, use the Softmax function to convert the output value into a relative probability. The sum of all the probabilities will be equal to 1, which is passed to the lower layer. The original neuron and the neuron after the attention mechanism will be matrix multiplied one by one. At this time, it increases the weight of the neuron. This action is called the attention mechanism, which can remember the vital part. In the follow-up experiment, we add a dual attention mechanism behind CNN. The other attention mechanism is behind the two-way GRI/LSTM.

3.3.1 Bidirectional layer:

Schuster and Paliwal [26] Build a reverse neural network. One is from the past to the future, and the other never comes to the past. The backward pass network retains future messages, and multiple hidden states will be used to combine so that this combination can keep history and future information. This method is widely used in language learning and added to subsequent experimental models. The bidirectional model is referred to as BiLSTM/BiGRU for short.

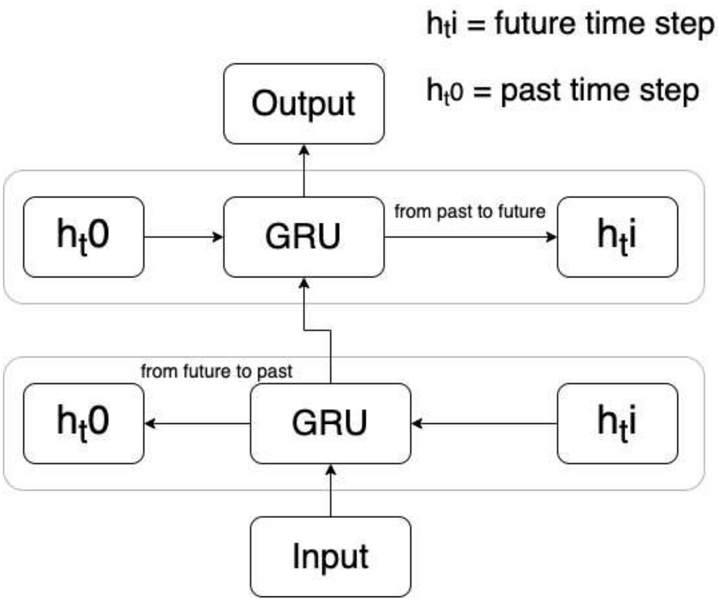

Figure 4 bi-directional GRU

Figure 4 shows the bidirectional mechanism used by the GRU layer. h_t^i to h_t^o represents the reverse network, h_t^o to h_t^i represents forward order. This structure can provide complete information about the past and future of the output layer. Subsequent experiments will show the combination of bidirectional GRU and LSTM models.

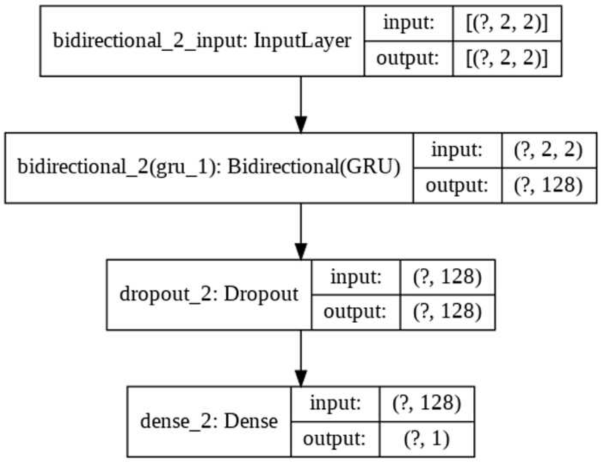

Figure 5 BIGRU: bidirectional GRU

The following figures, "?" represent the size of the first dimension of the model input because the first layer establishes a fully connected layer to accept complete data.

Figure 5 uses a two-way GRU with a total of 4 layers. First, use Bidirectional to wrap the GRU layer, making it a two-way GRU, or "BiGRU" for short. Next, the lower layer selects the Dropout layer to accelerate convergence and avoid coupling. Finally, the output layer is established. The model steps are as follows.

1. Use Bidirectional to package GRU to achieve a two-way GRU model.
2. Add Dropout to accelerate model convergence.
3. The output layer calls the sigmoid function for training. Experimental results show that calling this function improves accuracy.

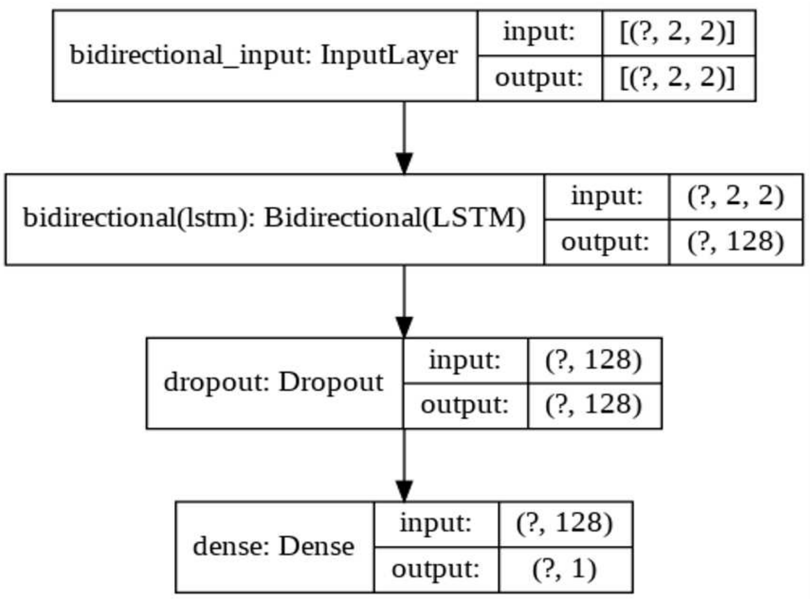

Figure 6 BiLSTM: Bidirectional LSTM

Figure 6 uses a two-way LSTM to receive input through the input layer and pass it to the two-way LSTM for training and reverse transmission. In the end, only one layer of Dropout is added to accelerate convergence without using redundant Flatten, Batch Normalization, etc.

The model steps are as follows.

1. Use Bidirectional to wrap LSTM to achieve a bidirectional LSTM model.
2. Add Dropout to accelerate model convergence.
3. The output layer calls the sigmoid function for training. The experimental results call this function to improve the accuracy.

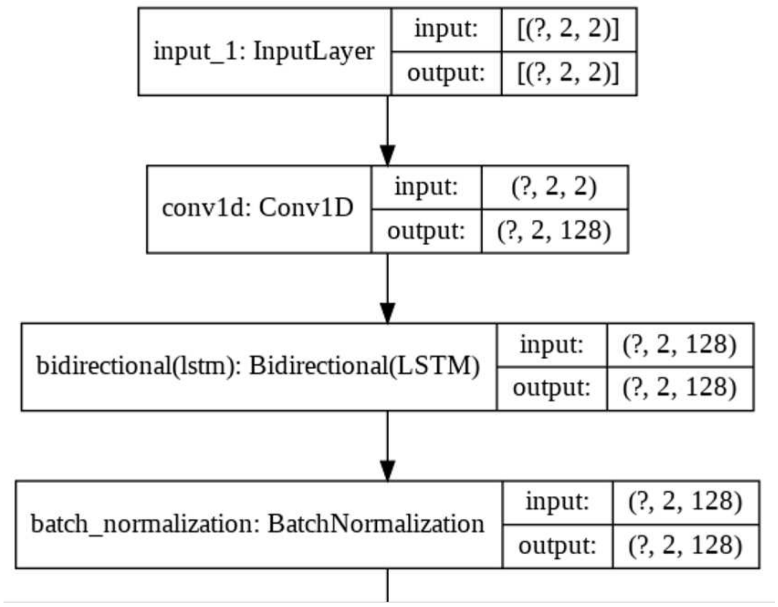

Figure 7 Attention CNNBiLSTM part 1

Figure 7 shows the first part of the Attention CNNBiLSTM model schema. The first layer establishes a fully connected layer to accept complete data. The neurons with short-term features are acquired through Conv1D (CNN) and then passed to the lower bidirectional LSTM for processing. Use bidirectional characteristics to learn neurons with complete information. The receiver calls the Batch normalization layer to speed up the training to prevent the gradient from disappearing and assist the model in converging.

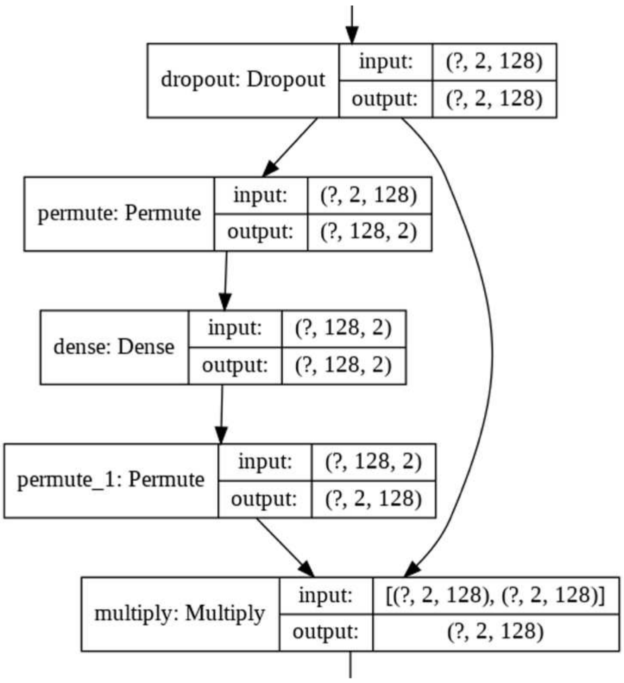

Figure 8 CNNBiLSTM part 2

Figure 8 constructs the attention mechanism of Figure 3. From the Permute layer, we can find the behavior of matrix dimension exchange in this layer. A dense layer is called a function for weight distribution. Finally, the Multiply layer is called to complete the matrix multiplication of the neurons after the weight distribution and the neurons that have not yet passed the attention mechanism to increase the weight of the essential neurons.

Algorithm 1 Pseudocode for attention mechanism

```
Input cell state,
Output Atention_multiply cell
attention_function ( input cell state from the upper layer cell state)
TimeSteps ← get size then sent second dimension size from backend
a ← swap the first and second dimensions of the input.
a ← use connection layer to receive upper layer a and set TimeSteps for
        the cell then call softmax function
```

5	a_probs ← swap the first and second dimensions of the a.
6	Attention_multiply←call input cell state and a_probs for matrix multiplication
7	Output Atention_multiply cell

Algorithm 1 shows how the attention mechanism is implemented in the model. First, row number 2 gets the tensor matrix size from the back end and assigns the second dimension to Timestep. Then row number 3 receives the first dimension, and the second dimension of the neuron is passed from the upper layer to exchange the sizes and name this layer a. The number of rows 4 defines the parameters of the connection layer, and the number of neurons is brought in and set to Timestep and called the softmax function to calculate. Finally, the number of rows 5 receives the output of the upper layer and exchanges the dimensions of the first dimension and the second dimension to restore the matrix size of the original data and multiply the initial input with the matrix to enhance attention.

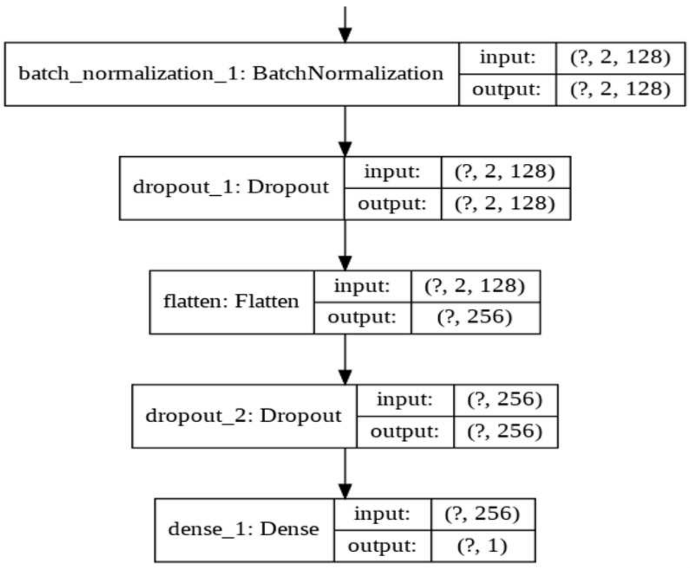

Figure 9 Attention with CNNBiLSTM part 3

Figure 9 shows the third part of the model, which uses the Dropout layer and batch normalization layer to improve the disappearance of gradients and the failure of the model to converge. Afterward, the filter layer is used to help the model reduce the data dimension.

After obtaining the neurons with short-term features through Conv1D (CNN), they are passed to the lower-layer bidirectional LSTM for processing. The bidirectional features are used to learn neurons with complete information. Next, the receiver calls the Batch normalization layer to speed up the training to prevent the gradient from disappearing and assist the model in converging. Then after constructing the attention mechanism, the behavior of matrix dimension exchange in this layer is discovered from the Permute layer. The dense layer is called the function for weight distribution. Finally, the Multiply layer is called to complete the matrix multiplication of the neuron after assigning the weight and the neuron that has not yet passed the attention mechanism to increase the neuron's weight. The dropout layer and batch normalization layer are successively used to improve the problems such as the disappearance of gradient and the failure of the model to converge. Continue to use the filter layer to help the model to reduce the data dimension. Finally, the neurons with attention are matrix multiplied with the BiLSTM output, and then the filter layer is added to reduce the dimensionality of the data.

The Attention with CNNBiLSTM model summary operation is as follows:

1. Obtain short-term features through CNN.
2. Use two-way LSTM to accept the features delivered by CNN.
3. Increase Batch Normalization to accelerate training to prevent the gradient from disappearing.
4. Use attention mechanisms to enhance learning.
5. Use Batch Normalization again.
6. Increase the Flatten layer to reduce the output dimension.
7. In the output layer, calling the sigmoid function can improve the accuracy.

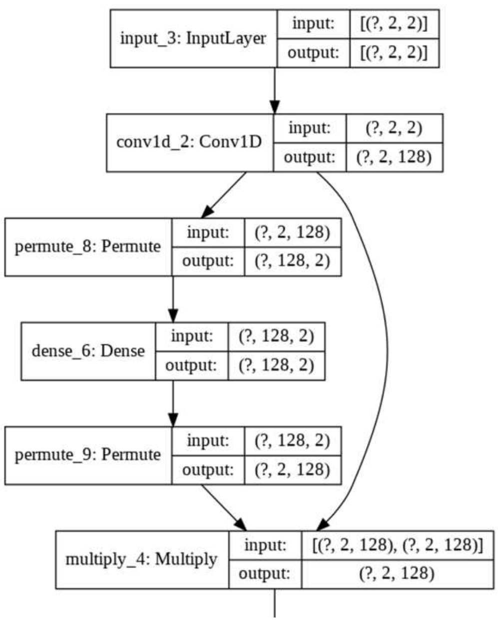

Figure 10 Dual Attention CNNBiGRU model part 1

Figure 10 shows the first part of Dual attention CNNBiGRU. After the short-term features are extracted through CNN, the attention mechanism is introduced to enhance the short-term features. Next, assign weights by inverting the matrix and calling the Softmax function in the Dense layer. Finally, the Multiply layer performs matrix multiplication to achieve the purpose of enhancing features.

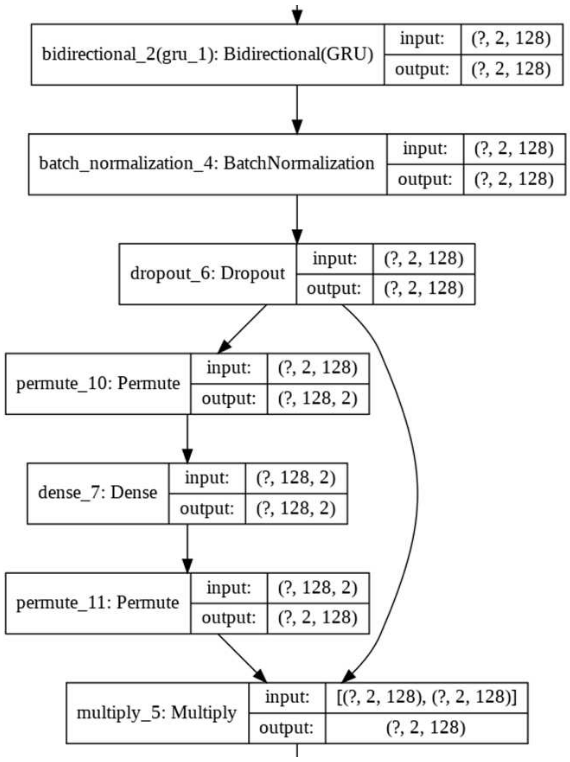

Figure 11 attention CNNBiGRU part 2

Figure 11 is a continuation of Figure 10, receiving the enhanced neuron features and importing BiGRU. BiGRU has the characteristics of remembering the past to the future and the future to the past. Train after obtaining complete information to enhance characteristic neurons. And add a Dropout layer in the lower layer to reduce the coupling problem, and then call the attention mechanism again after passing it down.

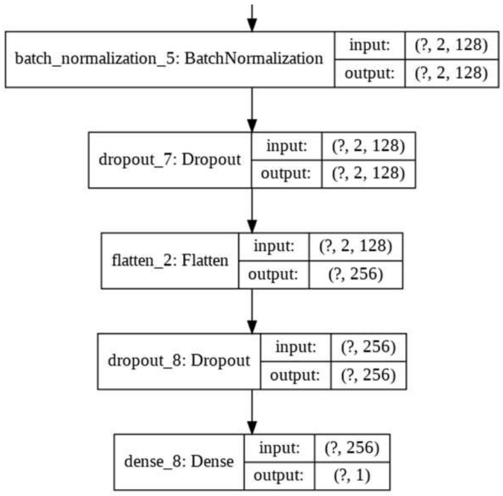

Figure 12 Dual attention CNNBiGRU part 3

Figure 12 continues Figure 11, calling the batch normalization layer to accelerate the convergence of the overall neural network and avoid excessive coupling. Next, cooperate with the Dropout layer, and finally, use Flatten to reduce the dimension to achieve the final output as a one-dimensional matrix.

Algorithm 2 Pseudocode for dual attention CNNBiGRU

```
Input Normalize data Output Predict data
Create DualAttention_with_CnnBiGRU_model()
input layer ← TimeSteps and dataset dims
Setting CNN's Conv1D layer then call 'relu.'
call attention_function
use Bidirectional GRU
Then add the BatchNormalization layer to prevent over coupling
add drop layer
call attention_function again
add BatchNormalization layer again
add drop layer
add flatten layer for down-sizing dimension
add drop layer
```

```
Dense ← use complete dense layer and call sigmoid
output ←Dense
```

In algorithm 2, after the short-term features are extracted through CNN, the attention mechanism is introduced to enhance the short-term features again. Next, assign weights by inverting the matrix and calling the Softmax function in the Dense layer. Finally, the Multiply layers perform matrix multiplication to achieve the purpose of enhancing features. Receive the enhanced neuron characteristics and import BiGRU. BiGRU has the parts of the past to the future and the future to the past. Obtain the aspects of complete information to train to enhance the typical neuron.
Moreover, add a Dropout layer in the lower layer to reduce the problem of coupling. After passing down, the attention mechanism is called again. Next, call the batch normalization layer to accelerate the convergence of the overall neural network and avoid excessive coupling. Finally, use Flatten to reduce the dimensionality to achieve the final output as a one-dimensional matrix.

The dual attention CNNGRU model operation is as follows:

1. Obtain short-term features through CNN.
2. Use attention mechanisms to enhance learning.
3. Use two-way GRU as the main body.
4. Use the Batch Normalization layer to accelerate convergence and reduce coupling.
5. Use the Dropout layer to reduce the model coupling.
6. Added attention mechanism again.
7. Use the Batch Normalization layer again to speed up training.
8. Increase the Flatten layer to reduce the output dimension.
9. Finally, use Dense to call the Sigmod function for calculation output.

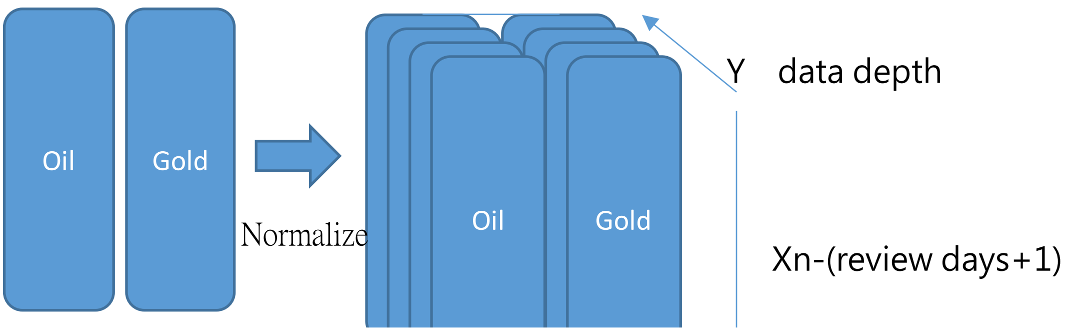

Figure 13 data structure with sliding windows

Figure 13 displays the entire sliding window process to describe the overall process. The final output is a three-dimensional matrix. The whole sliding window process is used in the processing of two-dimensional time-series data. This method allows the neural network to learn and predict better.

Pass the number of data items from the current *i* to *i* plus the number of days to look back and send to the second dimension. The number of data items will be reduced to increase the depth of the data matrix. The final total length will remain unchanged, and generating a three-dimensional matrix of data.

Assuming that the number of review days is 2, the day and yesterday are two days in total.

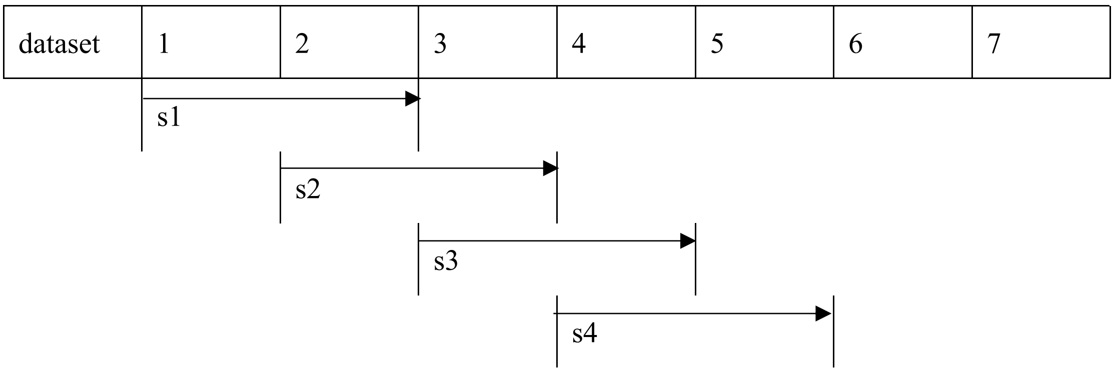

Figure 14 sliding window of the review day is 2

Figure 14 adds the number of review days to each S_n during the sliding window processing, and the length is unified to 2. When S_(n+1), remove the previous data. At this time, the range is S_(n+1) and S_(n+2). So let it learn two days of data at a time. This method can learn more than a single-day input, allowing the neural network to perform better on sequential data.

4. Experimental results and analysis

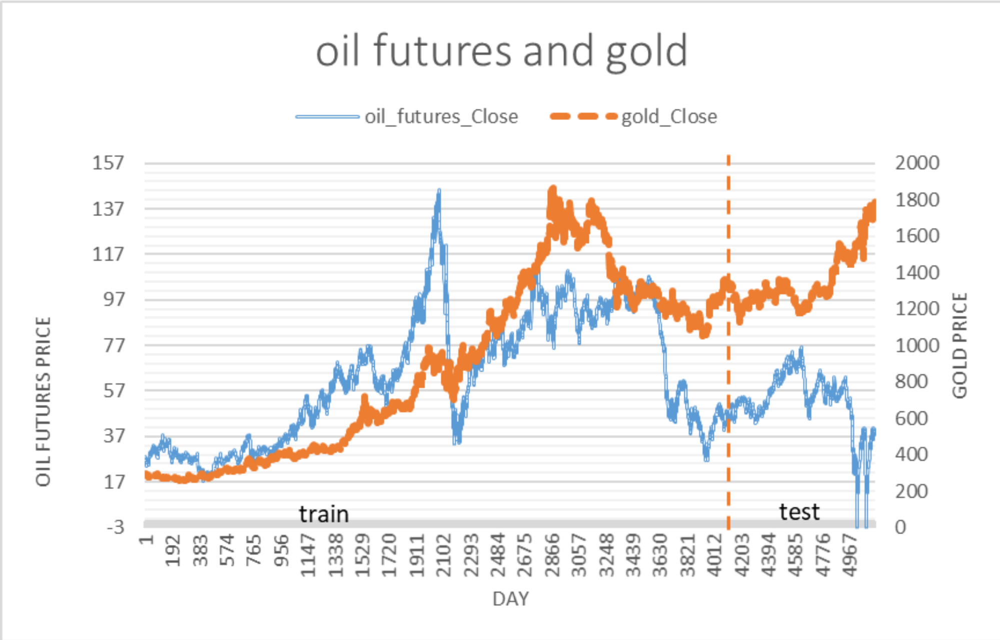

Figure 15 the overlapped diagram between the original trend of oil futures and gold prices

Figure 15 shows the overlapped diagram between the original trend of oil futures and gold prices. Using the data set provided by Yahoo, the length of the time series is from 2000/02/28 to 2020/06/29, a total of 5146 oil futures and gold prices. After importing the normalized and reduced data, use the sliding window method to study the predictions that affect the neural network for 1 to 20 days to find the best results.

Table 1 The accuracy of BiLSTM and BiGRU

Training/Testing		Add gold price	No gold price	Add gold price	No gold price
		BiLSTM		BiGRU	
"96/4"	RMSE	**2.82**	3.23	**2.12**	2.4
	MAE	1.98	2.01	1.31	1.6
"90/10"	RMSE	2.17	2.24	2.17	2.17
	MAE	1.38	1.4	1.38	1.24
"80/20"	RMSE	**2.47**	2.5	2.79	2.79
	MAE	1.52	1.5	1.95	1.83
"70/30"	RMSE	2.54	**1.83**	2.53	1.99
	MAE	1.65	1.06	1.64	1.1
"Average"	RMSE	2.50	2.45	2.40	**2.34**
	MAE	1.63	1.49	1.57	1.44

Table 1 shows that when the gold data is added or not, the accuracy of RMSE and MAE by BiLSTM and BiGRU with different sets of training/testing partitions. Overall, the best performance (RMSE 1.83) is from 70/30 training/testing combination with "no gold price" by BiLSTM.

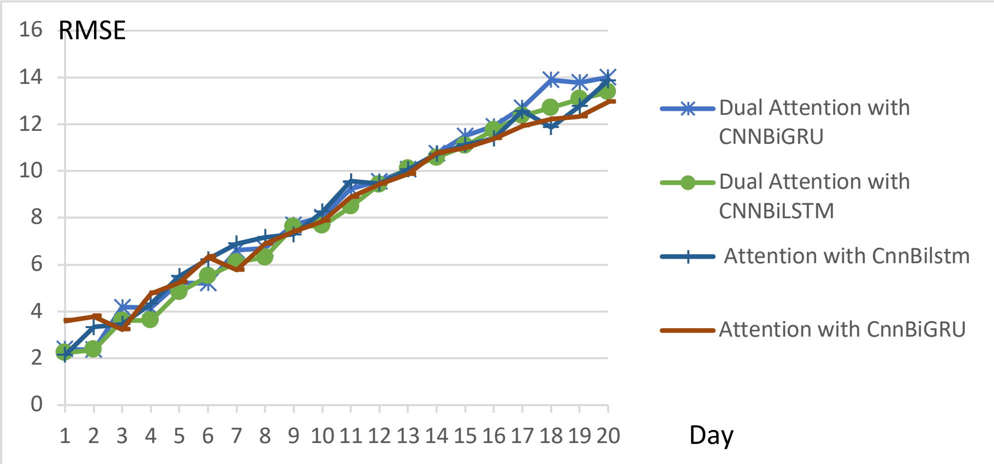

Figure 16 the accuracy from 4 models with different review days

Figure 16 shows the test results of the four models. It is found that the way the data set is divided will lead to the worse the prediction results as the longer the number of review days. In the study, the number of review days is 1-20 days, and it is observed that the best prediction results fall within 1-2 days according to different models. If it exceeds two days, it will cause a gradual decline in prediction accuracy regardless of which model.

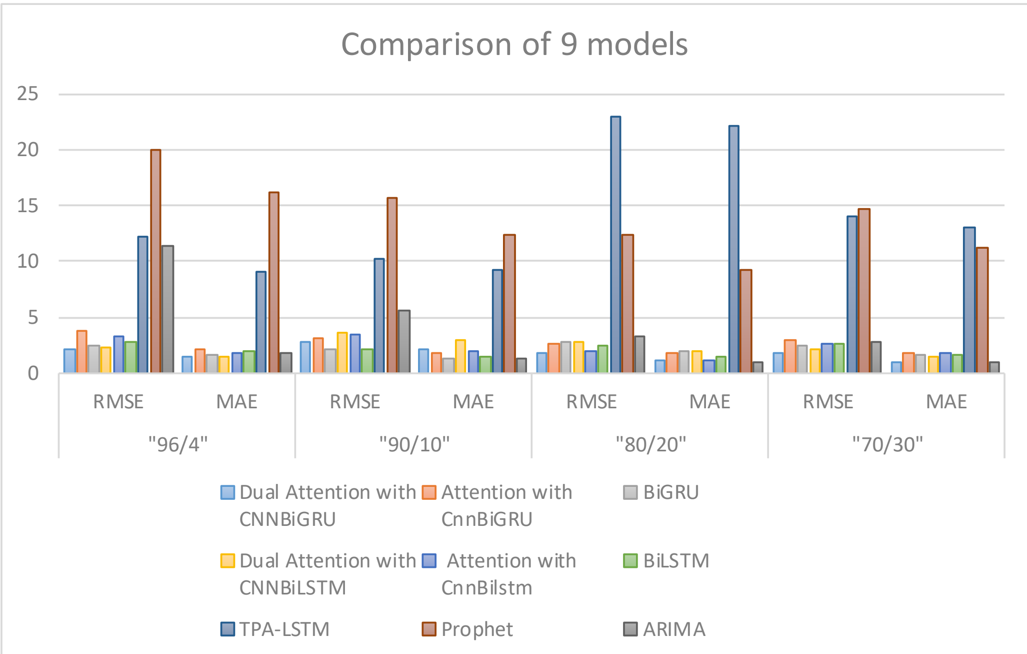

Figure 17 The accuracy among nine models

Figure 17 uses multiple models for comparison. The Dual Attention with CNNBiGRU and Attention with CNN BiLSTM models have improved accuracy through changes in the data set. The reason comes from the increase in the number of forecasts to improve the accuracy of the estimates. In addition, because the TPA-LSTM model uses critical variables, the TPA-LSTM model also has higher performance for long-term prediction. Among them, Dual Attention with CNNBiGRU is 0.94 (RMSE) lower than BiGRU.

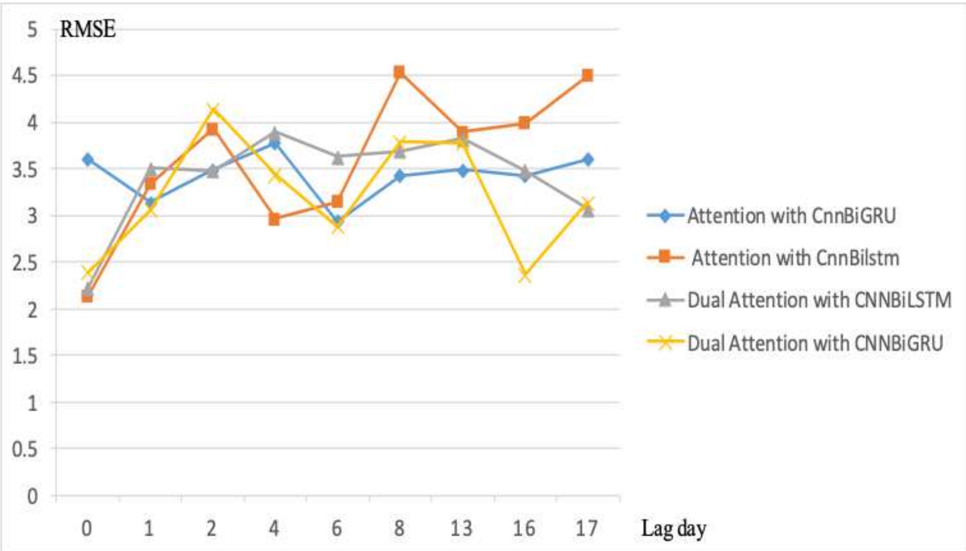

Figure 18 The accuracy of four models with various lag days

Figure 18 using four models to compare the RMSE calculation and obtain eight different days of delay parameters. Among the three kinds of Dual Attention with CNNBiGRU, Dual Attention with CNNBiLSTM, and Attention with CNNBiLSTM, the best prediction value is obtained in the number of days without delay. Furthermore, another model, Attention with CNNBiGRU, achieved the best accuracy on the sixth day. Thus, prove that the characteristic data can improve accuracy.

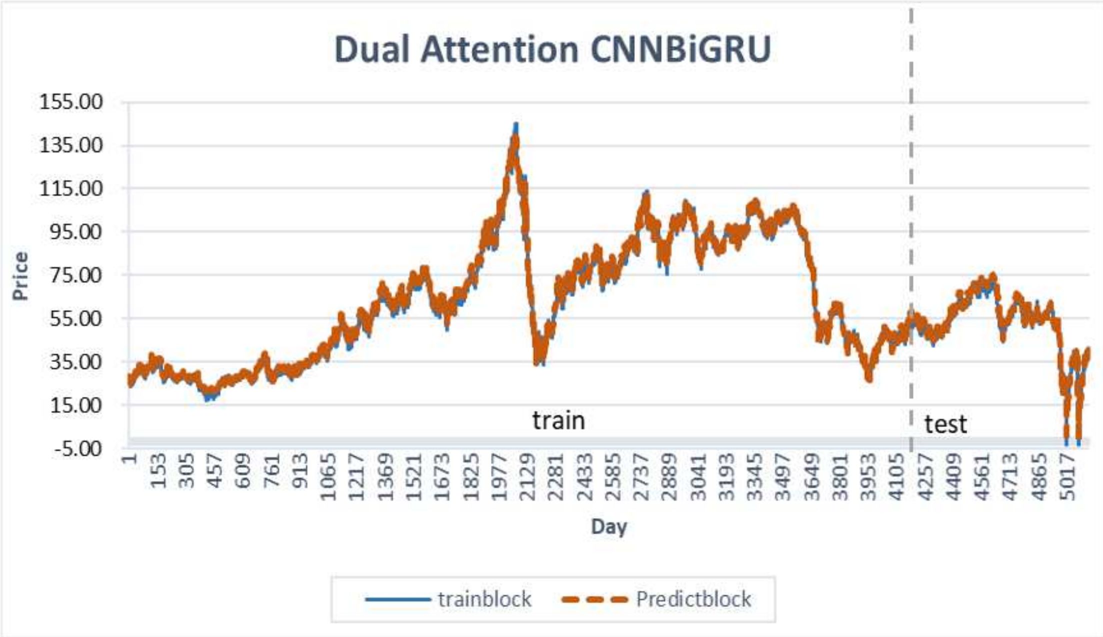

Figure 19 the predicted results of the Dual Attention with CNNBiGRU model under 80% training/20% testing data

Figure 19 shows the predicted results of the Dual Attention with CNNBiGRU model under 80% training/20% testing data, in which the training score is 1.89 (RMSE)/1.1 (MAE), and the testing score is 1.85 (RMSE)/1.06 (MAE). The blue line represents the actual data, and the brown line is the predicted result. The training period is from the first day to the 4105th day, and the testing period is from the 4257th to the 5107th day. The prediction gap has been entirely accurate results. The primary influence lies in the violent peak of volatility and the noise of oil futures approaching zero and negative values. The results show that this model is capable of reflecting and making accurate predictions for noise processing. The data in this research contains data on hostile oil prices during the epidemic, which leads to higher RMSE results. However, the model can still improve the accuracy by increasing the attention mechanism and the adjustment of CNN and increases the model prediction accuracy.

5. Conclusion

This paper proposes using a CNN bidirectional neural network with a dual attention mechanism and a sliding window to process data sets, improving the accuracy of neural network prediction. In the model experiment using GRU mixture, prediction accuracy is much improved than the original single model. Furthermore, it proves that the proposed Dual Attention CNNBiGRU model is better than the two-way GRU model and the traditional ARIMA and Prophet predictions. However, in the 94% training and 6% test data sets, if the prediction data is in large fluctuations, the regression model cannot accurately predict, promoting the gradual increase of RMSE and MAE. Among them, Prophet is suitable for use in cases where the number of predictions is small. In addition, the performance of the Dual Attention CNN BiLSTM model is worse than the simple two-way LSTM model. The reason is that the enhanced features make the model unable to learn long-term memory for the two-way LSTM.

This paper proposes an efficient neural network model and implements it. It uses CNN to extract short-term features and transfer them to the attention mechanism to enhance features. As a result, the RMSE of the original two-way GRU reduces from 2.79 to 1.85. As a result, a 2-day sliding window is used to process the data set. The GRU neural network integrates new features, delayed feature days, dual attention mechanism, and CNN to improve the model's accuracy and obtain the best RMSE compared to the two-way GRU and LSTM.

Finally, for model innovation, mixing, and ways to increase model depth, we will explore more algorithms for data cutting to enhance the existing sliding window cutting data problem.

Ethical Standards statements

Funding:
No funding is provided for the preparation of manuscript.

Ethical approval:
This article does not contain any studies with human participants or animals performed by any of the authors.

Conflict of interest:

The authors did not receive support from any organization for the submitted work. The authors have no conflicts of interest to declare that are relevant to the content of this article.

Authorship contributions

Hsin-Chun Yeh: Conceptualization, Methodology, Software, Data curation, Writing – original draft. **Wen-Chen Huang**: Supervision, Validation, Writing – review & editing.

Reference:

[1] Taylor SJ, Letham B. 2017. Forecasting at scale. PeerJ Preprints 5:e3190v2 https://doi.org/10.7287/peerj.preprints.3190v2

[2] Dong, Li, and Mirella Lapata. "Language to logical form with neural attention." arXiv preprint arXiv:1601.01280 (2016).

[3] Kalpakis, Konstantinos, Dhiral Gada, and Vasundhara Puttagunta. "Distance measures for effective clustering of ARIMA time-series." *Proceedings 2001 IEEE international conference on data mining*. IEEE, 2001.

[4] Ouyang, Hongbing, Xiaolu Wei, and Qiufeng Wu. "Discovery and prediction of stock index pattern via three-stage architecture of TICC, TPA-LSTM and multivariate LSTM-FCNs." *IEEE Access* 8 (2020): 123683-123700.

[5] Pierce, David A. "Least squares estimation in the regression model with autoregressive-moving average errors." *Biometrika* 58.2 (1971): 299-312.

[6] Hansun, Seng. "A new approach of moving average method in time series analysis." *2013 conference on new media studies (CoNMedia)*. IEEE, 2013.

[7] Benjamin, Michael A., Robert A. Rigby, and D. Mikis Stasinopoulos. "Generalized autoregressive moving average models." *Journal of the American Statistical association* 98.461 (2003): 214-223.

[8] Kazem, Ahmad, et al. "Support vector regression with chaos-based firefly algorithm for stock market price forecasting." *Applied soft computing* 13.2 (2013): 947-958.

[9] Boyacioglu, Melek Acar, and Derya Avci. "An adaptive network-based fuzzy inference system (ANFIS) for the prediction of stock market return: the case of the Istanbul stock exchange." *Expert Systems with Applications* 37.12 (2010): 7908-7912.

[10] Hiransha, M., et al. "NSE stock market prediction using deep-learning models." *Procedia computer science* 132 (2018): 1351-1362.

[11] Roondiwala, Murtaza, Harshal Patel, and Shraddha Varma. "Predicting stock prices using LSTM." *International Journal of Science and Research (IJSR)* 6.4 (2017): 1754-1756.

[12] Gilliland, Michael, Len Tashman, and Udo Sglavo. *Business forecasting: Practical problems and solutions*. John Wiley & Sons, 2016.
[13] Taylor, Sean J., and Benjamin Letham. "Forecasting at scale." *The American Statistician* 72.1 (2018): 37-45.
[14] Saldivar, Frank, and Mauricio Ortiz. "Stock Market Price Prediction Using Various Machine Learning Approaches."
[15] Selvin, Sreelekshmy, et al. "Stock price prediction using LSTM, RNN and CNN-sliding window model." *2017 international conference on advances in computing, communications and informatics (icacci)*. IEEE, 2017.
[16] Althelaya, Khaled A., El-Sayed M. El-Alfy, and Salahadin Mohammed. "Evaluation of bidirectional LSTM for short-and long-term stock market prediction." *2018 9th international conference on information and communication systems (ICICS)*. IEEE, 2018.
[17] Kim, Ha Young, and Chang Hyun Won. "Forecasting the volatility of stock price index: A hybrid model integrating LSTM with multiple GARCH-type models." *Expert Systems with Applications* 103 (2018): 25-37.
[18] Atsalakis, George S., and Kimon P. Valavanis. "Surveying stock market forecasting techniques–Part II: Soft computing methods." *Expert systems with applications* 36.3 (2009): 5932-5941.
[19] Sachdeva, Akshay, et al. "An Effective Time Series Analysis for Equity Market Prediction Using Deep Learning Model." *2019 International Conference on Data Science and Communication (IconDSC)*. IEEE, 2019.
[20] Nelson, David MQ, Adriano CM Pereira, and Renato A. de Oliveira. "Stock market's price movement prediction with LSTM neural networks." *2017 International joint conference on neural networks (IJCNN)*. IEEE, 2017.
[21] Shi, Xingjian, et al. "Convolutional LSTM network: A machine learning approach for precipitation nowcasting." *arXiv preprint arXiv:1506.04214* (2015).
[22] Shih, Shun-Yao, Fan-Keng Sun, and Hung-yi Lee. "Temporal pattern attention for multivariate time series forecasting." *Machine Learning* 108.8 (2019): 1421-1441.
[23] Narayan, Paresh Kumar, Seema Narayan, and Xinwei Zheng. "Gold and oil futures markets: Are markets efficient?." *Applied energy* 87.10 (2010): 3299-3303.
[24] Chkili, Walid. "Gold-oil prices co-movements and portfolio diversification implications." (2015).
[25] Ftiti, Zied, Ibrahim Fatnassi, and Aviral Kumar Tiwari. "Neoclassical finance, behavioral finance and noise traders: Assessment of gold–oil markets." *Finance Research Letters* 17 (2016): 33-40.
[26] Schuster, Mike, and Kuldip K. Paliwal. "Bidirectional recurrent neural networks." *IEEE transactions on Signal Processing* 45.11 (1997): 2673-2681.

Supplementary Files

This is a list of supplementary files associated with this preprint. Click to download.

- 1203dualattentationAuthor.pdf