Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2019.DOI

Precious Metal Price Prediction based on Deep Regularization Self-Attention Regression

JUNHAO ZHOU[1], ZHANHONG HE[2], YA NAN SONG[1], HAO WANG[3] (Member, IEEE), XIAOPING YANG[4], WENJUAN LIAN [5] AND HONG-NING DAI[1] (Senior Member, IEEE)

[1]Macau University of Science and Technology, Macau SAR (email: junhao_zhou@qq.com; ynsong@must.edu.mo; hndai@ieee.org).
[2]Department of Electronic and Computer Engineering, the Hong Kong University of Science and Technology, Hong Kong SAR. (email: calvinhzh@163.com)
[3]Department of Computer Science, Norwegian University of Science and Technology, Gjøvik, Norway (email: hawa@ntnu.no)
[4]Institute of Modern Economics and Management, Zhejiang Yuexiu University of Foreign Languages, China (email:20191025@zyufl.edu.cn)
[5]College of Computer Science and Engineering, Shandong University of Science and Technology, China (email: wenjuan.lian@126.com)

Corresponding author: Ya Nan Song (e-mail: ynsong@must.edu.mo).

ABSTRACT It is non-trivial to predict the prices of precious metals since a number of factors can affect the fluctuations of precious metal prices. Either parametric models or machine learning models cannot accurately forecast the precious metal prices. Though deep learning approaches show their strengths in extracting key features from complicated data, they have the limitations of learning localization and losing some temporal and spatial features. The recent advances in attention mechanisms bring the opportunities to overcome the limitation of deep learning models. In this paper, we originally propose a Regularization Self-Attention Regression Model for precious metal price prediction. In particular, the proposed RSAR model consists of convolutional neural network (CNN) component and Long Short-Term Memory Neural Networks (LSTM) component. Integrating with self-attention mechanism, this model can extract both spatial and temporal features from precious metal price data. Meanwhile, the proper configuration of regularization functions can also lead to the further performance improvement. Extensive experiments on realistic precious metal price dataset show that our proposed approach outperforms other conventional machine learning and deep learning methods.

INDEX TERMS Long short-term memory; Convolutional neural network; Attention Mechanism; Financial data analysis; Deep learning

I. INTRODUCTION

PRECIOUS metals typically have higher economic values while they are rare or difficult to be acquired. Historically, precious metals such as Silver and Gold have been used as currency equivalents (or money) while they have been mainly leveraged as financial and industrial commodities recently. It is non-trivial to predict their prices based on historical data since they are often influenced by a number of social-economic factors including production, circulation, industrial demands, sentiment of market.

We have also witnessed the rapid advances in machine learning and artificial intelligence. Meanwhile, the massive financial data becomes available. Consequently, economists and investors begin to employ machine learning (ML) methods to analyze the massive financial data and predict (a.k.a. forecast) the increment and decline trends of various financial factors. For example, Naïve Bayes algorithm [1] is one of the traditional statistic tools to help the financial industries to predict the tendency of financial products. In addition, the Capital Asset Pricing Model (CAPM) [2] describes the relationship between the expected return and risk of investing in a security. The Fama-French Three-factor Model [3] is an extension of CAPM via adding size risk and value risk factors in the market risk factor of CAPM. However, convention ML algorithms are struggling to process a large scale finance data with continuous features (e.g., historical gold prices). Even though many quantitative factor models have developed, these models cannot work especially considering some special events such as financial crises.

The recent advances in deep learning bring opportunities in extracting valuable information from massive financial and social-economic data. Deep learning (DL) is a broader family of machine learning methods based on multi-layer artificial neural networks. Among incumbent DL models, convolutional neural network (CNN) models show the advantages in learning complicated and hierarchical features of massive data [4]. Meanwhile, variants of deep learning models such as recurrent neural network (RNN) and Long Short-Term Memory Neural Networks (LSTM) also demonstrate the outstanding performance in dealing with time series data. Moreover, composite models consisting of two or more DL models show the superior performance than pure DL models. However, composite DL models also have their limitations like learning localization and losing temporal and spatial features (details to be explained in Section II).

The attention mechanism is essentially the best remedy to the limitations of composite DL models. The attention mechanisms can adaptively select learning regions from input data after calculating the attention probability distributions so as to highlight key features, consequently reducing disturbances from redundant information. As a result, the learning capability of DL models is greatly improved. Motivated by the advances in composite DL models and attention mechanisms, we propose and develop Regularization Self-Attention Regression Model (RSAR model) to predict daily precious metal prices.

In contrast to existing methods, the proposed RSAR model consists of both CNN component and LSTM component with regularization self-attention mechanism. LSTM component can extract the time series features from historical precious metal price data while CNN component is beneficial to learn the complex and hierarchical features. Moreover, Regularization Self-Attention mechanism can help to improve the learning performance through leveraging regularization functions. Our contributions of this paper are summarized as follows.

- We propose a novel deep learning model (RSAR model) to predict daily precious metal prices. In particular, we optimize the self-attention mechanism using regularization methods.
- We conduct extensive experiments for price forecasting on top of different realistic precious metal price datasets (including gold-price dataset and palladium-price dataset). The experimental results show that the proposed model outperforms other conventional ML and DL models.
- We also evaluate the impact of different parameters on our RSAR model. The major parameters include the size of window length, the number of LSTM layers and the number of CNN filters. We further show that there exists a trade-off between the number of CNN filters and the performance.

The remainder of this paper is organized as follows. Section II reviews related works on convention machine learning models and deep learning models. In Section III, we describe the overview of our architecture and the main proposed approaches in details. Then, the experiments evaluation and results are discussed in Section IV. Finally, we conclude our work and outline future research directions in Section V.

II. RELATED WORK

This section reviews recent advances on financial analysis, especially on precious metal price prediction. We roughly divide recent studies into two categories: machine learning approaches and deep learning approaches.

A. MACHINE LEARNING APPROACHES

Convention machine learning methods have been widely used in econometrics or statistics. Parametric models, such as Autoregressive Integrated Moving Average (ARIMA) and Autoregressive Conditional Heteroskedasticity (ARCH), have been used in various financial sectors. In particular, both the work of [5] and [6] applied ARIMA model in making prediction on banking stock market data and Chinese manufacturing industry. Yunus et al. [7] use ARIMA model to capture time correlation and offer possibility distribution of collection records for determined wind-pace time. Vaccaro et al. [8] suggest ARIMA model in hybrid architecture for electricity price forecasting. The work of [9] adopts ARIMA model for forecasting in Amman Stock Exchange. Moreover, ARIMA was used in daily gold-price prediction analysis in [10]. However, ARIMA and its alternatives have the following limitations: **1)** it is extremely time-consuming to make prediction due to the huge time consumption in reading and processing input data; **2)** it cannot achieve reasonable convergence in the long term forecasting task, consequently resulting in processing slow processing speed.

Besides ARIMA, ARCH and their alternatives, other machine learning (ML) based approaches, such as Support Vector Regression (SVR), Deep Regression (DR) and Logistic Regression (LR) have been applied in the finance field. For example, the work of [11] is based on Back Propagation Neural Network (BPN) to analyse stock market data. Zaidi et al. [12] adopted logistic regression in machine learning models. The work of [13] was conducted on support vector-machines (SVM). The core of SVM [14] methods is to use linear model to implement non-linear class boundaries and make classification in selecting stocks. However, most of ML approaches are suffering from low prediction accuracy and substantial efforts in data preprocessing.

B. DEEP LEARNING APPROACHES

In contrast to conventional ML approaches, deep learning (DL) approaches have the merits in capturing the complicated features from massive data [15]. For example, DL approaches have been used in sentimental analysis [16], electricity-theft detection [17] and traffic flow prediction [18].

Instead of a single DL model, which can only capture partial features from data, a composite model consisting of two or more DL models has the advantages in extracting various features from data. Therefore, composite DL approaches have recent extensive attention recently. Generally, composite DL models can be categorized into two types: parallel-composite models and serially-composite models. For example, the work of [19] designed a parallel-composite model (namely LSTM-CNN), which is a concatenation of the output of CNN and the input of LSTM. The work of [17] proposed a wide and deep model to learn from the time-series electricity consumption data to predict the electricity theft. However, both parallel-composite models and serially-composite models suffer have the limitations like learning localization (i.e., failing to prioritize each sub model when learning different features of data) [20], [21]. The main reason lies the decreased learning capability of each neuron (i.e., a basic unit in neural networks) when massive time-series data are fed into composite DL models persistently. Meanwhile, both temporal and spatial features in composite DL models are also lost.

To tackle this deficiency, attention mechanisms have been proposed [22]. Attention mechanisms can essentially overcome the limitations of composite DL models through adaptively selecting regions from input data after calculating the attention probability distribution. Since this mechanism can focus on the main features, redundant information can be greatly reduced. The attention mechanisms have demonstrated their effectiveness in composite DL models. As in [23], the authors use the attention mechanism for textual entity extraction. Furthermore, the work of [24] combines the attention mechanism with Bidirectional-LSTM (Bi-LSTM) to establish a Chinese part-of-speech tagging model, which achieves much higher accuracy than the traditional RNN methods.

Motivated by recent advances in composite DL models and attention mechanisms, we propose a RSAR combination model to conduct the prediction analysis for precious metal price. Compared with convention machine learning methods, our proposed framework has advantage in capturing the various features from the massive financial data via the LSTM and CNN components. Moreover, we optimize the regularization methods for the self-attention mechanism in the proposed model. Therefore, compared with convention DL models, our RSAR model can improve the learning efficiency via the regularization self-attention mechanism. Finally, our RSAR model is efficient to reduce the prediction error by concentrating on key weights and extracting the local features from the new generating sequence.

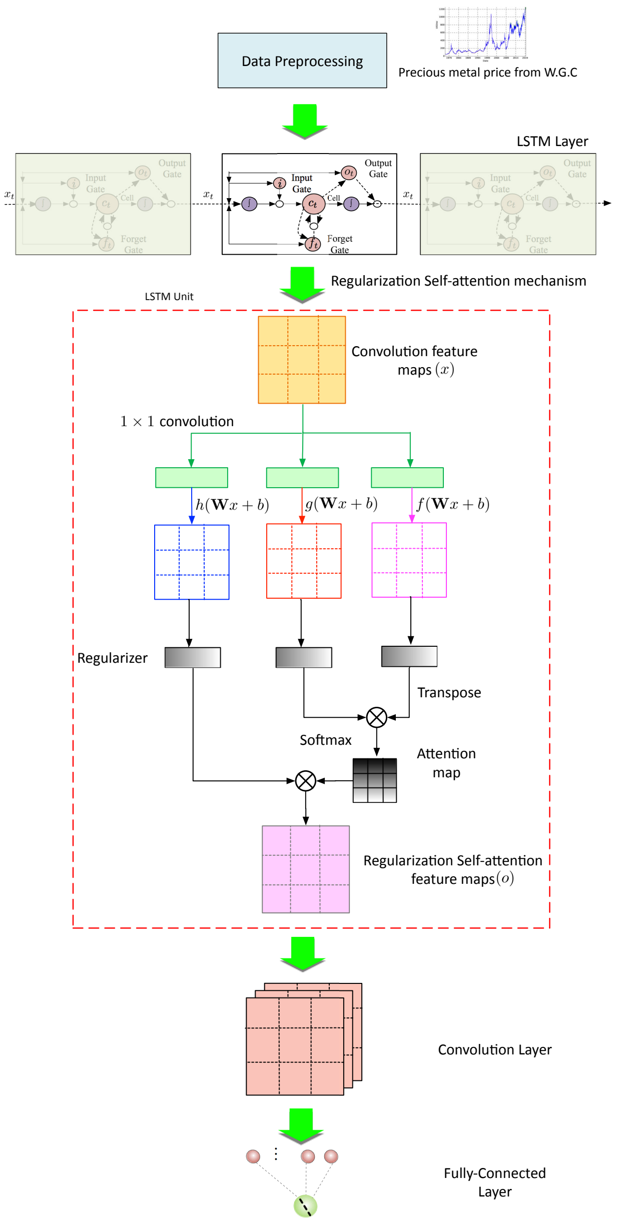

FIGURE 1. Regularization self-attention regression framework

III. OUR APPROACH

This section presents the main approach for precious metal price prediction.

A. OVERVIEW OF ARCHITECTURE

In this paper, we propose a deep Regularization Self-Attention Regression (namely RSAR) model to predict daily precious metal price. Fig. 1 shows an overview of RSAR model, which consists of the following components:

1) **Data Preprocessing.** In this phase, we conduct a preliminary analysis and perform data preprocessing for massive daily precious metal price datasets.
2) **LSTM layer.** Long short-term memory (LSTM) is essentially a modified version of recurrent neural network (RNN) proposed by [25] and improved by [26]. As shown in Fig. 1, the LSTM layer consists of three LSTM units with end-to-end training. We adopt LSTM layer to extract the time-series features from daily precious metal price datasets.
3) **Regularization Self-Attention Mechanism.** In order to better capture the effective information from the encoding data after LSTM layer, we propose a Regularization Self-Attention mechanism. In particular, we adopt three types of regularization functions and evaluate the performance. Details can be referred to Section III-C.
4) **Convolution Layer.** The convolution layer can extract the global spatial features from the daily precious

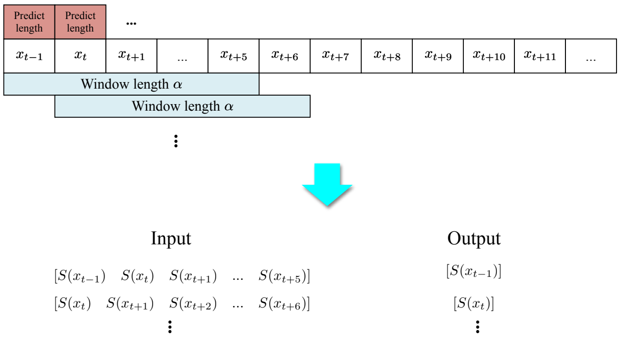

FIGURE 2. Data Preprocessing

metal price data. We exploit convolution neural network (CNN) structure here mainly motivated by previous studies [17], [27]–[29] because CNN component can extract the global spatial features as there exists the spatial dependency of daily precious metal price.

5) **Fully-Connected layer.** Finally, we adopt a fully-connected layer which consists a number of neurons to extract the key features.

Our goal is to design a robust model for precious metal price prediction. To achieve this goal, we first normalize the raw price data during data preprocessing. Then, we employ an LSTM layer to capture the temporal features from daily precious metal price data due to the temporal sequential dependency of the data. In particular, the incarnation of RSAR model in this paper is based on regularization self-attention mechanism, which can reduce the computational cost in the model. Moreover, in order to further improve the prediction precision, we exploit a CNN layer to extract the spatial features for the precious metal price data. Finally, we utilize a fully-connected layer to reduce the dimension of spatial representations for prediction.

B. DATA PREPROCESSING

We obtain precious metal price datasets formally released by Macrotrends[1], which is a premier research platform for both investors and researchers. We select Gold prices and Palladium prices from a number of precious metals. Specifically, the datasets contain 10,471 records of daily gold prices (from Dec. 29, 1978 to Feb. 15, 2019) and 10,645 records of daily palladium prices (from Jan. 5, 1977 to May. 10, 2019), where the records only exist on trading days. Since neural networks are sensitive to the diversity of input datasets, we need to normalize data via the standard scaling method. Motivated by the strength of normalizing the mean and standard deviation of the features, we adopt scikit-learn [30] tool to normalize the row datasets. In particular, calculations of standard scalar of precious metal price values can be expressed as follows,

$$S(x_i) = \frac{x_i - \mu}{\sigma}, \tag{1}$$

where x_i denotes the value of precious metal price of a day, $S(x_i)$ is the scalar of x_i, mean denoted by μ can be calculated by $\mu = \frac{1}{N}\sum_{i=1}^{N}(x_i)$, standard deviation denoted by σ can be calculated by $\sigma = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(x_i - \mu)^2}$ and N denotes the number of input data for trading days.

In order to process the input data, it is necessary to choose an appropriate sliding window length, which is denoted by α. Specifically, the dimension of the input data is $\alpha + 1$. Take Fig. 2 as an example, where we set $\alpha = 7$ because each week is composed of 7 days. According to window length α, we transform $x_{t-1}, x_t, \cdots, x_{t+5}$ into $S(x_{t-1}), S(x_t), \cdots, S(x_{t+5})$. The corresponding values of output are $S(x_{t-1})$. Fig. 2 shows the detailed process of the transformation.

C. REGULARIZATION SELF-ATTENTION MECHANISM

In this section, we introduce Regularization Self-Attention (RSA) mechanism which is a core building block in our proposed Model. It is challenging to extract both the temporal and spatial features from the precious metal price data. Motivated by recent work [31], we adopt self-attention mechanism to enhance the learning procedure of precious metal price data. Moreover, we also further improve the effectiveness of self-attention mechanism via different regularization methods. We describe details as follows.

1) Self-Attention Mechanism

Self-attention mechanism exhibits a better balance between the ability of modelling long-range dependencies and computational efficiency. In particular, the self-attention module calculates the response at one position in the feature map as a weighted sum of the features from all positions. As a result, the weights are calculated with only small computational cost.

Fig. 3 shows the working flow of RSA mechanism. In order to calculate the weighted attention value, the input features denoted by $x \in \mathbb{R}^{C\times N}$ are transformed into three feature spaces f, g and h via passing through 1×1 convolution. Therefore, the first step is to calculate matrices f, g and h. The embedding features were packed into a matrix, and then multiplied by the trained weight matrices, which can be calculated by the following equations,

$$f(x_i) = Wx_i + b, \tag{2}$$

$$g(x_j) = Wx_j + b, \tag{3}$$

$$h(x_i) = Wx_i + b, \tag{4}$$

where W denotes the weights and b denotes the bias parameters. Meanwhile, we adopt regularizers to normalize the weights and bias. The details of regularizers will be introduced in section III-C2. The next step is to multiply matrices $f(x_i)^\intercal$ and matrices $g(x_j)$ by $softmax$ function and to sum up the weighted value vectors, where $f(x_i)^\intercal$ denotes the transposed matrices of $f(x_i)$. After obtaining *the attention map* through above steps, we can calculate *regularization self-attention feature maps* by multiplying *attention*

[1] https://www.macrotrends.net/

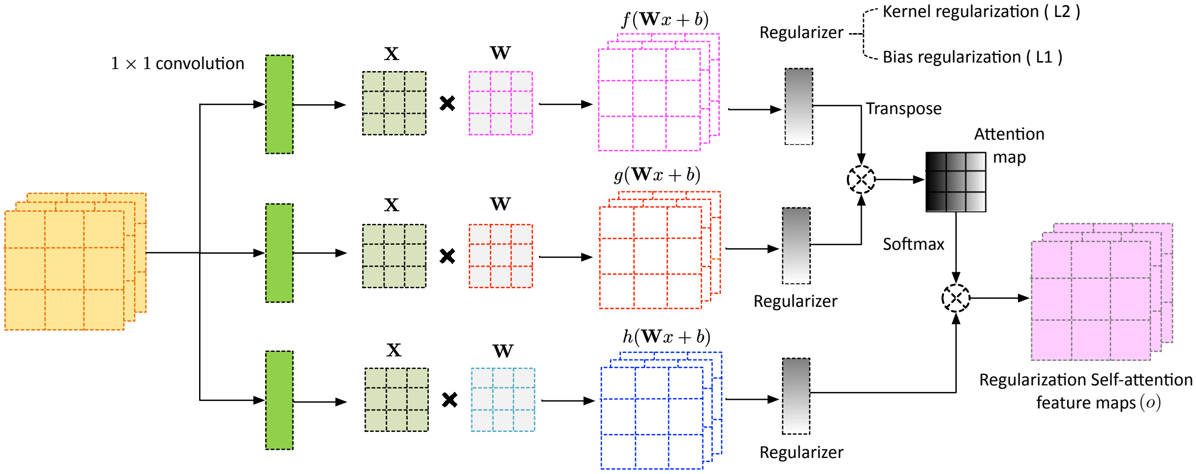

FIGURE 3. Regularization Self-Attention Mechanism

map and matrix $h(x_j)$. It is worth mentioning that Eq.(5) to Eq.(7) describe the operations of RSA Mechanism. The self-attention mechanism calculations are represented as follows.

$$e_{ij} = \frac{f(x_i)^T g(x_j)}{\sqrt{d_k}}, \tag{5}$$

$$a_{ij} = \mathsf{softmax}(e_{ij}) = \frac{\exp(e_{ij})}{\sum_j \exp(e_{ij})}, \tag{6}$$

$$o = a_{ij} h(x_i) = \mathsf{softmax}(e_{ij}) h(x_i), \tag{7}$$

where e_{ij} represents the relation between the i^{th} value and j^{th} value. In particular, in Eq. (5), *attention map* is divided by $\sqrt{d_k}$ which is the square root of the dimension of the matrix $f(x_i)$ vectors. Consequently, it leads to a faster convergence than previous methods. We denote the attention weight of the i^{th} value versus j^{th} value by a_{ij} via the softmax function. Meanwhile, o represents *regularization self-attention feature maps*.

2) Regularization Mechanism for Self-attention

In general, the embedding matrices f, g and h suffer from the redundancy if the attention mechanism always provides approximated summation weights every time. Therefore, we need regularization processing to improve the diversity of summation weight vectors across different attention-hops. We adopt two different regularizers [32] to improve the diversity of attention mechanisms.

- L_2 Regularization for Kernel Regularization. We use L_2 Regularization to regularize the weight matrices of a number of kernels [33]. The main idea of L_2 Regularization is to minimize the sum of the square of the differences D between the target for weight value W and the estimated weight value W_i:

$$D = \sum_{i=1}^{N} (W - W_i)^2. \tag{8}$$

- L_1 Regularization for Bias Regularization. In addition, we employ L_1 Regularization to regularize the bias values. The L_1 Regularization is basically minimizing the sum of the absolute differences D between the target for bias value b and the estimated bias value b_i:

$$D = \sum_{i=1}^{N} |b - b_i|. \tag{9}$$

D. AUXILIARY MODULES

In order to control the sequential order and capture the spatial features of the daily precious metal price. In this paper, we also employ an LSTM layer and a CNN layer as the auxiliary modules in RSAR model. We present more details about them as follows.

1) LSTM layer

We choose an LSTM component to process the time sequence T which consists of the input sequence $S(x_{T+1})$, $S(x_{T+2}), \cdots, S(x_{T+7})$. We select β as an adjustable parameter in the experiment, where β denotes the number of LSTM layers.

2) Convolution layer

We employ a convolution layer to capture the spatial features of daily precious metal price for our proposed RSAR model. We also select γ as an adjustable parameter in the CNN component, where γ denotes the number of filters in one convolutional layer, as shown in Fig. 1. The convolution layer, which essentially consists of a CNN component, can extract the global spatial features. Following the convolution layer, we also leverage a max pooling layer, which can be used to reduce the number of parameters and features and avoid over-fitting.

IV. EXPERIMENTAL RESULTS

In this section, we conduct a number of experiments to evaluate the performance of the proposed RSAR approach.

TABLE 1. Performance comparison with baseline approaches

Models	Palladium-price dataset			Gold-price dataset		
	RMSE	MAE	MAPE	RMSE	MAE	MAPE
SVR	1.28E+03	1.069E+03	1.26E+02	1.04E+03	1.011E+03	7.77E+01
ARIMA	1.66E+03	1.018E+03	1.74E+02	8.84E+02	7.896E+02	5.81E+01
Deep Regression	1.52E+02	1.061E+02	1.17E+01	1.63E+02	1.346E+02	9.37E+00
CNN	1.86E+02	1.742E+02	2.16E+01	6.67E+01	6.037E+01	4.33E+00
LSTM	5.60E+01	3.158E+01	3.65E+00	6.04E+01	5.402E+01	3.23E+00
LSTM-CNN	4.95E+01	4.02E+01	5.03E+00	4.25E+01	3.047E+01	2.12E+00
RSAR (w/o regularization)	**4.84E+01**	3.292E+01	3.99E+00	3.13E+01	2.209E+01	1.56E+00
RSAR Model (R1)	4.87E+01	3.283E+01	3.96E+00	3.05E+01	2.228E+01	1.61E+00
RSAR Model (R2)	4.88E+01	3.181E+01	3.81E+00	3.01E+01	2.162E+01	1.55E+00
RSAR Model (R3)	4.86E+01	**3.136E+01**	**3.75E+00**	**2.85E+01**	**2.034E+01**	**1.46E+00**

In particular, we give the experiment settings as well as performance metrics in Section IV-A. We then present the comparison results of the proposed RSAR approach with other baseline models in Section IV-B. We next further investigate the impacts of parameters on the performance of the proposed approach in Section IV-C.

A. EXPERIMENT SETTINGS

1) Dataset description: We obtain the precious metal prices from Macrotrends. In particular, we select Gold prices and Palladium prices for the analysis. Therefore, the experimental datasets contain (i) daily gold-price dataset, (ii) daily palladium-price dataset. The daily price is essentially represented in different currencies (e.g. US dollar, EUR, RMB, HK dollar) per ounce (i.e., oz). To unify the analysis, we choose the price in US dollar per ounce on trading days to perform prediction analysis.

2) Model Setting: In our experiment, we fix the window length to be $\alpha = 7$ in the data preprocessing. Therefore, there are 7 price values in each input matrix. In addition, the network weights are shuffled for initialization via a truncated normal distribution ($\mu = 0$ and standard deviation $\sigma = 1.0$). Moreover, in order to improve the training efficiency, we fix the batch size in training set at 80 and number of training epochs at 100.

3) Performance metrics: In order to compare the proposed approach with other baseline models, we adopt four performance metrics: root mean square error (RMSE), mean absolute error (MAE) and mean absolute percentage error (MAPE) and loss. RMSE of a model is the standard deviation of the residuals between predicted values and observed values [34]. MAE represents a measure of difference between two continuous variables and calculate the average of all absolute errors. MAPE represents a measure of prediction accuracy of a predictive method in statistics and usually expresses the accuracy as a percentage especially when MAE is too small. These three metrics have been widely used in predication tasks. These three performance metrics can be computed as follows,

$$\text{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(x_{1,i} - x_{2,i})^2}, \tag{10}$$

$$\text{MAE} = \frac{1}{N}\sum_{i=1}^{N}\left|x_{1,i} - x_{2,i}\right|, \tag{11}$$

$$\text{MAPE} = \frac{100}{N}\sum_{i=1}^{N}\left|\frac{x_{2,i} - x_{1,i}}{x_{1,i}}\right|. \tag{12}$$

Furthermore, we also evaluate the impacts of different parameters of our RSAR model by comparing the loss of training set of each epoch. The loss is evaluated by mean square error (MSE), which is essentially the square of RMSE (i.e., Eq. (10)). The loss is presented as follows,

$$\text{Loss} = \frac{1}{N}\sum_{i=1}^{N}(x_{1,i} - x_{2,i})^2, \tag{13}$$

where $x_{1,i}$ and $x_{2,i}$ represent the forecast value and the actual value, respectively. In particular, the lower values of these metrics imply higher performance of models.

B. PERFORMANCE COMPARISON

Baseline models: We compare the proposed method with six representative baseline models as follows:

- **Deep Regression:** It is a basic Machine Learning model with various activation functions applied in computer vision [35]. This method consists of two layers of neural

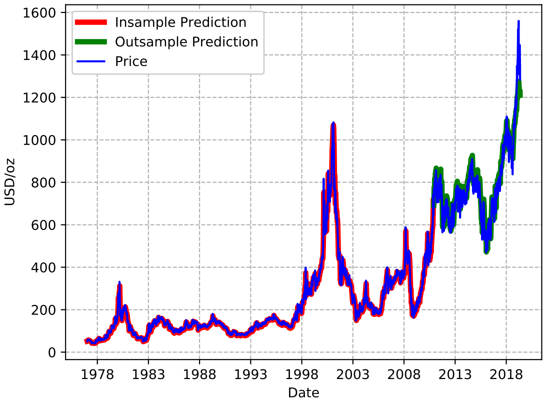

FIGURE 4. Prediction of RSAR Model in Palladium-price dataset

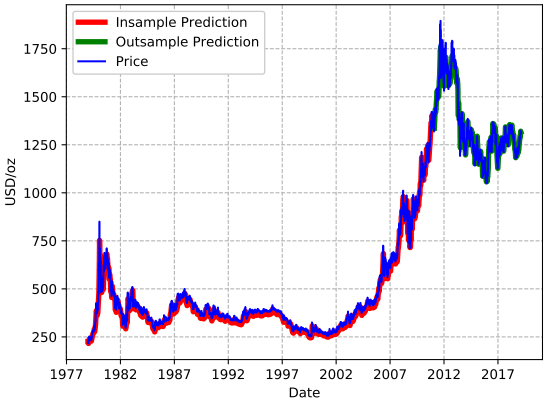

FIGURE 5. Prediction of RSAR Model in Gold-price dataset

networks with $\tanh$ activation function in our experiment.

- **Support Vector Regression (SVR):** SVR is an alternative to Support Vector Machine (SVM) which is a basic support vector classifier (SVC) with radial basis function (RBF) kernel. It is also a typical Machine Learning model to support stock price prediction [13].
- **Autoregressive Integrated Moving Average model (ARIMA):** This model can capture a variety of standard temporal structures from time series data. ARIMA as a financial prediction and analysis tool has been widely applied in finance analysis [7], [8].
- **Convolutional Neural Network (CNN):** CNN consists of several convolutional layers, pooling layers and a fully-connected layer. This model has the advantages in learning complicated data. In this paper, we implement CNN model with 2 convolutional layers to conduct the precious metal price prediction experiment.
- **Long Short-Term Memory Neural Networks (LSTM):** LSTM is well-suited to predictions based on time series data. We implement LSTM model with 2 layers to conduct the prediction experiment.
- **LSTM-CNN:** This scheme consists of a LSTM layer alternating with a CNN layer. We implement LSTM-CNN model with 1 LSTM layer and 1 convolutional layer to conduct the prediction experiment.

We conducted experiments with the training ratio equal to 80% in both palladium-price dataset and gold-price datasets. Meanwhile, we let the number of LSTM layers be $\beta = 1$ and the number of CNN filters be $\gamma = 64$ in our RSAR model. In each dataset, we evaluate the proposed RSAR model and baseline models in terms of RMSE, MAE and MAPE.

Table 1 presents the performance comparison of our RSAR model with other baseline methods. First, we compare conventional ML models including SVR, ARIMA and Deep Regression model for the precious metal price prediction. Compared with other DL methods, SVR, ARIMA and deep regression have much higher values of RMSE, MAE and MAPE, implying the poorer performance. For example, as shown in Table 1, SVR achieved $1.28E+03$, $1.069E+03$ and $1.26E+02$ in RMSE, MAE and MAPE, respectively, in Palladium-price prediction; these values are the largest among all the methods.

Second, we analyze the conventional DL models including CNN, LSTM and LSTM-CNN. Compared with ML models, DL models (such as CNN, LSTM and LSTM-CNN) achieved the better performance (in terms of lower values of RMSE, MAE and MAPE). The reason may lie in the strength of DL models in generalization especially after learning massive palladium-price and gold-price data.

Third, we propose four RSAR models to evaluate the performance of these models with different regularization: a) RSAR Model without (w/o) regularization is a simplified version of our proposed RSAR models with the removal of regularization module, b) RSAR Model (R1) is the proposed RSAR model with L_2-*Regularization* only for kernel regularization, c) RSAR Model (R2) is the RSAR model with both L_1 and L_2 *regularizations*, d) RSAR Model (R3) is an improved version of RSAR Model (R2) with the weight optimization via the attention regularizer. It is shown Table 1 that all of our proposed RSAR models, such as RSAR (w/o regularization), RSAR (R1), RSAR (R2) and RSAR (R3) outperform conventional ML and DL models in both Palladium-price and Gold-price datasets. Moreover, compared with RSAR (w/o regularization), RSAR (R1), RSAR (R2) and RSAR (R3) can achieve even better performance due to the regularization in kernel and bias parameters via L_2-*regularization* and L_1-*regularization* except for RMSE in Palladium-price dataset where RSAR (w/o regularization) performs slightly better than other three models. Furthermore, RSAR Model (R3) even outperforms RSAR (R1) and RSAR (R2) in terms of RMSE, MAE and MAPE due to the optimized weights in L_1-*regularization* and L_2-

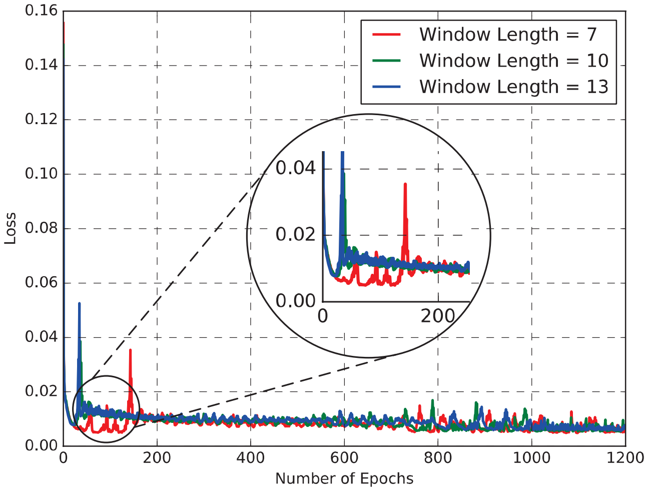

FIGURE 6. Loss of RSAR Model with different values of window length

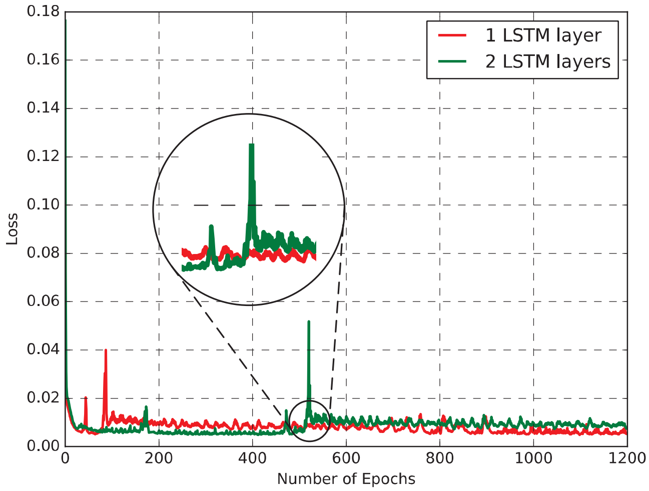

FIGURE 7. Loss of RSAR Model with different numbers of LSTM layers

regularization.

C. PARAMETER STUDY

We investigate the impacts of various parameters on the performance of the proposed RSAR model. In particular, we first analyze the forecasting tendency of both Palladium and Gold price datasets. The results are shown in Fig. 4 and Fig. 5, in which *Insample Prediction* (a.k.a. in-sample forecast) implies the training dataset fitting the model and *Outsample Prediction* (a.k.a. out-of-sample forecast) means the test dataset fitting the model. It is worth mentioning that *Insample Prediction* and *Outsample Prediction* are represented in orange curves and green curves, respectively. Moreover, the blue curves represents the real prices of precious metals in USD/oz. It is shown in both Fig. 4 and Fig. 5 that the prediction (forecast) results fit well with the real prices in both Palladium-price and Gold-price datasets.

We then investigate the impacts of parameters on our proposed RSAR model. As indicated in Table 1, the different datasets have the little effect on performance. Thus, we conduct the following experiments mainly based on Gold-price dataset. In particular, we take the following parameters into account: 1) the window length denoted by α; 2) the number of LSTM layers denoted by β; 3) the number of CNN filters denoted by γ.

1) Impact of window length (α): We first investigate the impact of the window length for data preprocessing. In particular, we fix the number of LSTM layers (β) to 1 and the number of CNN filters (γ) to 64, then we vary the size of window length α from 7, 10 and 13, representing a week and nearly one third a month and half a month, respectively. Table 2 shows the loss results after training 1200 iterations. We observe from Table 2 that the prediction loss increases with the increased size of window length. Therefore, we can learn that window length being equal to 7 can result in a slightly better performance than other two window lengths.

Moreover, Fig. 6 shows the performance comparison for different values of window length. We can observe that loss values decline dramatically after 50 iterations. In particular, Fig. 6 also shows that the model with window length $\alpha = 7$ achieves faster convergence than others in 200 training iterations. Moreover, Fig. 6 indicates that the loss keep relatively stable after training converges.

2) Impact of number of LSTM layers (β): We next investigate the impact of number of LSTM layers in LSTM component. We vary the number of LSTM layers from 1 to 2 in LSTM component (the number of LSTM layers denoted by β). Meanwhile, we fix α to be 7 and the number of CNN filters (γ) to be 64. Similarly, we conduct the experiments on Gold-price dataset after 1200 iterations. Table 2 shows the results.

It is shown in Table 2 that the loss increases with the increased number of LSTM layers, implying that the larger number of LSTM layers may not contribute to the performance improvement. Fig. 7 further investigates the impacts of number of LSTM layers. In particular, as shown in Fig. 7, the loss of the proposed model with 2 LSTM layers is very close to that with 1 LSTM layer. The reason behind the results may lie in the fact that the increment of number of LSTM layers may not be helpful in reducing loss especially in sparse dataset.

3) Impact of number of CNN filters (γ): We also investigate the impact of number of CNN filters in CNN component. To investigate the impact of number of CNN filters (the number of CNN filters denoted by γ), we set the number of filters to 32, 64 and 128. At the same time, we fix α at 7 and β at 1. Table 2 shows the loss values in different numbers of CNN filters. We observe that the loss value with 64 CNN filters are lower than others. Fig. 8 shows that the loss value with 32 CNN filters still fluctuates after training 400 iterations, implying no convergence. In contrast, the models with 64 CNN filters and 128 CNN filters converge much faster than that with 32 CNN filters. Therefore, the increased number of CNN filters can lead to the fast convergence of the model

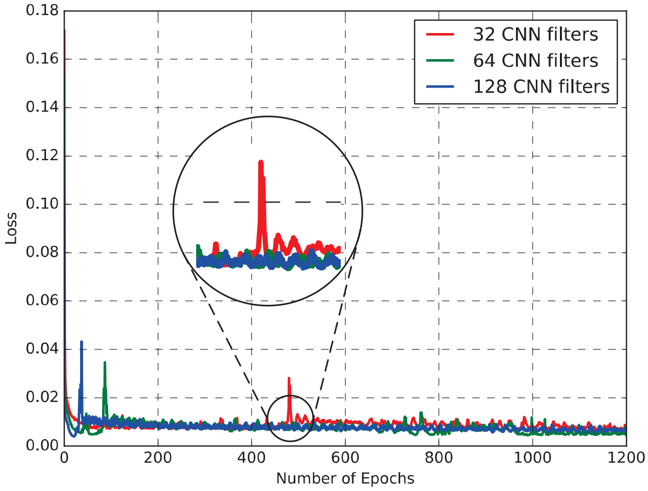

FIGURE 8. Loss of RSAR Model with different number of CNN filters

TABLE 2. Impact of Various Parameters

Investigation	Parameters	**Loss**
Window Length (α)	7	0.0056
	10	0.0070
	13	0.0076
Number of LSTM Layers (β)	1	0.0054
	2	0.0075
Number of CNN Filters (γ)	32	0.0077
	64	0.0047
	128	0.0068

while the larger number of CNN filters may take a longer training time, thereby there existing a trade-off between the number of CNN filters and performance.

V. CONCLUSION

In this paper, we propose Regularization Self-Attention Regression Model (RSAR model) for daily precious metal price forecasting. Our proposed model mainly consists of LSTM component, CNN component and Regularization Self-Attention. In particular, RSA mechanism can improve the performance via employing regularization functions (i.e., L_1 and L_2). Meanwhile, both LSTM and CNN modules can help to extract both spatial and time series features from precious metal price dataset. We also conduct extensive experiments to evaluate the performance of the proposed model with comparison with other existing ML and DL methods. The results show that our proposed model outperforms conventional ML and DL methods such as ARIMA, SVR, CNN and LSTM. Regarding future directions, we will investigate the performance improvement of the proposed model via adjusting different numbers of LSTM and CNN layers when considering different types of financial data.

REFERENCES

[1] A. McCallum, K. Nigam et al., “A comparison of event models for naive bayes text classification,” in AAAI-98 workshop on learning for text categorization, vol. 752, no. 1. Citeseer, 1998, pp. 41–48.

[2] F. Black, M. C. Jensen, M. Scholes et al., “The capital asset pricing model: Some empirical tests,” Studies in the theory of capital markets, vol. 81, no. 3, pp. 79–121, 1972.

[3] E. F. Fama and K. R. French, “Common risk factors in the returns on stocks and bonds,” Journal of Financial Economics, vol. 33, no. 1, pp. 3 – 56, 1993. [Online]. Available: http://www.sciencedirect.com/science/article/pii/0304405X93900235

[4] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “Imagenet classification with deep convolutional neural networks,” Communications of the Acm, vol. 60, no. 2, p. 2012, 2012.

[5] M. Almasarweh and S. Alwadi, “Arima model in predicting banking stock market data,” Modern Applied Science, vol. 12, p. 309, 10 2018.

[6] R. C. Chung, W. Ip, and S. Chan, “An arima-intervention analysis model for the financial crisis in china’s manufacturing industry,” International Journal of Engineering Business Management, vol. 1, p. 5, 2009. [Online]. Available: https://doi.org/10.5772/6785

[7] K. Yunus, T. Thiringer, and P. Chen, “Arima-based frequency-decomposed modeling of wind speed time series,” IEEE Transactions on Power Systems, vol. 31, no. 4, pp. 2546–2556, July 2016.

[8] A. Vaccaro, T. H. M. EL-Fouly, C. A. Cañizares, and K. Bhattacharya, “Local learning-arima adaptive hybrid architecture for hourly electricity price forecasting,” in 2015 IEEE Eindhoven PowerTech, June 2015, pp. 1–6.

[9] R. Alwadi, “Forecasting short term financial data,” European Scientific Journal, pp. 251–255, 2015.

[10] B. Guha and G. Bandyopadhyay, “Gold price forecasting using arima model,” Journal of Advanced Management Science, vol. 4, no. 2, pp. 117–121, 03 2016.

[11] F. Li and C. Liu, “Application study of bp neural network on stock market prediction,” in 2009 Ninth International Conference on Hybrid Intelligent Systems, vol. 3, Aug 2009, pp. 174–178.

[12] M. Zaidi and A. Amirat, “Forecasting stock market trends by logistic regression and neural networks evidence from ksa stock market,” International Journal of Economics, Commerce and Management, vol. IV, no. 6, pp. 220–234, 2016.

[13] K.-j. Kim, “Financial time series forecasting using support vector machines,” Neurocomputing, vol. 55, pp. 307–319, 09 2003.

[14] C. Cortes and V. Vapnik, “Support-vector networks,” Mach. Learn., vol. 20, no. 3, pp. 273–297, Sep. 1995. [Online]. Available: https://doi.org/10.1023/A:1022627411411

[15] A. Mohamed, T. N. Sainath, G. Dahl, B. Ramabhadran, G. E. Hinton, and M. A. Picheny, “Deep belief networks using discriminative features for phone recognition,” in 2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), May 2011, pp. 5060–5063.

[16] J. Zhou, Y. Lu, H.-N. Dai, H. Wang, and H. Xiao, “Sentiment analysis of chinese microblog based on stacked bidirectional lstm,” IEEE Access, vol. 7, pp. 38 856–38 866, 2019.

[17] Z. Zheng, Y. Yang, X. Niu, H.-N. Dai, and Y. Zhou, “Wide and deep convolutional neural networks for electricity-theft detection to secure smart grids,” IEEE Transactions on Industrial Informatics, vol. 14, no. 4, pp. 1606–1615, April 2018.

[18] Z. Zheng, Y. Yang, J. Liu, H.-N. Dai, and Y. Zhang, “Deep and embedded learning approach for traffic flow prediction in urban informatics,” IEEE Transactions on Intelligent Transportation Systems, pp. 1–13, 2019.

[19] T. Kim and H. Y. Kim, “Forecasting stock prices with a feature fusion lstm-cnn model using different representations of the same data,” PLOS ONE, vol. 14, no. 2, pp. 1–23, 02 2019. [Online]. Available: https://doi.org/10.1371/journal.pone.0212320

[20] A. M. Alayba, V. Palade, M. England, and R. Iqbal, “A combined cnn and lstm model for arabic sentiment analysis,” in Machine Learning and Knowledge Extraction, A. Holzinger, P. Kieseberg, A. M. Tjoa, and E. Weippl, Eds. Cham: Springer International Publishing, 2018, pp. 179–191.

[21] Y. Wen and B. Yuan, “Use cnn-lstm network to analyze secondary market data,” in Proceedings of the 2Nd International Conference on Innovation in Artificial Intelligence, ser. ICIAI ’18. New York, NY, USA: ACM, 2018, pp. 54–58. [Online]. Available: http://doi.acm.org/10.1145/3194206.3194226

[22] V. Mnih, N. Heess, A. Graves, and K. Kavukcuoglu, “Recurrent models of visual attention,” in Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2, ser. NIPS’14. Cambridge, MA, USA: MIT Press, 2014, pp. 2204–2212. [Online]. Available: http://dl.acm.org/citation.cfm?id=2969033.2969073

[23] R. He, W. S. Lee, H. T. Ng, and D. Dahlmeier, “An unsupervised neural attention model for aspect extraction,” in Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). Vancouver, Canada: Association for Computational Linguistics, Jul. 2017, pp. 388–397. [Online]. Available: https://www.aclweb.org/anthology/P17-1036

[24] N. Si, H. Wang, W. Li, and Y. Shan, “Chinese pos tagging with attention-based long short-term memory network,” in Advances in Intelligent Systems and Interactive Applications, 06 2018, pp. 586–592.

[25] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” Neural Computation, vol. 9, no. 8, pp. 1735–1780, 1997. [Online]. Available: https://doi.org/10.1162/neco.1997.9.8.1735

[26] A. Graves, A. Mohamed, and G. Hinton, “Speech recognition with deep recurrent neural networks,” in 2013 IEEE International Conference on Acoustics, Speech and Signal Processing, May 2013, pp. 6645–6649.

[27] Y. Lecun and Y. Bengio, Convolutional networks for images, speech, and time-series. MIT Press, 1995.

[28] R. Yamashita, M. Nishio, R. K. G. Do, and K. Togashi, “Convolutional neural networks: an overview and application in radiology,” Insights into Imaging, vol. 9, no. 4, pp. 611–629, Aug 2018. [Online]. Available: https://doi.org/10.1007/s13244-018-0639-9

[29] N. Tajbakhsh, J. Y. Shin, S. R. Gurudu, R. T. Hurst, C. B. Kendall, M. B. Gotway, and J. Liang, “Convolutional neural networks for medical image analysis: Full training or fine tuning?” IEEE Transactions on Medical Imaging, vol. 35, no. 5, pp. 1299–1312, May 2016.

[30] https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.StandardScaler.html.

[31] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, “Attention is all you need,” in Advances in neural information processing systems, 2017, pp. 5998–6008.

[32] G. Zhou, G. Xu, J. Hao, S. Chen, J. Xu, and X. Zheng, “Generalized centered 2-d principal component analysis,” IEEE Transactions on Cybernetics, pp. 1–12, 2019.

[33] C. Cortes, M. Mohri, and A. Rostamizadeh, “L2 regularization for learning kernels,” in Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence, ser. UAI ’09. Arlington, Virginia, United States: AUAI Press, 2009, pp. 109–116. [Online]. Available: http://dl.acm.org/citation.cfm?id=1795114.1795128

[34] P. Tsang, P. Kwok, S.-O. Choy, R. Kwan, S. Ng, J. Mak, J. Tsang, K. Koong, and T.-L. Wong, “Design and implementation of nn5 for hong kong stock price forecasting,” Engineering Applications of Artificial Intelligence, vol. 20, pp. 453–461, 06 2007.

[35] S. Lathuilière, P. Mesejo, X. Alameda-Pineda, and R. Horaud, “A comprehensive analysis of deep regression,” IEEE transactions on pattern analysis and machine intelligence, 2019.