PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ

Escuela de Posgrado

Pronóstico del Precio de Cobre utilizando técnicas de Aprendizaje Profundo

Trabajo de Investigación para optar el grado académico de Magíster en Informática con mención en Ciencias de la Computación que presenta:

Milton César Carhuas Ñañez

Asesor:

Edwin Rafael Villanueva Talavera

Lima, 2022

Informe de Similitud

Yo, Edwin Rafael Villanueva Talavera, docente de la Escuela de Posgrado de la Pontificia Universidad Católica del Perú, asesor(a) de la tesis/el trabajo de investigación titulado Pronóstico del precio del cobre utilizando técnicas de aprendizaje profundo, del autor Milton César Carhuas Ñañez, dejo constancia de lo siguiente:

- El mencionado documento tiene un índice de puntuación de similitud de 14%. Así lo consigna el reporte de similitud emitido por el software *Turnitin* el 18/10/2023.
- He revisado con detalle dicho reporte y la Tesis o Trabajo de Suficiencia Profesional, y no se advierte indicios de plagio.
- Las citas a otros autores y sus respectivas referencias cumplen con las pautas académicas.

Lugar y fecha:
Lima, 18 de enero del 2023.

Apellidos y nombres del asesor / de la asesora: Villanueva Talavera, Edwin Rafael	
DNI:29714308	Firma:
ORCID: 0000-0002-6540-1230	

RESUMEN

Pronosticar los precios futuros de cobre es una tarea desafiante dadas las características dinámicas y no lineales de varios factores que afectan el precio del cobre. Este artículo describe modelos de pronóstico, basados en arquitecturas de redes neuronales, para predecir los retornos del precio de cobre en tres horizontes de tiempo: un día, una semana y un mes adelante. Diversas variables se consideran como variables de entrada, como los precios históricos de diferentes materias primas metálicas y variables macroeconómicas globales. Evaluamos los modelos con datos diarios de 2007 a 2020. Los resultados experimentales mostraron que los modelos de salida única presentan un mejor rendimiento predictivo que los modelos de salida múltiple. Las arquitecturas de mejor rendimiento fueron los modelos de memorias largas a corto plazo (LSTM) en datos de prueba.

On multi-step forecasting of copper price returns using deep learning techniques

Milton Carhuas[a] and Edwin Villanueva[a]

[a]Pontifical Catholic University of Peru, Av. Universitaria 1801, 15088, Lima, Peru;

ARTICLE HISTORY
Compiled January 27, 2023

ABSTRACT
Forecasting the future prices of copper commodity is a challenging task given the dynamic and non-linear characteristics of various factors that affect the copper price. This article describes forecasting models, based on neural network architectures, to predict copper price returns at three time horizons: one-day, one-week, and one-month ahead. Several variables are considered as input variables, like historical prices of different metallic commodities and global macroeconomic variables. We evaluated the models with daily data from 2007 to 2020. The experimental results showed that mono-output models present better predictive performance than multi-output models. The best-performing architectures were the Long Short-Term Memories (LSTM) models on test data.

KEYWORDS
Deep Learning, and Recurrent Neural Networks, and Time Series, and Copper Price Forecasting and Multi-step Forecasting.

1. Introduction

The volatility of copper prices generates a high impact on many economic activities because it is a metal commodity widely used in various industries due to its remarkable properties of being a high electrical and thermal conductor, great chemical stability and resistent to corrosion. For these reasons, it is currently one of the most traded commodities in the world. Peru is currently one of the largest global producers of this commodity and, consequently, its contribution to the peruvian economy is significant since it is the sector that generates the largest earnings for the country, so it is desirable to be able to adequately estimate the future international price of this material. However, according to copper industry analysis, copper prices are highly volatile and dependent on many external factors (Alameer et al., 2019b).

Several methods have been proposed to predict copper prices, including recent Deep Learning (DL) approaches. Most of these methods are designed to forecast at a prefixed time horizon (Alameer et al. (2019b,a); Jianwei et al. (2019)). However, depending on the user and the level of decision where the forecast will be used, it is desirable to have forecasted prices at different time horizons. Most of the works in the literature on commodity price forecasting propose models to predict a specific time horizon (mono-output models) (Díaz et al., 2020; Alameer et al., 2019b; Livieris et al., 2020). However, it is also possible to build a prediction model to forecast simultaneously

Email Adresses: mcarhuasn@pucp.edu.pe (Milton Carhuas), ervillanueva@pucp.edu.pe (Edwin Villanueva)

several time horizons ahead (multi-output model) (Xiong et al., 2019), which could be advantageous as it is easier to deploy and maintain. However, it is not clear which of these approaches may be more suitable for multi-step copper price prediction.

This paper investigates neural network models to predict copper price returns at 3 time horizons: one-day ahead, one-week ahead and one-month ahead. We compare mono-output models against multi-output models. As input variables, we use different variables related to cooper price returns. The importance of this work lies in the possibility of its use for government agents in countries where the economy is strongly linked to copper, as is the case of Peru and Chile, and also for private agents who wish to make investment decisions in the commodity market (current and future) so that they can better allocate their assets and diversify their investment portfolio. It is worth mentioning that currently more than 90% of industrial projects require copper, which confirms its importance in economic terms and its impact on society WANG et al. (2013). In the new global economy, forecasting future mineral prices has become a central issue for mining projects and related businesses. Therefore, an accurate multi-horizon forecasting model may play an important role in helping to make correct decisions Sabour and Poulin (2006).

This paper is structured as follows: Section 1 describes an introduction and the context in which the research paper is proposed. Section 2 presents a review of related works. In section 3 we describe the dataset, data preprocessing and forecasting models. Section 4 describes the experimentation carried out and the comparison of forecasting models and results. Finally, Section 5 presents the conclusions.

2. Related Works

During the last years, the use of Machine Learning techniques and especially Deep Learning for the forecasting of commodities prices has gained in popularity in the scientific and industrial fields. These techniques have shown interesting results about the behavior of the price of commodities along with details of models, variables, time horizons, metrics and results achieved.

In a recent article, Díaz et al. (2020) propose to forecast the copper price using ensemble algorithms based on decision trees (RF: Random Forest, GB: Gradient Boosting) and the Random-Walk (RW) model. Diaz concludes that the ensemble algorithms (RF, GB) and the RW model have better performance than the decision tree algorithm (DT) for different configurations of predictors and time horizons of the target variable using data between January 2008 and May 2020.

Alameer et al. (2019b) propose the GA-ANFIS hybrid model to forecast copper price volatility. His model combines an Adaptive Neuro-Fuzzy Inference System (ANFIS, which is a type of artificial neural network based on the Takagi-Sugeno fuzzy inference system) and a meta-heuristic method called the genetic algorithm (GA: Genetic Algorithm). The data set used comprises between September 1987 and August 2017 with a monthly granularity. Their results demonstrated the superiority of the GA-ANFIS model over other individual traditional models such as ANFIS, SVM, GARCH, ARIMA.

Alameer et al. (2019a) in a subsequent work proposed the ANN-WOA hybrid model to forecast gold price fluctuations. The proposed model combines a Multi Layer Perceptron (MLP) neural network and a meta-heuristic method called Whale Optimization Algorithm (WOA). The model was developed with a dataset of 360 monthly observations between September 1987 and August 2017. The results showed that the proposed

Table 1.: Works on metal commodity price forecasting

Ref.	Target Var.	Proposed model	Reference model	Predictor Var.	Forecast Time Horizon	Metrics	Result(s)
Díaz et al. (2020)	Daily copper price	Random Forest, Gradient Boosting, Random-Walk	Statistical models	Prices of other commodities (gold, silver, oil, gas, pork, coffee), Dow Jones US Stock Index (Dow Jones Index)	01 day, 01 week, 01 month, 06 months, 01 year, 02 years	RMSE, MAE	Random-Walk model gets the best performance
Alameer et al. (2019b)	Monthly copper price volatility	GA-ANFIS	ANFIS, SVM, GARCH, ARIMA	Prices of other commodities (steel, gold, silver, oil), dollar exchange rate prices (USD/CLP, USD/PEN, USD/CNY), inflation rates (USA, China)	—	MSE, RMSE, MAE	The GA-ANFIS model obtains an improvement of approx. 40%, 21%, 70% and 25% respectively compared to the reference models
Alameer et al. (2019a)	Monthly Gold Price Fluctuations	ANN-WOA	ANN, GA-ANN, PSO-ANN, GWO-ANN y ARIMA	Commodity prices (copper, silver, steel and gold itself, crude oil), currency exchange rate prices (China, India, South Africa), inflation rates (USA and China)	—	R2, MSE, RMSE, STD	The ANN-WOA model obtains an improvement of approx. 24%, 14%, 12%, 8% and 63% compared to the reference models
Kristjanpoller and Minutolo (2015)	Daily gold price volatility (spot and future)	ANN-GARCH	–	Currency exchange rate prices (USD/EUR, USD/JPY), stock market indices (Dow Jones Industrial[DJI], Financial Time Stock Exchange[FTSE]), oil price variation and secondary variables (forecast of the GARCH model and the square of the gold price return)	14, 21 y 28 días	MAPE, MSE, RMSE, MAE	The ANN-GARCH model obtains an improvement of approx. of 25% and 38% for gold spot and future price volatility forecasts respectively
Livieris et al. (2020)	Daily gold price; Daily Gold Price Variation	CNN-LSTM	LSTM, SVM, ANN-MLP	Daily copper prices	3, 6 y 9 días	Regression: RMSE, MAE; Classification: Acc, AUC, Sensibility, Sensitivity	The CNN-LSTM model significantly outperformed all state-of-the-art models in forecasting the price or variation of gold
Jianwei et al. (2019)	Monthly gold price	ICA-GRUNN	ARIMA, RBFNN, LSTM, GRUNN, ICA-LSTM	Daily copper prices	—	MAD, RMSE y MAPE	The hybrid models have a much higher performance than the other compared models of the state of the art, with the ICA-GRUNN model being the one that obtains the best performance for the price forecast.

WOA-ANN model presented better forecast accuracy than other meta-heuristic models such as PSO-ANN (PSO: Particle Swarm Optimization), GA-ANN (GA: Genetic Algorithm), GWO-ANN (GWO: Gray Wolf Optimization) and ARIMA. From this study, it can be also examined the power of the predictor variables used (commodity prices, crude oil prices, exchange rates, inflation rates) to forecast gold price volatility.

Kristjanpoller and Minutolo (2015) proposed the ANN-GARCH hybrid model, which is an artificial neural network (ANN) with the GARCH (Generalized Autoregressive Conditional Heterodasticity) method, which was applied to forecast the volatility of the gold price (spot and future). In turn, it considers a data set from September 1999 to March 2014, for both the spot and future gold price observations. The results showed that the proposed ANN-GARCH model improves the forecast results by 25% for the volatility of the spot gold price and by 38% for the volatility of the future gold price, where the best results were found for the volatility forecast with a horizon of 21 days using input variables USD/EUR, USD/JPY, FTSE, DJI and the return of the oil price.

Livieris et al. (2020) proposed a hybrid model called CNN-LSTM, which is composed of 2 types of neural networks: CNN (Convolutional Neural Networks) and LSTM (Long Short Term Memory) for the prediction of the gold price and its variation (increase or decrease), being a regression and classification problem, respectively. The authors used a set of daily gold price data between January 2014 and April 2018. Additionally, for the classification forecast, the target variable was established as two classes: the gold price increases or decreases the next day. The 2 versions of the proposed model: CNN-LSTM-1 obtained the best performance for predicting the variation in the price of gold (increase or decrease in price), while the CNN-LSTM-2 model significantly outperformed all models in the state of the art in gold price forecasting, getting the best performance of MAE and RMSE. The models reported the best performance for the forecast horizon of 6 days.

Jianwei et al. (2019) proposed a hybrid model, called ICA-GRUNN, to forecast gold monthly closing prices, which applies the technique Independent Component Analysis (ICA) to decompose the time series and retrieve hidden factors from the time series. Then, a recurrent neural network called Gate Recurrent Unit Neural (GRUNN) is trained on the ICA factors. The data set used had a monthly frequency and comprised from January 1979 to December 2017. According to their experiments, the interpretation ratio of the model exceeded 98%, which was notably higher than traditional techniques such as ARIMA and RBFNN, and in turn, these two methods had low values of MAD, RMSE and MAPE. On the other hand, the ICA-LSTM and ICA-GRUNN hybrid models are better than the other state-of-the-art models, with the ICA-GRUNN model being the one with the best performance among these 2 hybrid models.

Table 2 shows a summary of the most relevant works on time series forecasting of metallic commodities prices. It can be observed that there is a lack of works for multi-resolution forecasting of cooper prices. The present work describe a model for this purpose.

3. Methods and Materials

We followed the CRISP-DM (Cross Industry Standard Process for Data Mining) methodology for data analysis and modeling (Schröer et al. (2021)).

3.1. Dataset

The dataset is formed by several financial time series of comodity prices collected from the Bloomberg platform, and additionally some macroeconomic series obtained from public repositories. Table 2 shows relevant information of the collected dataset.

Table 2.: Features of the dataset used

Dataset	**Bloomberg**
Number of Target Series	3
Target Series	Retorno_Diario_Cobre_Log_f3m
	Retorno_Semanal_Cobre_Log
	Retorno_Mensual_Cobre_Log
Number of Related Series	14
Time Rank	2011/Jan/03 - 2021/Oct/01 (aprox. 10 years)
Frequency (Time intervals)	1 day (daily without considering weekend)
Data volume	2,805 observations
Training / Test	2,105 obs. / 700 obs. (75% / 25%)
	2011/Jan/03 - 2018/Dec/28 (Training)
	2018/Dec/31 - 2021/Oct/01 (Test)

(1) **Dataset Overview**: The dataset consists of information on commodity prices, macroeconomic indicators, exchange rates between currencies, and stock market indicators. The original dataset is of daily granularity and covers from 2011 to 2021, with a total of 2,807 observations. This dataset only comprises observations on working days, that is, weekends or holidays are not considered.

(2) **Description of Variables**: Data collection has been carried out for a group of variables that have been considered to have the potential to carry predictive information from copper price returns for the different forecast horizons considered. The detailed description of the collected variables are shown in the table 3.

(3) **Description of Target Variables (Series)**:
The series of logarithmic returns of cooper prices are the following:
- **Logarithmic return of copper price one day ahead**
- **Logarithmic return of copper price one week ahead**
- **Logarithmic return of copper price one month ahead**

These variables are calculated using Equation 2

Table 4 presents summary statistics of the 3 target variables (series) to describe the nature of the distribution of each variable.

Table 4.: Summary statistics of the target variables

Variable	**Avg.**	**Std.Dev.**	**Min.**	**Max.**	**Med.**
Retorno_Diario_Cobre_Log_f3m	-0.000016	0.012790	-0.080740	0.066330	0.000000
Retorno_Semanal_Cobre_Log_f3m	-0.000062	0.030310	-0.179657	0.193763	0.000216
Retorno_Mensual_Cobre_Log_f3m	0.000436	0.065177	-0.280693	0.229158	0.000552

As for the input time window we considered 3, 5, 7 and 9 days in the experimentation. This is called the **"steps in"** parameter.

Figure 1 shows the historical behavior of the daily copper prices between 2011 and 2021. It can be seen that the copper price has had higher values in years 2011

Table 3.: Summary of input variables considered in the modeling

Group	**Variable**	**Description**
Commodities Market	**Retorno_Diario_Cobre_Log_f3m**	Daily Logarithmic Return of Copper based on the 3-months future Copper price
Commodities Market	**Retorno_Semanal_Cobre_Log**	Weekly Logarithmic Return of Copper based on the 3-months future Copper price
Commodities Market	**Retorno_Mensual_Cobre_Log**	Monthly Logarithmic Return of Copper based on the 3-months future Copper price
Stock Market	LME_Cobre_Futures_Volume_Total	Traded volume (total quantity of a product or commodity traded for a specific value during a specific period of time)
Stock Market	Comex_Cobre_Inventory_Data	Copper Inventory
Macroeconomic Indicators	PMI_China	China Purchasing Management Indicator
Macroeconomic Indicators	PMI_USA	United States Purchasing Management Indicator
Macroeconomic Indicators	PBI_China	China's Gross Domestic Product
Exchange Rates	Indice_Dolar_USA	US dollar index (a measure of the value of the dollar in relation to the value of a group of currencies of the countries that are the most important trading partners for the United States (euro, swiss franc, japanese yen, canadian dollar, pound sterling and swedish krona))
Exchange Rates	Rate_Exchange_USD_CNY	Exchange Rate of US Dollar and Chinese Yuan
Macroeconomic Indicators	FED_Rate	US Federal Reserve Interest Rate
Macroeconomic Indicators	CPI_Inflation_China	China Inflation Rate
Commodities Market	Oil_WTI	Crude oil price
Commodities Market	Gold	Gold price
Commodities Market	Silver	Silver price
Commodities Market	Iron_ORE	Steel price
Commodities Market	MSCI_World	MSCI Stock Market Indicator

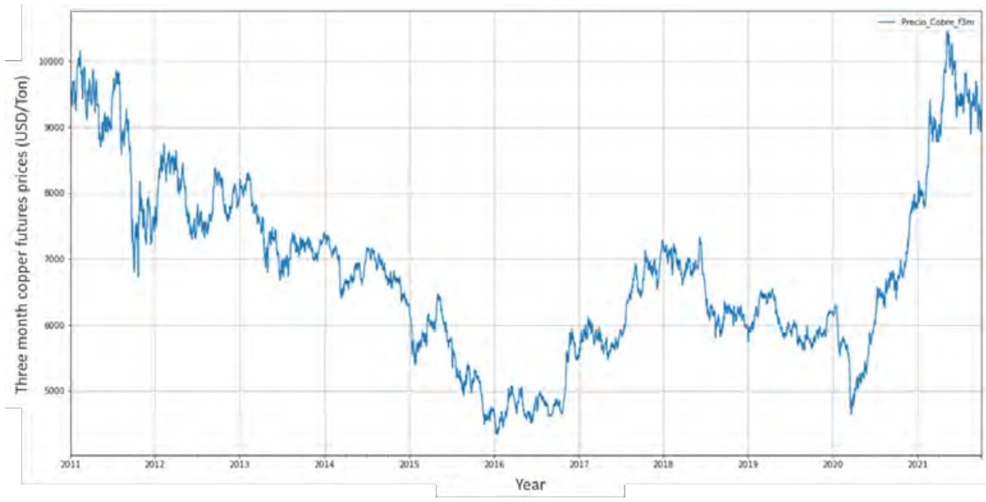


Figure 1.: Historical behavior of the daily copper price between the years 2011 and 2021.

and in 2021 and the lowest values in 2016. The series also exhibits a notorious price volatility over time.

3.2. Correlation Analysis

To understand the correlation between the variables of the dataset we used the Pearson Correlation Coefficient (cp), which measures the degree of linear relationship between 2 variables. The coefficient cp can take values in the range between -1 $\leq$ cp $\leq$ 1, where cp $>$ 0 corresponds to a positive correlation (an increase in the value of one variable corresponds to an increase in the value of the another variable and viceversa), while cp $<$ 0 expresses the opposite behavior. A large value of the absolute value of the coefficient cp indicates a strong correlation, while a small value of the absolute value of the coefficient rp indicates a weak correlation. Figure 2 shows the Pearson Correlation Matrix among the different variables. It can be seen that there are no variables with strong correlation with the 3 target variables, but there are a structure of correlations between them.

3.3. Data Preprocessing

As an initial step, it was verified that certain variables have null values: United States Purchasing Management Indicator, China's Gross Domestic Product, US Federal Reserve Interest Rate, China's Inflation Rate, US Dollar and Yuan Exchange Rate and Steel Price. From this group of variables, it was decided to remove the Steel Price variable because it presents a high percentage of null values (93%) and for the rest of the variables, linear interpolation was applied to complete the null values.

We also applied a standardization procedure to all variables (Equation 1), since they present very different magnitudes.

$$X_{i(standardized)} = \frac{X_i - \mu}{\sigma} \tag{1}$$

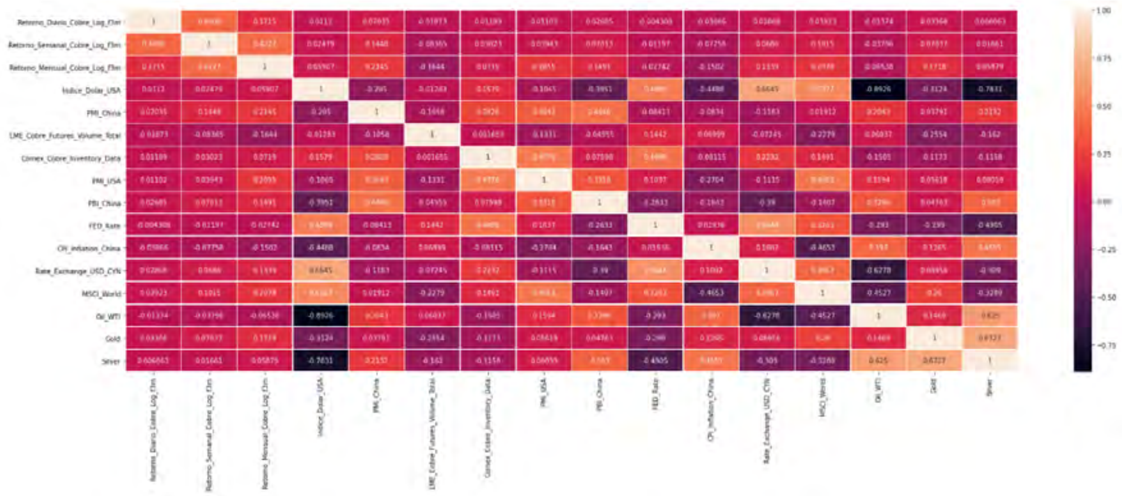


Figure 2.: Pearson Correlation Matrix.

Where:

- μ: mean of the variable X_i.
- σ: standard deviation of the variable X_i.

3.4. Copper price returns

The future price of Copper plays an important role in the market for metallic commodities worldwide. The return of the cooper prices is a useful indicator for investors. For a given day t, the logarithmic price return of the cooper price at w days ahead with respect to day "t" is calculated as Equation (2):

$$R_t = \ln(\frac{P_{t+w}}{P_t}) \tag{2}$$

Where P_{t+w} is the price at day $t + w$

3.5. Algorithms

Multi Layer Perceptron (MLP) Model)
An MLP network is a type of artificial neural network (ANN), which receives inputs, changes its internal states according to its inputs, and then computes outputs based on the inputs and internal states. These artificial neurons have weights that can be modified through a learning process Hu et al. (2020).

Convolutional Neural Network (CNN) Model)
Convolutional and pooling layers [23] are specially designed data preprocessing layers which process the input data and extract useful information (feature maps) by using a bank of convolutional filters. That information is further processed by fully connected neural network layers that produce the final prediction.

Long Short-Term Memory (LSTM) Model

LSTM is a kind of classic Recurrent Neural Networks (RNN), which can deal with problems of the gradient explosion and fading. It is normally made up of recurring gates. Unlike classical neural networks (ANNs), LSTM has the natural ability to learn temporal patterns and it can also handle inputs or signals that have both low and high frequency components. LSTM networks are effective at identifying dependencies in the short and long term Hu et al. (2020).

3.6. Forecasting Models

In the present work, the models have been divided into 3 categories: persistence models, univariate models and multivariate models. For the univariate and multivariate models, 3 types of neural networks have been used: Multi Layer Perceptron (MLP), Convolutional Neural Networks (CNN) and Recurrent Neural Network (RNN)(specifically using LSTM: Long Short Term Memory)

The developed models are described below:

(1) **Persistence Models**
In a first set of experiments, a baseline has been established, which consists of daily persistence models for the 3 forecast target variables, respectively. For the persistence models, the following models have been considered for each target variable:

  (a) Persistence model based on the latest Copper Return value (PST_UV)
  (b) Persistence Model based on the average of the last 3 Copper Return values (PST_3P)
  (c) Persistence Model based on the average of the last 5 Copper Return values (PST_5P)

(2) **Univariate Models**
In a second group of experiments, it has been established to work with univariate models, which means that only the same variable of interest is used to predict the respective target variable. For the univariate models, the following models have been considered for each target variable:

  (a) Univariate Model MLP / One-Step (MLP_MU)
  (b) Univariate Model CNN / One-Step (CNN_MU)
  (c) Univariate Model RNN / One-Step (RNN_MU)

(3) **Multivariate Models**
In a third group of experiments, it has been established to work with multivariate models, which means that to predict the respective target variable, the same variable of interest is used together with other predictor variables. In the multivariate models, there are 2 scenarios: a single multivariate model forecasts the 3 target variables simultaneously and three multivariate models that forecast each target variable separately and independently. For the multivariate models, the following models have been considered for each target variable:

(a) **Multivariate Multiple Output Model**
   (i) Multivariate Model MLP Multiple Output / One-Step (MLP_MM_MO)
   (ii) Multivariate Model CNN Multiple Output / One-Step (CNN_MM_MO)
   (iii) Multivariate Model RNN Multiple Output / One-Step (RNN_MM_MO)

(b) **Multivariate Single Output Model**
   (i) Multivariate Model MLP Single Output / One-Step (MLP_MM_SO)
   (ii) Multivariate Model CNN Single Output / One-Step (CNN_MM_SO)
   (iii) Multivariate Model RNN Single Output / One-Step (RNN_MM_SO)

For the multivariate models and considering that the predictor variables that are added to these models are in different magnitudes, a standardization has been applied to all the variables of the dataset.

For the group of persistence, univariate and multivariate models with a single output, and for the group of multivariate models with multiple output, the following nomenclature has been established:

$$Model_i \tag{3}$$

Where:

- i = {0, 1, 2}
  - 0: Model that individually forecasts Series 0 (one-day logarithmic return)
  - 1: Model that individually forecasts Series 1 (one-week logarithmic return)
  - 2: Model that individually forecasts Series 2 (one-month logarithmic return)

$$Model_{jkl_i} \tag{4}$$

Where:

- jkl = {012}, for i = {0, 1, 2}
  - 012-0: Multiple Output Model forecasts Series 0 (one-day logarithmic return)
  - 012-1: Multiple Output Model forecasts Series 1 (one-week logarithmic return)
  - 012-2: Multiple Output Model forecasts Series 2 (one-month logarithmic return)

The table 5 summarizes the topologies implemented by each model type.

4. Experimentation and Results

4.1. Experimental Setup

4.1.1. Model Evaluation Metrics

To evaluate and compare the performance of predictive models, the following error prediction metric is used: Root Mean Squared Error (RMSE). Equation 5 shows the formula to calculate it.

Table 5.: Specification of network topologies for each type of model

Model	**Description**
MLP_MU_i	1 Dense layer (hidden), optimizer: adam, loss: mse
$MLP\_MM\_SO_i$	1 Dense layer (hidden), optimizer: adam, loss: mse
$MLP\_MM\_MO_{jkl}$	1 Dense layer (hidden), optimizer: adam, loss: mse
CNN_MU_i	1 1D Convolutional layer, 1 1D MaxPooling layer of size 1, 1 Flatenn layer, 1 Dense layer of 50 neurons, optimizer: adam, loss: mse
$CNN\_MM\_SO_i$	1 1D Convolutional layer, 1 1D Convolutional layer, 1 1D MaxPooling layer of size 3, 1 1D Convolutional layer, 1 1D MaxPooling layer of size 3, 1 Flatten layer, 1 Dense layer of 100 neurons, optimizer: adam, loss: mse
$CNN\_MM\_MO_{jkl}$	1 1D Convolutional layer, 1 1D Convolutional layer, 1 1D MaxPooling layer of size 3, 1 1D Convolutional layer, 1 1 1D MaxPooling layer of size 3, 1 Flatten layer, 1 Dense layer of 100 neurons, optimizer: adam, loss: mse
RNN_MU_i	1 LSTM layer, 1 Dense layer of 50 neurons, optimizer: adam, loss: mse
$RNN\_MM\_SO_i$	1 LSTM layer, 1 LSTM layer, 1 Dense layer of 50 neurons, optimizer: adam, loss: mse
$RNN\_MM\_MO_{jkl}$	1 LSTM layer, 1 LSTM layer, 1 Dense layer of 50 neurons, optimizer: adam, loss: mse

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(R_t) - \hat{R}_t)^2} \tag{5}$$

Where:

- n: Number of Predictions.
- R_t: Real Copper Return Series.
- $\hat{R}_t$: Forecasted Copper Return Series.

4.1.2. Hyperparameters Optimization

Many hyperparameters need to be configured and optimized during model training to achieve good forecast performance. After performing the optimization of the hyperparameters for each model, the best final hyperparameter configuration obtained for each experiment is shown in the tables 6, 7 and 8.

All models were trained with the Adaptive Moment (ADAM) optimizer using the MSE (Mean Squared Error) loss function. The ADAM optimizer ensures that the learning steps, during the training process, do not rescale relative to the gradients of the parameters. In this study, 4 different values of input time window (steps in) were tried: 3, 5, 7 and 9 days. The range of values for the batch size was 32 and 64. Likewise, 2 different values have been used for the number of epochs: 100 and 200.

4.1.3. Computational Environment

The model implementation and experiments were done in Python programming language on a computer with the following characteristics: Intel(R) Core(TM) i7-950H

Table 6.: Optimal hyperparameter configuration for MLP models

Model	n_steps_in	n_steps_out	n_neurons	n_epochs	n_batch	process_time (hrs)
MLP_MU_0	3	1	50	100	64	1.15
MLP_MU_1	7	1	50	100	64	1.19
MLP_MU_2	3	1	150	100	64	1.18
$MLP\_MM\_SO_0$	3	1	150	100	64	1.25
$MLP\_MM\_SO_1$	9	1	100	100	64	1.19
$MLP\_MM\_SO_2$	3	1	50	100	32	1.17
$MLP\_MM\_MO_{012-0}$	3	1	50	100	64	1.18
$MLP\_MM\_MO_{012-1}$	3	1	50	100	64	1.18
$MLP\_MM\_MO_{012-2}$	5	1	50	200	64	1.18

Table 7.: Optimal hyperparameter configuration for CNN models

Model	n_steps_in	n_steps_out	n_filters	n_kernel	n_epochs	n_batch	process_time (hrs)
CNN_MU_0	3	1	50	3	100	64	1.24
CNN_MU_1	7	1	50	3	100	64	1.29
CNN_MU_2	3	1	150	3	100	64	1.30
$CNN\_MM\_SO_0$	9	1	100	2	200	32	1.40
$CNN\_MM\_SO_1$	9	1	150	2	200	64	1.41
$CNN\_MM\_SO_2$	9	1	100	2	200	32	1.41
$CNN\_MM\_MO_{012-0}$	9	1	150	2	200	32	1.43
$CNN\_MM\_MO_{012-1}$	9	1	150	2	100	32	1.43
$CNN\_MM\_MO_{012-2}$	7	1	100	2	100	64	1.43

Table 8.: Optimal hyperparameter configuration for RNN models

Model	n_steps_in	n_steps_out	lstm_units	n_epochs	n_batch	process_time (hrs)
RNN_MU_0	7	1	50	100	64	4.03
RNN_MU_1	7	1	150	100	64	4.01
RNN_MU_2	9	1	50	100	32	3.98
$RNN\_MM\_SO_0$	3	1	150	200	32	7.03
$RNN\_MM\_SO_1$	5	1	150	200	32	7.21
$RNN\_MM\_SO_2$	5	1	100	200	32	7.30
$RNN\_MM\_MO_{012-0}$	5	1	150	200	32	6.93
$RNN\_MM\_MO_{012-1}$	5	1	150	200	32	6.93
$RNN\_MM\_MO_{012-2}$	3	1	50	100	64	6.93

Table 9.: RMSE value comparison table for the daily (i=0), weekly (i=1) and monthly (i=2) target series

	Dataset		
	Bloomberg		
Model	$RMSE_{i=0}$	$RMSE_{i=1}$	$RMSE_{i=2}$
PST_UV_i	1.39834	0.59271	0.26412
PST_3P_i	1.09633	0.70073	0.33049
PST_5P_i	1.05285	0.82580	0.40101
MLP_MU_i	0.98461	0.49797	0.26915
CNN_MU_i	0.99861	0.52433	0.26631
RNN_MU_i	**0.97190**	**0.49091**	**0.26401**
$MLP\_MM\_SO_i$	1.72130	1.09477	0.68686
$CNN\_MM\_SO_i$	1.19237	0.75802	0.57865
$RNN\_MM\_SO_i$	1.27573	1.05153	0.80799
$MLP\_MM\_MO_{012-i}$	1.742940	1.02287	0.69850
$CNN\_MM\_MO_{012-i}$	1.12892	0.74234	0.72531
$RNN\_MM\_MO_{012-i}$	1.38699	1.70517	0.50152

CPU 2.6 GHz (12 CPUs), RAM: 16 GB, Graphics Card: NVIDIA GeForce RTX 2070 16 GB under a Windows 10 operating system. We used the Keras an Tensorflow libraries for the neural network implementations.

4.2. Results and Discussion

4.2.1. Model Comparison

Based on the RMSE metric, all developed models are compared to find the model with the best performance. The best results for each target series are shown in the table 9. The comparison of results are shown in the figures 3, 4 y 5.

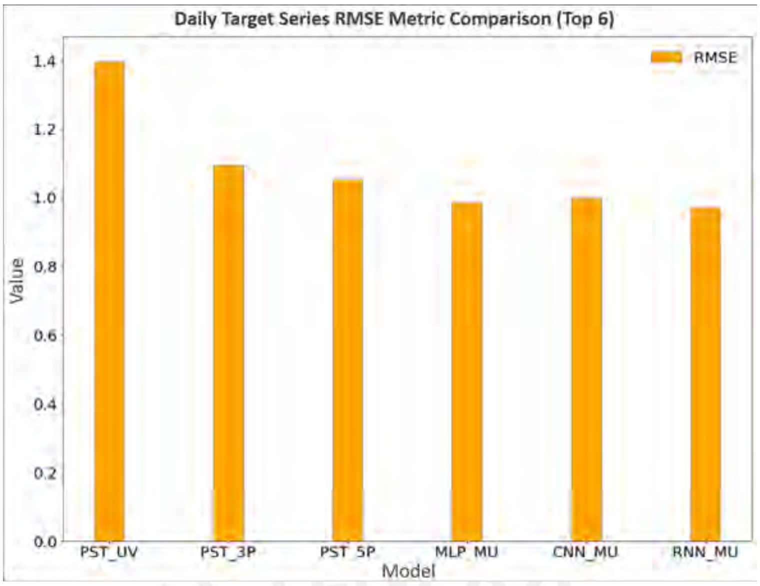


Figure 3.: RMSE Metric Comparison Table for Daily Target Series
RMSE values of the six best models for forecasting the Daily Log Return. **PST_UV**: Persistence Model based on Last Value; **PST_3P**: Persistence Model based on the Average of the Last 3 Values; **PST_5P**: Persistence Model based on the Average of the Last 5 Values; **MLP_MU**: Univariate MLP Model; **CNN_MU**: Univariate CNN Model; **RNN_MU**: Univariate RNN Model

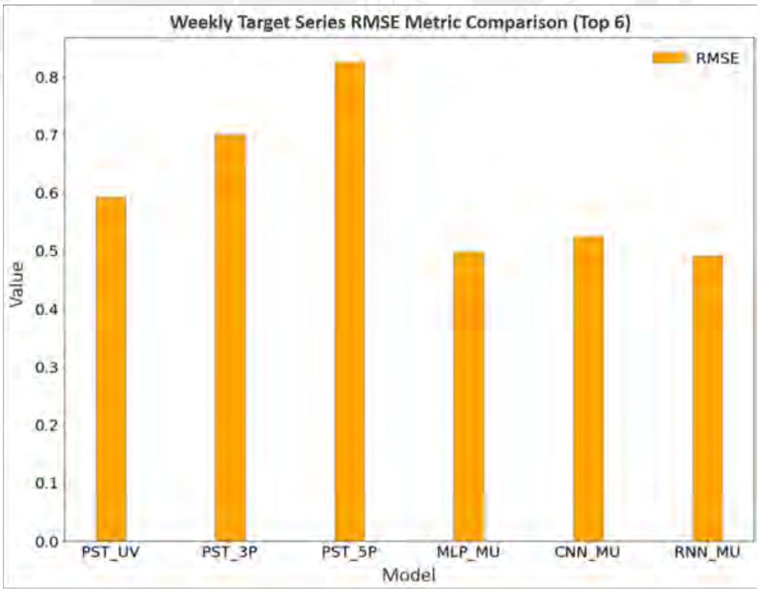


Figure 4.: RMSE Metric Comparison Table for Weekly Target Series
RMSE values of the six best models for forecasting the Weekly Log Return. **PST_UV**: Persistence Model based on Last Value; **PST_3P**: Persistence Model based on the Average of the Last 3 Values; **PST_5P**: Persistence Model based on the Average of the Last 5 Values; **MLP_MU**: Univariate MLP Model; **CNN_MU**: Univariate CNN Model; **RNN_MU**: Univariate RNN Model

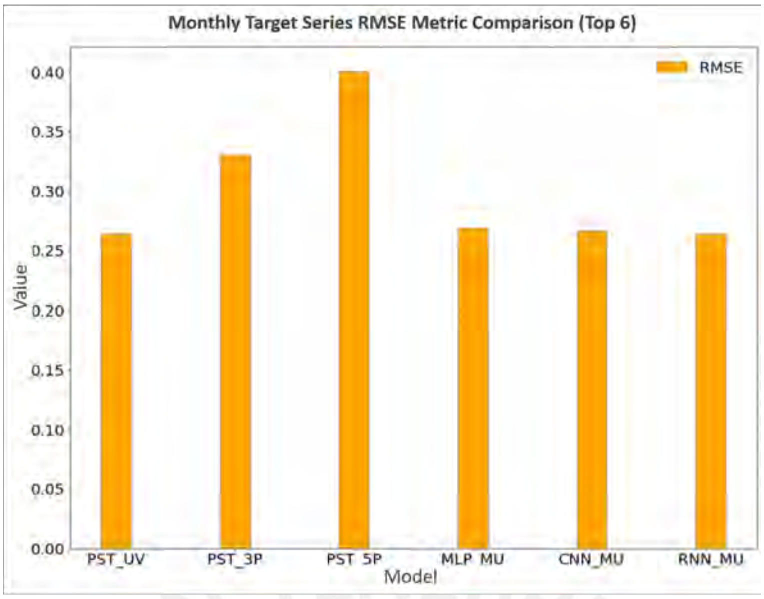


Figure 5.: RMSE Metric Comparison Table for Monthly Target Series
RMSE values of the six best models for forecasting the Weekly Log Return. **PST_UV**: Persistence Model based on Last Value; **PST_3P**: Persistence Model based on the Average of the Last 3 Values; **PST_5P**: Persistence Model based on the Average of the Last 5 Values; **MLP_MU**: Univariate MLP Model; **CNN_MU**: Univariate CNN Model; **RNN_MU**: Univariate RNN Model

According to the above results of table 9 and figures 3, 4 and 5 we summarize the following findings:

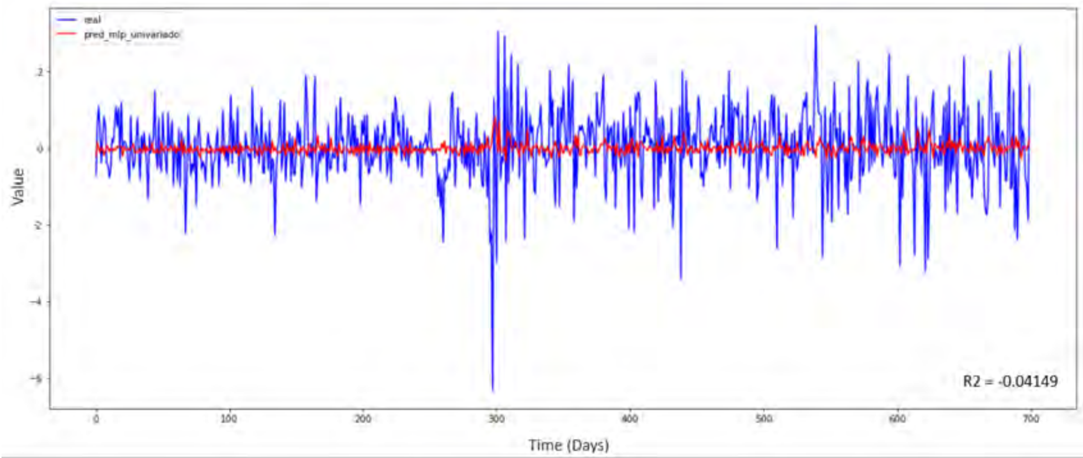
(1) For the one-day target series, the persistence model that performs best is the one that estimates the forecast as the average of the last 5 values. Regarding the one-week target series, the best model was the one that estimates the forecast as the last value. Finally, in the one-month series, the model that best estimates the forecast is the one based on the last value.
(2) The best performing models among all evaluated models are the univariate recurrent models.

4.2.2. Models Forecast

For each model, the experimentation was divided into 3 parts. First, each model was trained on the training dataset and a preliminary evaluation of the model was made. Then, the hyperparameters were optimized, evaluating in each possible combination the predictive capacity of the model through the performance metric and obtaining a final version of the model. Then, using the test dataset, the trained model with the best hyperparameters was used to perform the forecast. Finally, the actual data was compared with the predicted data for the test dataset. The comparison of the prediction results for the 3 best performing models for each target series are shown in the figures. 6, 7 y 8.

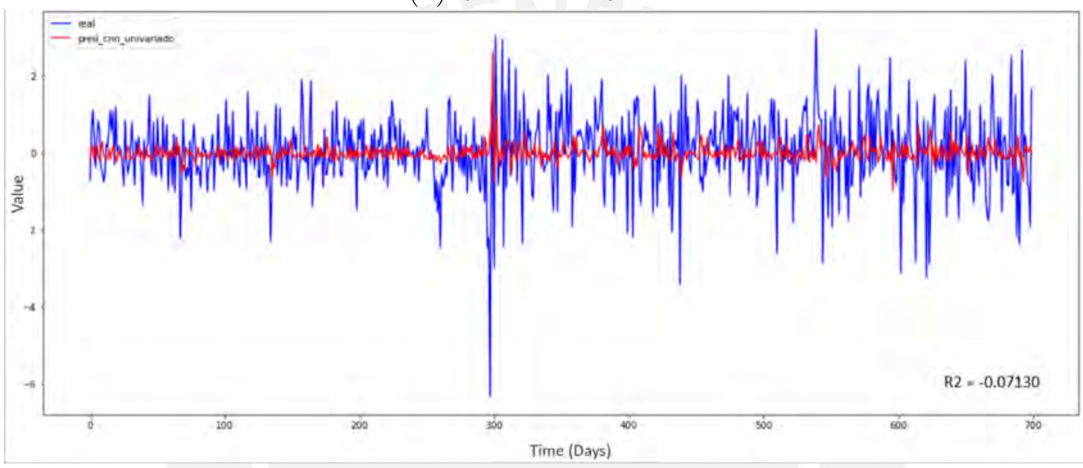
The figure 6 shows the results of the prediction and compared with the real data for the 3 models with the best performance for the Daily Target Series: Univariate MLP, Univariate CNN and Univariate RNN. It can be seen that none of the models that forecast one-day returns has a good fit. This can be verified by the negative values of the Coefficient of Determination (also known as R^2), which could indicate that it does not fit adequately to the real data. Among these models, taking into account only the value of the metric, the Univariate RNN model is the one that presents the best performance, which can be seen in the figure 6(c). Also, after analyzing the variability of the graph, it can be seen that the series has several peculiarities that should be taken into account: it can be seen that around point 300 of the series, there is an important variation peak with respect to the others points and, in turn, that from point 300 onwards there is greater variability in the series with respect to the section corresponding to the beginning of the series up to point 300. It has been identified that point 300 of the series corresponds approximately to the beginning of the month of February of the 2022, and this date coincides with the start of the COVID-19 pandemic. The impact of COVID-19 on the behavior of the copper price can clearly be observed for the daily logarithmic return series and, therefore, the possibility of dividing the pre- and post-pandemic analysis could be considered in future works to obtain better results in the analysis of the daily target series forecast.

(a) Univariate MLP



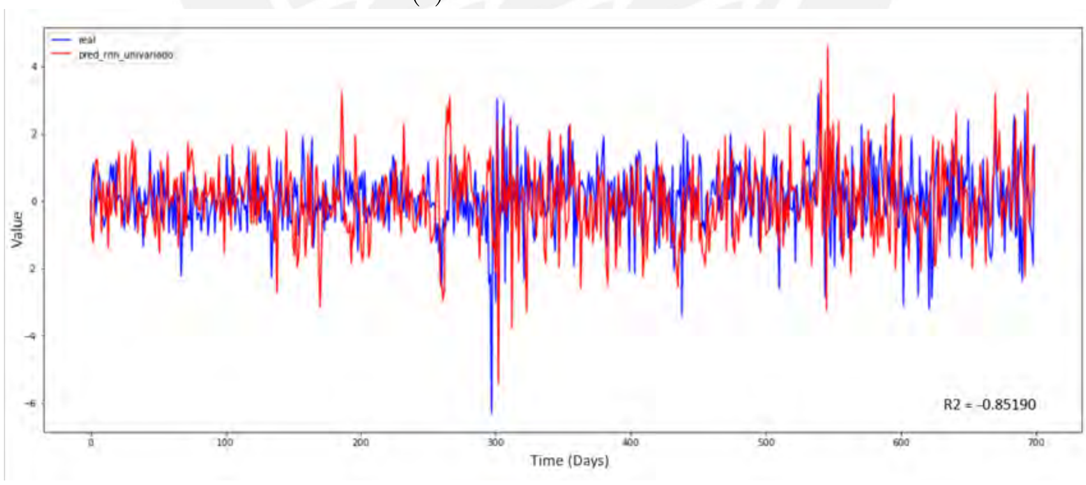
.

(b) Univariate CNN



.

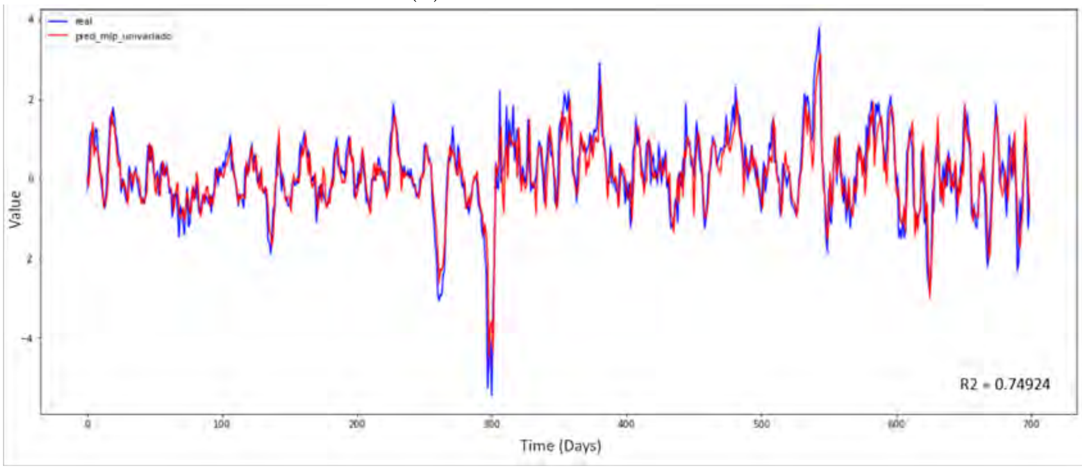
(c) Univariate RNN



.

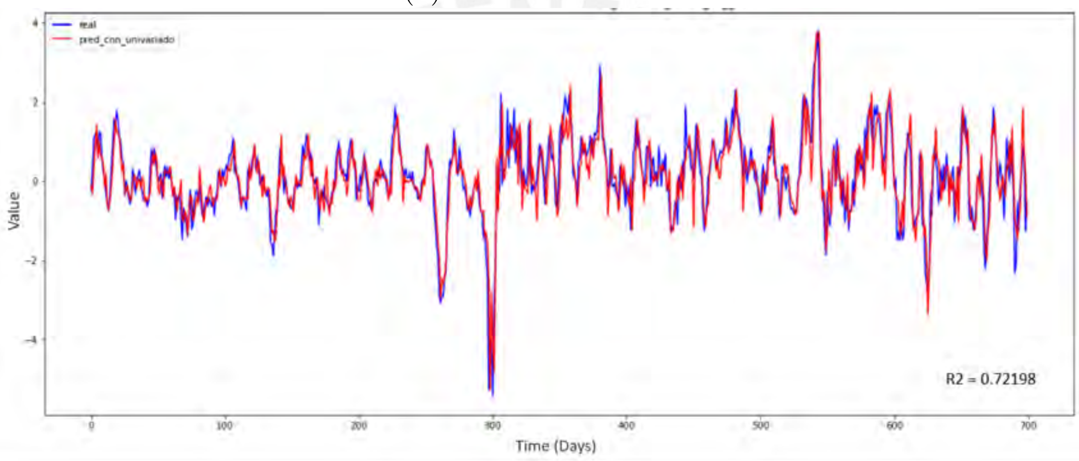
Figure 6.: Actual vs. forecasted values in test data of the three univariate models that forecast one-day ahead of cooper price returns.

(a) Univariate MLP



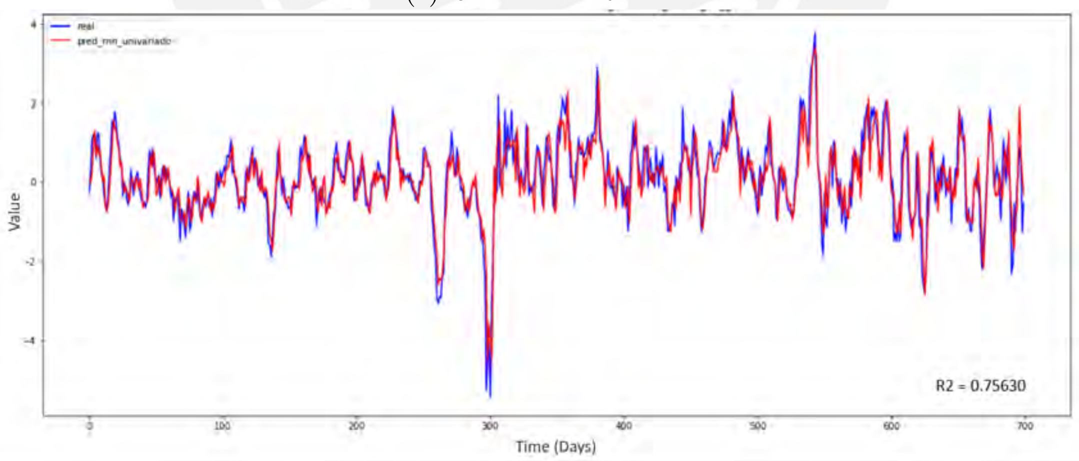
.

(b) Univariate CNN



.

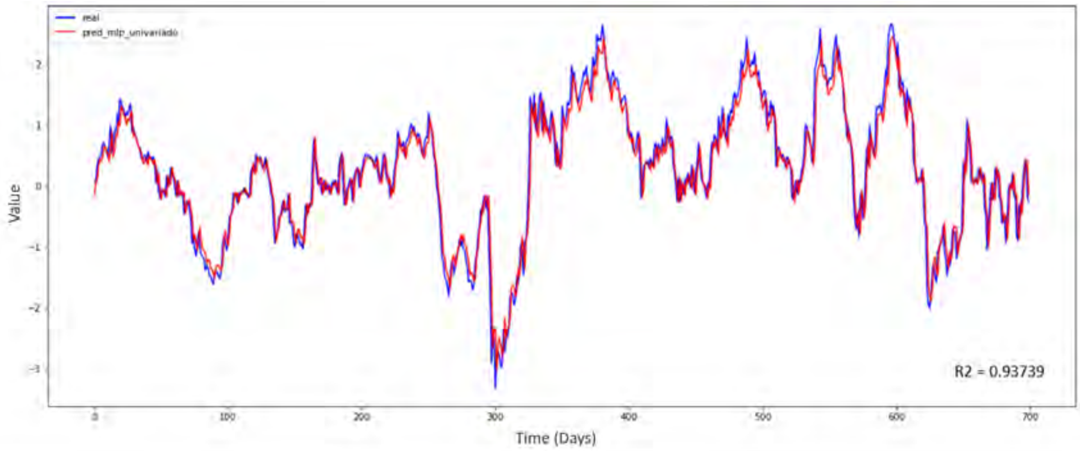
(c) Univariate RNN



.

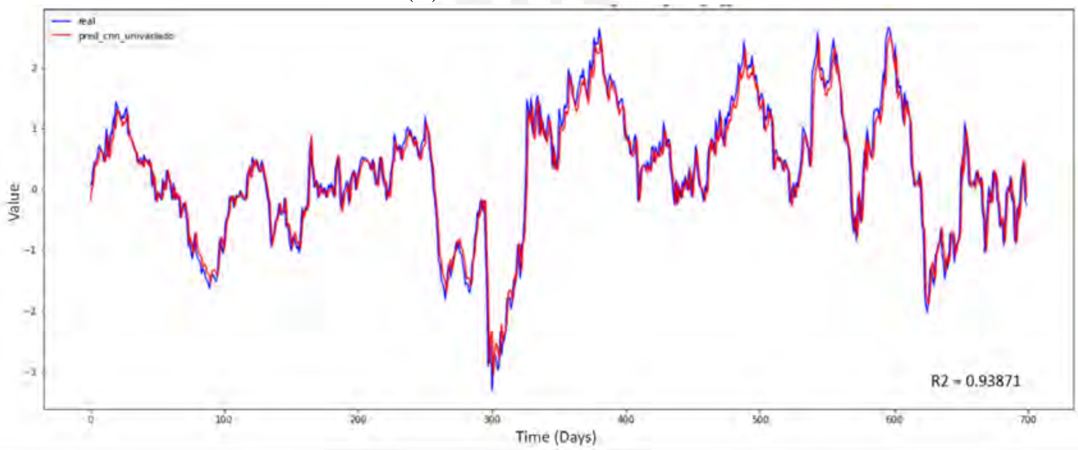
Figure 7.: Actual vs. forecasted values in test data of the three univariate models that forecast one-week ahead of cooper price returns

(a) Univariate MLP



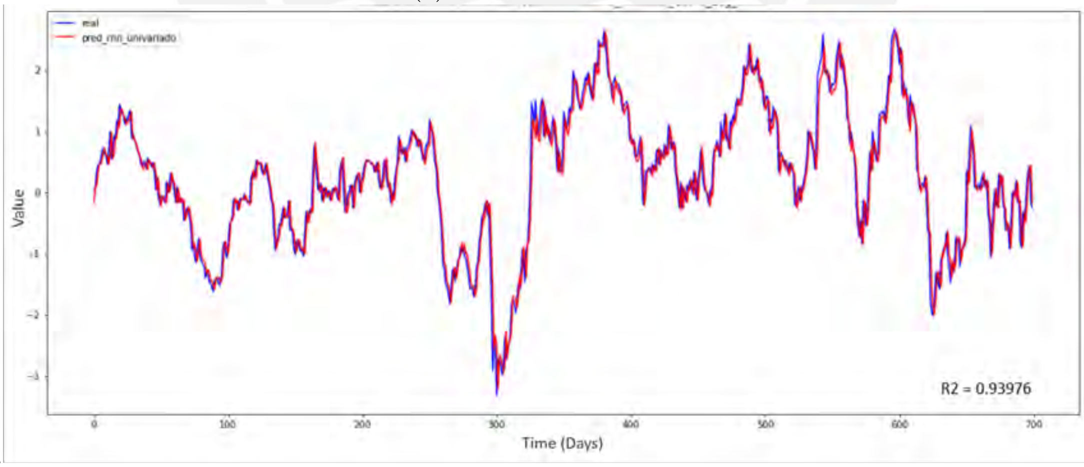
.

(b) Univariate CNN



.

(c) Univariate RNN



.

Figure 8.: Actual vs. forecasted values in test data of the three univariate models that forecast one-month ahead of cooper price returns

.

On the other hand, in figure 7, the results of the forecast are displayed and compared with the real data for the 3 models with the best performance for the one-week return series: Univariate MLP, Univariate CNN and Univariate LSTM. It can be seen that the Univariate LSTM model in figure 7(c) is the one that obtained the best performance and the one that, in turn, has an acceptable R^2 value; consequently, this model is the best fit for the trend of the real data as a whole, indicating that it successfully learns and captures the variability of the dataset. Finally, in figure 8, the results of the prediction are shown and compared with the real data for the 3 models with the best performance for the one-month returns series: Univariate MLP, Univariate CNN and Univariate LSTM. For this series, it can be seen in figure 8(a), that the prediction curve of the Univariate LSTM model is the one that obtains the best performance and the one that also has a value of R^2 very close to 1, which indicates that it expresses very well the adjustment to the curve of real data. Therefore, this model is the one that best fits the real data among the 3 forecast temporalities studied.

5. Conclusions

This work proposed univariate and multivariate models based on MLP, CNN and LSTM architectures to forecast the logarithmic return of copper prices at three different time horizons: one-day, one-week and one-month ahead. The univariate forecasting models predict the target using past data of the same variable. The multivariate models allow several variables to be predicted simultaneously using historical data of the observed variables. The experimental results revealed that, in general, the univariate LSTM model exhibit the highest predictive capacity and better efficiency (lower RMSE value). However, the prediction of one-day ahead was the most difficult task, as the performance indices were low. The forecasting of one-week and one-month returns exhibit a good performance, verified by a high value of the R^2 index. We also found that the incorporation of additional variables did not improve the prediction quality of the models. This may be due to the large number of parameters that have to be adjusted when introducing more variables and probably the amount of data was insufficient to have a good fit. A previous process of dimensional reduction could be useful to improve the results of the multivariate models.

The initial hypothesis that a single model predicting all three time horizons simultaneously is better than having three separate models has not been verified in practice. The same explanation as above can be outlined here, probably the increased complexity of having a single model is not advantageous due to the limited amount of data.

We recommend for future research explore new neural network architectures that have recently exhibited promising results in other domains, such as attention-based models. In the financial context, the works by Zhang et al. and Ju and Liu (2021) described applications of forecasting models using these models with interesting results.

Acknowledgements

This project has received financial support from the Pontifical Catholic University of Peru (CAP project PI0762). We would like to thanks the support of *Julio Villavicencio* and *Oscar Miranda*, professors of Pontifical Catholic University of Peru by their advise in the project.

References

Z. Alameer, M. A. Elaziz, A. A. Ewees, H. Ye, and Z. Jianhua. Forecasting gold price fluctuations using improved multilayer perceptron neural network and whale optimization algorithm. *Resources Policy*, 61:250–260, 2019a. ISSN 0301-4207.

Z. Alameer, M. Elsayed Abd Elaziz, A. Ewees, H. Ye, and Z. Jianhua. Forecasting copper prices using hybrid adaptive neuro-fuzzy inference system and genetic algorithms. *Natural Resources Research*, page 1385–1401, 02 2019b.

J. D. Díaz, E. Hansen, and G. Cabrera. A random walk through the trees: Forecasting copper prices using decision learning methods. *Resources Policy*, 69:101859, 2020.

Y. Hu, J. Ni, and L. Wen. A hybrid deep learning approach by integrating LSTM-ANN networks with GARCH model for copper price volatility prediction. *Physica A Statistical Mechanics and its Applications*, 557:124907, Nov. 2020. .

E. Jianwei, J. Ye, and H. Jin. A novel hybrid model on the prediction of time series and its application for the gold price analysis and forecasting. *Physica A: Statistical Mechanics and its Applications*, 527:121454, 05 2019. .

J. Ju and F.-A. Liu. Multivariate time series data prediction based on ATT-LSTM network. *Applied Sciences - MDPI*, 2021.

W. Kristjanpoller and M. C. Minutolo. Gold price volatility: A forecasting approach using the artificial neural network–garch model. *Expert Systems with Applications*, 42(20):7245–7251, 2015.

I. Livieris, E. G. Pintelas, and P. Pintelas. A cnn-lstm model for gold price time-series forecasting. *Neural Computing and Applications*, pages 1–10, 2020.

S. A. A. Sabour and R. Poulin. Valuing real capital investments using the least-squares monte carlo method. *The Engineering Economist*, 51(2):141–160, 2006. .

C. Schröer, F. Kruse, and J. M. Gómez. A systematic literature review on applying crisp-dm process model. *Procedia Computer Science*, 181:526–534, 2021. ISSN 1877-0509. . URL `https://www.sciencedirect.com/science/article/pii/S1877050921002416`. CENTERIS 2020 - International Conference on ENTERprise Information Systems / ProjMAN 2020 - International Conference on Project MANagement / HCist 2020 - International Conference on Health and Social Care Information Systems and Technologies 2020, CENTERIS/ProjMAN/HCist 2020.

C. WANG, L. shui ZUO, P. jie HU, H. lin YAO, and H. ZHU. Evaluation and simulation analysis of china's copper security evolution trajectory. *Transactions of Nonferrous Metals Society of China*, 23(8):2465–2474, 2013. ISSN 1003-6326. . URL `https://www.sciencedirect.com/science/article/pii/S1003632613627569`.

S. Xiong, C. Wang, Z. Fang, and D. Ma. Multi-step-ahead carbon price forecasting based on variational mode decomposition and fast multi-output relevance vector regression optimized by the multi-objective whale optimization algorithm. *Energies*, 12(1), JAN 1 2019. ISSN 1996-1073. .

X. Zhang, X. Liang, A. Zhiyuli, S. Zhang, R. Xu, and B. Wu. AT-LSTM: An attention-based LSTM model for financial time series prediction. *IOP Conference Series: Materials Science and Engineering.*