ISSN 1816-6075 (Print), 1818-0523 (Online)
Journal of System and Management Sciences
Vol. 13 (2023) No. 5, pp. 585-601
DOI:10.33168/JSMS.2023.0538

Enhancement of Transaction Management Strategies for Volume Forecasting in Commodity Trading Based on Integrated Neural Prophet and Anomaly Detection

Barani Shaju, N. Valliammal

Department of computer science, Avinashilingam institute for home science And higher education for women, Coimbatore, India

Abstract. Transaction management in commodity trading operates based on trading strategies and traded volumes. The current study aims to enhance commodity trading benefits by raising the degree of effectiveness in transaction management strategies. The outcome of this study is derived based on commodity trading for Nickel by means of forecasting trading volume in lot sizes with anomaly detection and evading white noises. Novelty of this approach revolves around removal of white noise, thus results in normalized data patterns. To manage transaction volume, key element in commodity trading is to focus on the lot size where it refers to the volume or units per transaction. Learning derivative of this approach indicates that calculative tradeoff can be made by traders based on their ability to forecast trading volumes. With non-linear factors that impact trading volumes, this approach is proposed to predict future values of transactional volume using NeuralProphet with triggers to fit, predict and options to create future timeframes with seasonality after normalizing anomalies. Evading the outliers and white noise along with focus on seasonality brings ease of focus on the business implementation.

Keywords: commodity trading, time series, volume forecasting, anomaly – outliers / noise detection, cross validation - prediction model.

1. Introduction

Commodity trading is an approach followed from ancient days to current era. The modern way of handling trade focuses on investment portfolio of commodity market. In commodity trading, the trader enters into a contractual agreement with using Futures or Options contract to transact in units of business volume. The trader either transacts against purchase and sale of physical goods or in today's market based on contract using Lot Size or contractual volume for a particular commodity. Execution of contractual deals based on market dynamics has a directly proportional impact on profit making. This model aims at forecasting trading volume for commodity contracts based on time series analysis. This approach focuses on time series predictions on cleansed data as outliers or anomaly are removed before exploratory data analysis. Early indicators of noise, noise detection and cleansing of data before forecasting increases the efficiency of the model to forecast trading volumes.

The key success factor for effective transaction management relies on high returns within specific turnaround timelines. In commodity trading market, transaction management approaches are devised and strategized by the trader based on traded volume and lot size of the purchase or sale during each commodity bid investment. Considering current economic situations in commodity market, its trends and variations, direct use of straight forward or traditional prediction models may not provide a defensible solution.

Forecasting based on time series has gathered momentum as decision making is supported based on data based derivatives.

Research studies indicate for attributes that continuous and based on trend / time series, it is recommended to use Auto Regressive Integrated Moving Average (ARIMA). This technique uses analytics based on history of data over series of time. This model is based on grey forecasting framework and well formulated statistical models. Forecasting models enable decision making to be done based on facts, numbers, past experiences, trend over time. It is a fact that contractual trading of commodities is a very dynamic market and investment decision either to purchase or sell suitable commodity is hard to make without analysis. Seasoned traders use various strategies and forecasting model helps them with various degree of information.

Time series refers to collection of statistical data in chronological order. Time Series Analysis refers to a mechanism of predicting the data for the future, based on historical data of past time period. Volume of commodity trade is determined based on price and quantity or the lot size of transactions. When decision making is backed based on the data insights, quality of decision making of the trader in response to market dynamics will yield long term benefits. The choice to apply this library "Neural Prophet", integrated with a layer of anomaly detection has resulted in valuable outcomes. In comparison to traditional neural network and LSTM methods, it is observed, using cross validation measures of this prediction model brings reduction in error rate based on this research study.

With advent of artificial intelligence and technological advancements, there is an increasing demand for forecasting models for investors. There is a lot of focus in the area of price and volume predictions in the field of stock trading. With Artificial Neural Network (ANN), focus on varying dynamic traits has gathered momentum. But, performance efficiency of the model versus tuning the system with volume of data will be trade off using ANN. Enhancements to traditional approaches are brought in with use of Recurrent Neural Network (RNN). RNN brings the balance to forecasting model as a system with human state of mind behavioral patterns using well-defined learning models.

Section I of this study sets the context of study with an Introduction followed by Section II, which provides the consolidations of the literature survey based on existing approaches. Section III details about the working of the forecasting model using time series and Section IV tabulates the comparative benefits and enhancements based on the efficiency of the model. It also indicates the drop in the error rate thereby helping investors and traders to take guided directions towards effective transaction management and decision making on trading volumes. Section V summarizes the derivative from this study and research with resulting outcomes along with additional suggestions and recommendations for further study or research work.

2. Literature Survey

Ghosh et al. (2020) has developed a novel hybrid granular ensemble model that predicts closing prices one day ahead on the basis for various essential commodities like metals, oil and gas. The study describes use of ensemble machine learning frameworks, random forest and bagging for making predictions on granular subcomponents. This paper presents details on model's ability to forecast figures for respective commodities using ELM (Extreme Learning Machine). Trading dataset of this study uses daily closing price of various metals, oil and gas from July 2014 to August 2019 for predictive modeling exercise. This study made from this system indicates aspects related to forecast model of selected commodities with an objective to diversify risk. This framework was evaluated to predict the future closing price of commodity markets for various horizons of time ranging from 5 minutes to hourly time intervals. This study helps us to consolidate learning with results and prediction made in consideration of timespan of the selected commodities.

Merkel et al. (2017) applied methodology based on deep neural network for predicting prices for natural gas using short-term load. This study is experimented using historical data that represent a variety of climates across 62 operating areas scoping wide area of geographical regions in U.S. local distribution companies. The proposed model from the study was evaluated in comparison to traditional models such as linear regression and artificial neural network. Derived outcome from this numerical model indicates that the use of this deep learning model exhibits higher efficiency using short-term load of forecasts on average in comparison to traditional approaches. In the process of evaluation, during certain climatic contexts simpler approaches like direct linear regression model showed outperforming predictions. Though there may be certain contextual relationship to climatic context in this study, various parameters used in this model for forecasting can be used as basis for understanding of a system under a context for study, though mapping to every other operating area cannot be mapped directly.

Othman et al. (2020) used Artificial Neural Network model based on Rapid-Miner Programme for development of a prediction model for Bitcoin price index. The input data feed for the ANN model are based on symmetric volatility information of bitcoin currency with data over the period of 5+ years during 2014 – 2019. Experimental results of this study indicates that Bitcoin prices can be predicted in a significant and efficient manner employing ANN model. More specifically, the predicting accuracy of Bitcoin price index is adequate and accounts for very high accuracy levels to the actual price, whereas low-price attribute has been found to be the key predictor of Bitcoin price trend, scoring 63%, next followed by close price, high price, and open price with percentages of 49%, 46%, and 37%, respectively.

Weng et al. (2019) describes use of genetic algorithm and regularized extreme learning machine to calculate the optimal input layer weight matrix and hidden layer neurons of the gold price prediction model. This reduces the impact of its randomness on the prediction results on the sequential update stage. With introduction of the regularization factor, dependency of the model on the number of hidden layer nodes has reduced. This improves the generalization ability of the prediction model. Data set over period from July 6, 2010, to July 4, 2017 was collated with different parameters viz., opening price of gold and other comparison commodities per day and its respective index factors. Study indicates the output of this system generates predictions of the current opening price of the gold. As part of the study, author compares genetic algorithm regularization online extreme learning machine (GA-ROSELM) model with Autoregressive Integrated Moving Average model (ARIMA), Support vector machine (SVM), BP neural network and traditional extreme learning approaches with results that shows the effectiveness of the proposed model where the observations from the literature study is given in table 1.

Table 1: Literature Study Observations

Author Details	Paper	Journal Title	Key element observed from this study
Ghosh et al. (2020)	An Ensemble of Framework for Predictive Analytics of Commodity Market	2020: 4th International Conference on Computational Intelligence and Networks (CINE)	This study indicates use of novel hybrid granular ensemble model of forecasting framework that predicts closing prices on one day ahead basis for various metals, oil and gas commodities. It describes use of ensemble machine learning frameworks, random forest and bagging for making predictions on granular subcomponents.
Merkel et al. (2018)	Deep Neural Network Regression for Short-Term Load Forecasting of Natural Gas	International Symposium on Forecasting	This study is based on ANN & linear regression based forecasting model. This study applied deep neural network methodologies for predicting natural gas short-term load. The authors experimented using historical data that represent a variety of climates across 62 operating areas scoping wide area of geographical regions in U.S. local distribution companies.
Othman et al. (2020)	Prediction accuracy improvement for Bitcoin market prices based on symmetric volatility information using artificial neural network approach	Journal of Revenue & Pricing Management vol.19 (2020)	This study has used Artificial Neural Network (ANN) model based on Rapid-Miner Programme for development of a prediction model for Bitcoin price index. The input feed for the ANN model are based on symmetric volatility information of bitcoin currency with data over the period of 5+ years during 2014 – 2019.
Weng et al. (2019)	Gold-price forecast research based on improved online-extreme machine learning algorithm	Journal of Ambient Intelligence & Humanized Computing Vol. 11, (2020)	This study describes use of genetic algorithm and regularized extreme learning machine to calculate the optimal input layer weight matrix and hidden layer neurons of the gold price prediction model. This reduces the impact of its randomness on the prediction results on the sequential update stage.

3. Motivation and Methodology

3.1. Motivation

Today, commodity market is in a widely prevalent zone for investment trading. Here, a trader or stockist agrees on a contractual trade (either to buy or sell) a specific commodity of a defined lot size. Commodity contract can be categorized as futures and options with respective contractual terms on delivery settlement either in form of currency or physical delivery of the product. In case of options contract, required margins based on the lot size should be maintained by the trader. The aim of this study was motivated with the advent of technology trends which can be used to bring in cross functional benefit of integration predictive modelling in the domain of commodity trading investment strategies.

This study uses data set of Nickel commodity under metal category from Metal Commodity Exchange (MCX). The system focuses on transactional volume over time period from 2004 to 2022. Learnings from this study and forecasting model can provide accelerators and decision making indicators to investor or trader to strategize on transaction management. This can help them in planning margins for trade in case of options or future contract. This forecasting model is built on time series analysis taken at equal intervals, which is in reference to each trading day over the last 18 years. Market dynamics have an influential impact on commodity market prices which vary on day –today basis according to their supply and demand (Hopman, 2007). Considering the same, this model built based on time series considers seasonality of data, evaluates white noise and anomalies while deriving at predictions with use of an open source library from python.

Data volume from Nickel trading in Metal Commodity Exchange (MCX) is used for experimental basis from period Jun'04 to Sep'22. NeuralProphet library works on time series based on core elements which are Signal and Noise. This library is designed and best suited for forecasting univariate time series by decomposing time series into segments or pieces as in the similar approach to model of exponential smoothing. Basic assumption of the time series is built on the assumptions that the observations at a certain point in time depends on the observations on previous time period(s). The success rate of this mechanism is high as it applies weighting scheme that decreases exponentially to back date of time periods. This helps to smoothen signal data with exponential window function and apply low pass filters to evade or minimize the impact of high frequency noise. Figure 1 depicts flowchart describing flow of activities used in building this prediction model for volume forecasting.

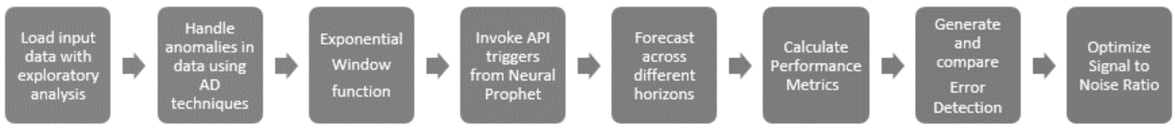


Fig. 1: Flowchart of activities involved in below algorithm

This model is built for prediction of transactional volume of Nickel commodity trade. Various strategies using combination of dropout layers for regularization, detection of white noise to ensure time series data points are suitable for the predictive model and trigger calls to API enhanced with anomaly detection followed by cross validation makes this model stronger and provide defensive results.

This model aims to provide the trader with a foresight on transactional volumes of trade for a specific period based on the parameter set as Daily, Weekly, Fortnightly or Monthly. In cases, where future contracts are to be closed by end of each month (Last Thursday of each month as in MCX trade), this forecast can help the trader to strategize decision making and make insightful decision.

3.2. Methodology

This section discusses about the methodology used to build the forecasting model to predict volume, which refers to traded lot size in terms of Nickel Commodity trading contracts. This is built upon time series analysis using NeuralProphet library. NeuralProphet has improvisations benefits on Prophet (Base version as released by Facebook called as "fbprophet) as it enhanced with greater degree of extensibility with add-on features to the framework. Researcher Pandey suggests that prediction in stock market is an approach in which historical information can be looped as basis for feedback construction in identifying and foretelling or deciding the direction of future than traditional methods using fbprophet with higher accuracy (Taylor & Letham, 2018). But, in comparison to fbprophet, aspects related to missing local context for predictions has been addressed in NeuralProphet thereby increases forecast accuracy. NeuralProphet also has inherent benefits in terms of scalability. This framework is built entirely on Pytorch, which brings ease to use. It is designed standard deep learning methods and techniques. With use of gradient descent from Pytorch for optimization this approach results in creating modeling much faster. The core of the framework focusses on time-series autocorrelation using Auto-Regressive Network. Feed-Forward Neural Network is used separately to create lagged regressors.

This approach aims to strengthen and scale the key performance indicators with an integrated layer of anomaly detection. This helps the model to detect white noise and outliers. The outcome of the model demonstrates efficiency enhancement in prediction of traded volume for metal commodity viz., Nickel.

Cleansing the dataset and detection of anomalies along with this integrated approach has improved accuracy in prediction levels. Considering the non-linear factors that impact trading volumes, this research study focusses on building a micro framework for analyzing financial markets using trends, patterns and time series. Using time series is a key topic in financial domain as preciseness is essential (Zohar & Rosenschein, 2008; Lahmiri, 2016). This forecasting model supports traders to derive financial strategies and take directed decisions, thereby making transaction management in commodity trading a seamless one. Managing trade volumes is a key element for a trader in investment portfolio. Volume of trade is a direct indicator of the market. Using time series analysis methodology, there is an inherent advantage of analyzing data patterns and trends over time from multivariate parameters.

Use of this methodology for building the forecasting model with use of the NeuralProphet library is to predict future values volume forecast of our time series. NeuralProphet library toolkit has abstracted the inherent complexities of time series forecasting. Methods such as fit, predict and options to create future timeframes with seasonality brings ease of focus on the business implementation rather than Application programming Interfaces (API) derivatives. This utility had provided a platform for financial analysis and made it more intuitive for analysts and developers alike to work with time series data. Research authors Shah, Isah and Zulkernine have indicated numerous issues exist in stock market forecasting (Shah et al., 2019). Rackauckas has mentioned use of ARIMA and ARMA models are recommended for predicting straight revised standards. Use of LSTMs and RNNs are recommended for use where dynamics and market variants are evident and derived outcomes that are more precise than traditional methods (Rackauckas, 2018; Shewalkar et al., 2019).

3.3. Use of Neural Prophet

Using time series data sets, the goal of NeuralProphet (NP) is to extract information from the signal and use it for forecasting future. This API uses indicators of Growth / Trend, Season and Holiday from Signal to predict future trend. This model uses trend lines, which is an indicator of time as regressors in this model. In this context, trend lines are used as regressors for investigating and to predict the outcome. Use of regressors is a proven technique based on statistics and autoregressive dynamics for creating future predictions based on history.

The structure of time series model is shown in Figure 2, where the time series data may not be linear, NeuralProphet applies piecewise linear regression on to it by means of breaking different linear trends of the data into segments called "pieces" and apply regression function on the same. This feature also handles impact of a dampened trend as well. The next component which is "Seasonal" uses fourier variables to account for daily data patterns with weekly or yearly seasonal effects. This component brings in expansion based on flexibility using seasonal terms.

Fig. 2: Structure of Time series model

Signal = Forecasts extrapolate the signal portion of the model
Noise = Confidence intervals that account for uncertainty

Cleaned and preprocessed data is fed into NP, which is defined in terms of deep learning to enhance the prediction accuracy levels. Incorporating white noise detection techniques strengths this model to cut down on noise to signal ration. The novelty of this approach lies in enhancing high level efficiency to handle commodity trading. Through this learning model, observations of the smallest curve of improvement brings better visualization on market trends and nearness to real time price predictions. Nickel commodity dataset is considered as the base for data validation in this learning model. Three core components of NP are holidays, trend and seasonality. It is represented in a below equation (1):

$$NP = fn(TMF) + fn(SF) + fn(HF) + fn(EV) \quad (1)$$

Where $fn(TMF)$ represents trend-modeling function, $fn(SF)$ represents seasonality function, $fn(HF)$ represents Holiday function and $fn(EV)$ to handle error variation and outliers. Additional benefits are foreseen with use of Neural Prophet as the use of Generative Additive technique helps to dissect unstructured patterns and enhances performance of this prediction model. With aim to improve the prediction accuracy, this research approach uses integrated Neural Prophet with anomaly detection to build a learning model. It is observed that with enhanced weight adaptations in dropout regularization, outliers could be evaded.

With S as an indicator of seasonal period, t as time, and applying sine-cosine on variables of different frequencies multiplied with a numeric series of increasing order ranging from 1..n depending the domain / context in which it is applied. This model is designed by default for daily data and with use of the seasonal components it can accommodate weekly and yearly seasonal effects. This model provides expanded flexibility on seasonal terms. It is 10 times away from the yearly component of the data because the original design nature was to adopt for daily data, hence S is considered as 365.25 and weekly terms are accommodated in this model using three Fourier variables to help model them. The formulation between equations (2) to (9) is mentioned as follows:

$$Xf(x) = a_0 + \sum_{n=1}^{\infty} \left(a_n \cos\frac{n\pi x}{L} + b_n \sin\frac{n\pi x}{L}\right) + \cos\left(\frac{6\pi t}{365.25}\right) + \cdots + \sin\left(\frac{20\pi t}{365.25}\right) \quad (2)$$

$$X1, t = \sin\left(\frac{2\pi t}{S}\right) \quad (3)$$

$$X3, t = \sin\left(\frac{2\pi t}{S}\right) \quad (4)$$

$$X5, t = \sin\left(3\,X\ \frac{2\pi t}{S}\right) \quad (5)$$

$$X2, t = \cos\left(\frac{2\pi t}{S}\right) \quad (6)$$

$$X4, t = \cos\ (2x\frac{2\pi t}{S}) \quad (7)$$

$$X2, t = \cos\ (3x\frac{2\pi t}{S}) \quad (8)$$

$$Yt = \beta 0 + \beta 1X1t + \beta 2X1t + \cdots + et \quad (9)$$

Adaptability to the type of data in the data set can be performed with ease after careful understanding of the seasonal terms of data. This prediction model also introduces a point of intervention variable called the holiday component which provides an ability to use the binary indication for a certain day or period.

The purpose of this component is to indicate a period which may cause a spike or drop so that high-highs and low-lows are not affecting the data. This is also called pulse intervention. In line with the inherent behavior, this model does not use any lag values for prediction of the target variable. Rather, with curve fitting approach with extending curves to future considering time, season and pulse intervention periods are used for optimization and performance.

This model applies weighted scheme that decreases exponentially further back in time. This approach brings more emphasis on recent observations and less emphasis on previous observations with exponential smoothing windows using below equation (10):

$$Y\,e^x = 1 + \frac{x}{1!} + \frac{x^2}{2!} + \frac{x^3}{3!} + \cdots,\ -\infty < x < \infty = \cos\left(\frac{2\pi t}{365.25}\right) + \sin\left(\frac{4\pi t}{365.25}\right) + \cos\left(\frac{6\pi t}{365.25}\right) + \cdots + \sin\left(\frac{20\pi t}{365.25}\right) \quad (10)$$

In comparison to traditional neural network approaches with enhanced LSTM methods, it is observed there is a reduction in the error rate based on this experimental study. Authors Pham et al.

indicated that the application of dropouts in LSTMs prevents impact of overfitting and enhances performance and execution efficiency (Pham et al., 2014). Researcher Zhou mentioned in his study that LSTM has been used in price prediction modelling for stock market and pattern recognition model overtime and demonstrated extraordinary forecasts (Zhou et al., 2015; Gers et al., 2000). In same lines, this approach also aims to detect anomalies and evade outliers using dropout regularization to cleanse volumes of dataset and prepare a clean base for forecasting. NeuralProphet uses decomposable time series model for forecasting. Existing research investigations interprets non-linear factors that affect commodity market. Defined boundaries based on kernel abstains ability to explore data volumes that are dynamic and composite with structural data attributes (Strobl & Visweswaran, 2013).

3.4. Forecasting Model with Anomaly Detection

Outliers in time series are considered as weeds that can affect the prediction accuracy of the forecast model. Extreme high or low values in time series needs to be identified, examined for root cause which could be an exceptional demand leading market variation or a mistake or an error. In either case, it needs to be smoothened. In this approach, exponential window function and window slicing is used handle outliers. Any anomaly identified by means of seasonal patterns, is considered as an outlier. It is handled by means of seasonal decomposition using Daily and Weekly pattern identifiers.

In real time, trader can evaluate anomalous condition as outliers. If the value lies outside the range, it indicates a sign of warning which requires observation on parameters such as software or algorithm error analysis (if repeated situation arises then perform 5Y or 8D analysis). If it is not a software error and over time if the results have proved with required confidence levels, then it is a probability of a critical incident or change in consumer behavior.

3.5. Exponential Smoothing

Exponential window function is applied to handle the complexity of time series. Weights are assigned proportional to its geometric progression in the form of 1, (1- ∞), (1-∞) 2,…,(1-∞) n for observations over time. As time moves, variable smoothened statistic St is defined as the weighted average of the highest number of the past observations namely St-1, St-2, and do on to St-n. is substituted for exponential smoothening back into the time series.

3.6. Use of Seasonal trends

Fourier analysis studies the pattern of sin and cosine waves of different frequencies. Two or more waves of sin and cosine with varied frequencies can be combined to derive patterns. These patterns can be used to model seasonal waves in time series using NeuralProphet. Fourier analysis demonstrate series of Sine and Cosine terms of the right frequencies for approximation of periodic series. Inclusion of the Fourier variables to a regression model is ideal for predicting the target. This process considers removal of a seasonal pattern. The mathematical representation of daily seasonal pattern and the weekly seasonal pattern is given in equation (11) and equation (12) respectively.

Daily Seasonal Pattern

$$Xy = cos\left(\frac{2\pi t}{365.25}\right) + sin\left(\frac{4\pi t}{365.25}\right) + cos\left(\frac{6\pi t}{365.25}\right) + \cdots + sin\left(\frac{20\pi t}{365.25}\right) \quad (11)$$

Weekly Seasonal Pattern

$$\mathrm{Xw} = cos\left(\frac{2\pi t}{7}\right) + \sin\left(\frac{4\pi t}{7}\right) + \cos\left(\frac{6\pi t}{7}\right) \quad (12)$$

4. Algorithm

This model aims to forecast transaction volume for Nickel in commodity trading market using Neural Prophet integrated with anomaly detection techniques. The functional workflow of this model is depicted in Figure 3 as an algorithm using Input ⇔ Process ⇔ Output cycle.

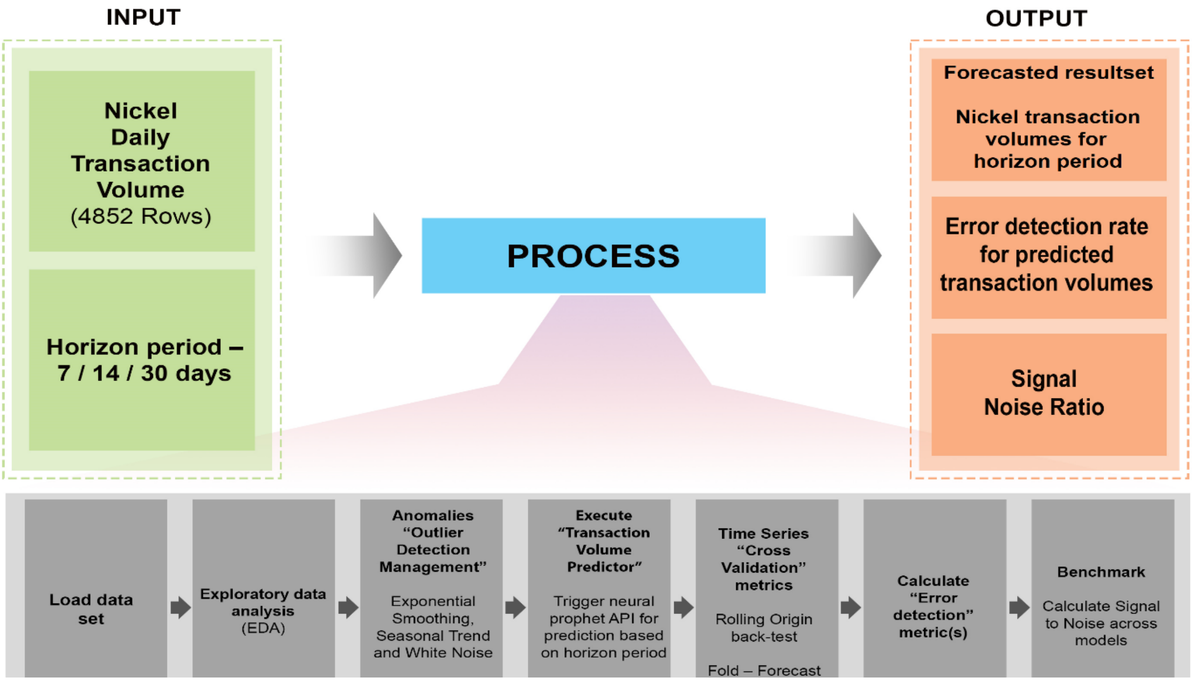


Fig. 3: Functional workflow of proposed model

Input: Data set with n sequence data of Nickel – Daily Transaction Volume and forecasting horizon period

Process:

Step 1: Load data set with daily transactional volume for period from 2004 to 2022

Step 2: Perform Exploratory Data Analysis (EDA)

Step 3: Detect Anomalies using "Outlier Detection Management"

- ➢ Set dropout regularization to ensure layers do not synchronously optimize weights, rather decorrelate the weights
  - o Activate hidden units
- ➢ Perform White Noise Detection
  - o Split transactional volume (History data) into four slices or chunks
  - o Design and create four data frames
  - o Calculate mean and standard deviation for each data frame
  - o Create graphical plot to see variations in mean and standard deviation

Step 4: Execute "Transaction Volume Predictor"

$\boldsymbol{for}(each\ neuron\ in\ Transaction\ volume\ layer)$

Configure activation function for weight optimization and hidden unit turns to be sparse

$$\boldsymbol{for}(index = 1; index \leq number\ of\ rows\ in\ dataset; index++)$$

Tune the model based on network and obtain neural network's defined outcome

Evaluate output (predicted transaction volume, p based on weight of neurons)

Apply exponential window function

To handle the complexity of time series, weights are assigned to observations over time in geometric progression [in the form of 1, (1- ∞), $(1\text{-}\infty)^2$,…, $(1\text{-}\infty)^n$]

Learn timeframe series, pattern, season and holiday

Adapt dropout for normalization rate

Apply auto-correlation, to find if previous data impacts the next day's data, proceed furthers

Observe signs of seasonality degree in each iteration

$\boldsymbol{end\ for}$

Invoke neural prophet API triggers to diagnose seasonality, period and mode

Create future prediction horizons for specific period – 7 / 14 / 30 days

$\boldsymbol{end\ for}$

Step 5: Calculate "Cross Validation" metrics

$\boldsymbol{for}(predindex = 1; \mathrm{pred}index \leq number\ of\ rows\ in\ predicted\ horizon; \mathrm{pred}index ++)$

Execute cross validation metric triggers procedure

Use following three parameters:

- o initial, variable that indicates days count for time series data for training purpose of cross validation
- o period, variable that indicates spacing between cut-off dates
- o horizon, variable that indicates forecast horizon period

$\boldsymbol{end\ for}$

Step 6: Calculate "Error detection" metric(s)

Generate prediction outcomes - yhat, yhat_upper, yhat_lower values which indicate predicted transactional volume, upper bound value indicating the maximum possible transactional volume and lower bound value indicating the minimum possible transactional volume of nickel trade on that date and horizon period

Step 7: Benchmark ⬄ Signal to Noise across models (Neural Prophet, FbProphet and Multi-Kernel LSTM)

- ▪ In prediction results, it is most observed that, actual value of y lies within the range of yhat_lower and yhat_upper
- ▪ There is greater advantage seen in using cleansed set after considering outliers and reduction of noise
- ▪ which in turn enhances Signal-to-Noise ratio (SNR)
- ▪ Use of exponential window function, results in advantage that indicate nearness of the predicted value
- ▪ yhat, in its closeness to actual value y

Output: 1. Predicted nickel transaction volumes for upcoming period (7 / 14 or 30 days)

2. Seasonal patterns

3. Error detection rate and Signal ⬄ Noise ratio (Across compared models)

5. Experimental Results

Figure 4 represents graphical depiction of forecasted transactional volume of Nickel in Commodity using Neural Prophet. Source data used for this experiment is gathered from Metal Commodity Exchange (MCX). This attribute when described in detail represents forecast of traded lot size in Nickel commodity trading market. This parameter and its associated attributes are considered as the basis for this prediction model. Let's consider a business context when a trader has entered into agreement in commodity trading market for Nickel, investment for the agreement is evaluated in terms of "Lot Size". Historical data set used for building this learning model is collated from https://in.investing.com for a series of 18+ years from Jun 2004 to Sep'2022 with 4852 rows of data. The key purpose of using data across wide time span is to enhance insights based on this learning model. It also helps to observe impact of behavioral patterns over time using each row in training data instance. The model is built and executed as per algorithmic steps detailed in Figure 1. Outcomes and evaluation measures with cross validation are explained in detail in Section V and VI.

5.1. A Rolling Basis – Cross Validation [Training – Testing Data Set]

The model is designed for execution on commodity data set viz., in this context it is specifically applied on metal – Nickel's training data set, which is prepared based on weights of neurons across layers using deep learning algorithms. In this model, data set is segmented into training (80%) and test data (20%) partition. To reaffirm behavior, iterative approaches in partitions ratio of "70-30" on the same dataset is evaluated to test outcome for trend conformance. High level of benefit is observed using this approach.

Rather than going forward with the approach of 80-20 and 70-30 in a direct approach in one iteration, it is implemented using a rolling basis over time periods in multiple iterations. For each time period split of "N" years of data, subset of data is taken as the basis for training (say 80% of the first 6 years) and forecast the next 20%. This validation with real time values to observe and compare with actual values which is shown in Figure 4. On a rolling basis, while performing for the next split period, already covered period will be used as part of training set and further the forecast is generated for validation. This iteratively marches to the phase where the forecast is made for the current time period. By the time, this model generates forecasted output for the current time period, a series of iterations would have been executed creating learning patterns for the model. This process has helped us enhance the performance and learning capability of the model.

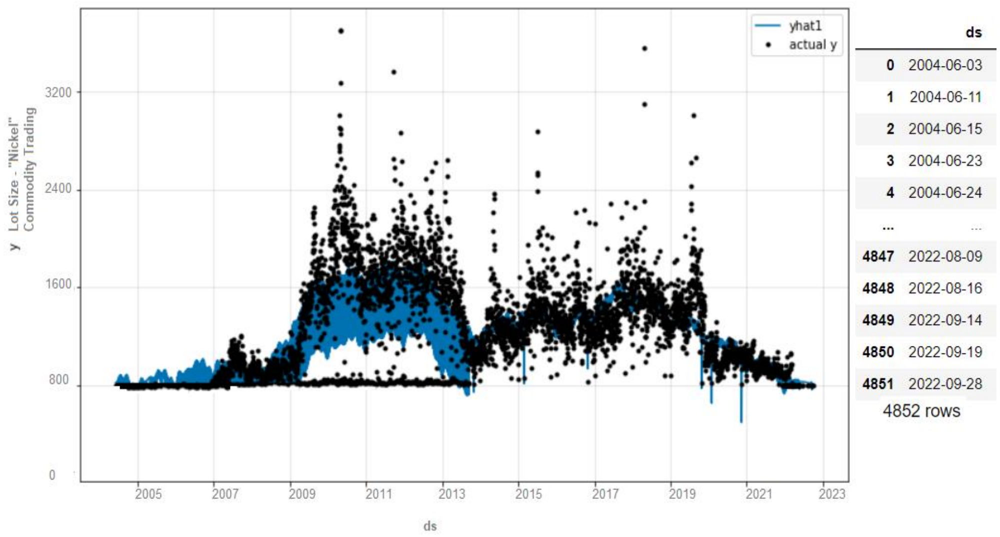


Fig. 4: Validation process of actual and real-time values

5.2. Prediction Results Based on "Horizon Window Period"

This model provisions ability to predict the range of values covering minimum, maximum and probable value of traded volume for horizon period using API triggers of Neural Prophet. During validation, it is observed that most of the time, actual value of y lies within the range of yhat_lower and yhat_upper. Forecast results provided in the form of a range segment also helps the trader take decision knowing the lower and upper limits. Where, the Nickel Traded Volume data between the range of 2004 – 2022 from Metal Commodity Exchange (MCX) is depicted in Figure 5.

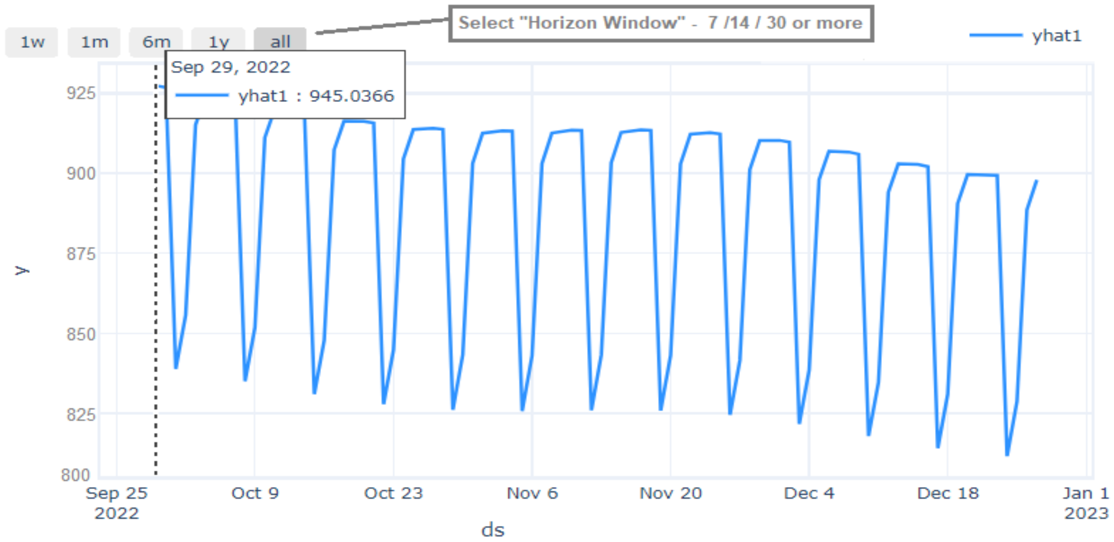


Fig. 5: Nickel Traded Volume 2004 – 2022 data range from Metal Commodity Exchange (MCX)

5.3. White Noise Detection

As part of steps to detect anomalies, this model uses identification of white noise in time series so as to ensure predictability. In the time series of daily transactional volume, it is expected to have a logical relationship as traders who are operating in this field, will plan for future investment based on the past and current trade volumes. From statistical point of view, white noise can be identified in a time series, if the calculated mean and standard deviation are as below:

- Mean of the series is zero
- The standard deviation calculated for the series remains constant and doesn't change over time
- There is no significant correlation between this time series in comparison to its lagged version

Data set volume runs across 18 years. The data is divided into 4 chunks and tested for presence of white noise which is shown in Figure 6. With the defined statistical measure, it is clear that there is no white noise present and this enable the data suitable for forecasting.

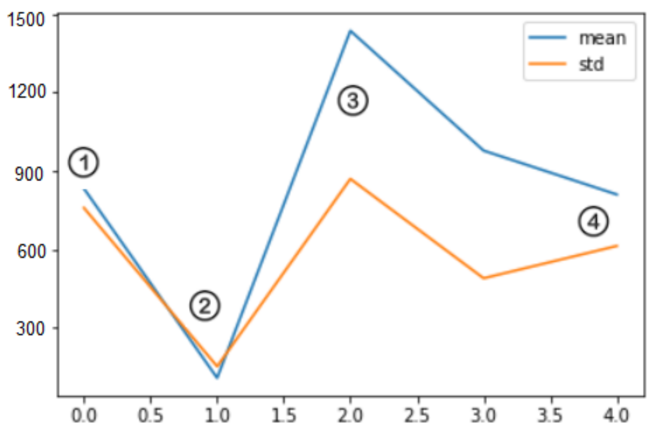


Fig. 6: White Noise Detection with Data split process

For detection of white noise, traded lot size of nickel (history data) is split into four slices or chunks. Four data frames are created and designed out of it for calculation of mean and standard deviation for white noise detection. This approach is executed in iterative cycles for each row of information in the data set. This approach covers "n" time sequences using x_index, y_index where iterator ranges from 1,…,n to evaluate experimental results. Outcome is created using the prediction model algorithmic function f based on row execution of n time sequences $(x_{ndex}, y_{index}), index = 1, \dots, n$.

5.4. Benchmarking - Error Detection Measures

Various measures on "epochs" are used to enhance forecast accuracy using machine learning algorithms. Future scope of this study indicates use of optimal sets in terms of data length and best fit epochs to enhance prediction accuracy **(Chhajer et al., 2022; Venkateswararao & Reddy, 2023)**. This learning model uses rolling basis for deriving at optimal data set length to derive patterns, includes weight adaptions to reduce impact of outliers and Neural Prophet along with anomaly detection to enhance forecast accuracy, thereby deriving high bench mark levels on error detection measures.

In table 2, the obtained results indicate outcome and derivatives of function f as described in $\frac{1}{n}\sum_{index=1}^{n} p\big(y_{index}, f(x_{index})\big)$ where p stands for the neural prophet triggers after performing exponential window smoothing and handling seasonality patterns.

Results are validated with cross validation for enhancing of performance metrics measures. Different error measures act as indicators for success of this approach. Error indicators are calculated based on actual and predicted values for traded volume over time sequences. Commonly used error rate indicators include Mean Square Error [MSE], measure of Normalized Mean Square Error [NMSE] and ratio of Signal to Noise [SNR]. On the other hand, there are few more statistical measures for validations such as prediction response, Mean, Variance and Skewness. As shown in Figure 7, the formulas are used for calculation of the degree of prediction whereas the proposed method detection performance is validated using various performance measures and its obtained results are given in table 2.

$MSE = \frac{1}{N_S}\sum_t (y_t - \mu_t)^2$ $\quad N_S$ Samples Size

$NMSE = \frac{\sum_t (y_t - \mu_t)^2}{\sum_t (y_t - \bar{y})^2}$ $\quad \bar{y}$ Represents mean of the real N_S samples of y_t

$NSR\,(dB) = 10\log_{10}\frac{\sum_t (y_t - \mu_t)^2}{\sum_t y_t^2}$ $\quad \mu_t$ Predicted sequences

$Skewness = \sum_t \frac{1}{N_S \sigma_t^2}(y_t - \mu_t)^3$ $\quad \sigma_t^2$ Forecasted conditional variations

Fig. 7: Description of formulae and its attribute

Table 2: Error Detection Measures Using Integrated Neural Prophet Model with Anomaly Detection

Outcome using Neural Prophet Model integrated with anomaly detection (dropout regularization and white noise)	**Seasonality**	**Errors (indicated in powers of ×10^{-3})**	
		R-MSE RootMean Square Error $RMSE = \sqrt{MSE}$	**Mean Absolute Error** $MAE = \underbrace{\frac{1}{n}\sum_{i=1}^{n}}_{\text{test set}} \lvert \underbrace{y_i}_{\text{predicted value}} - \underbrace{\hat{y}_i}_{\text{actual value}} \rvert$
07 days Horizon	Daily seasonality factor enabled using Prophet pf=NeuralProphet (daily_seasonality=true)	1.404	1.048
14 days Horizon	Weekly seasonality factor with Prophet pf=NeuralProphet (weekly_seasonality=true)	1.590	1.365
30 days Horizon		2.262	1.674

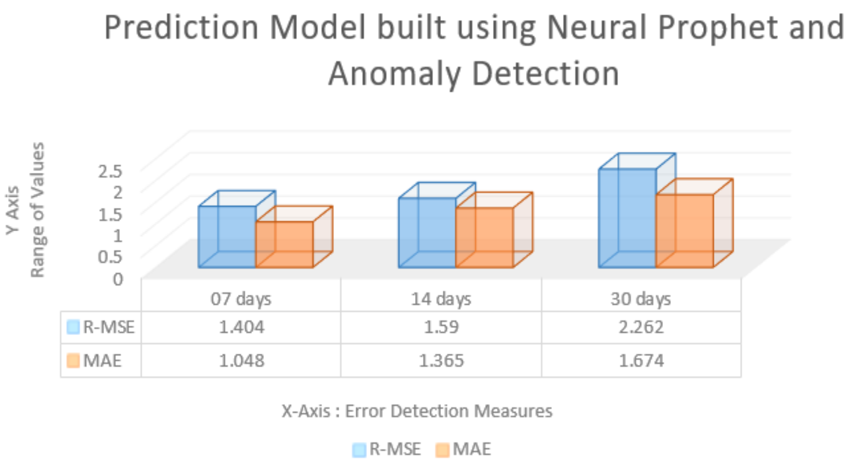


Fig. 8: Error Detection Measures using proposed approach across different horizon

Figure 8 depicts error detection measures using proposed approach over different horizon periods like 7 days with daily_seasonality attribute and 14, 30 days with weekly_seasonality attribute. Benefits of using exponential smoothing is applied in this algorithm and merits are evidently seen in the outcome as patterns are observed based on seasonality and time frame. It is also observed that anomalies and outliers are evaded using dropout regularization and white noise detection which has increased signal strength. In terms of error detection measures such as MSE, N-MSE, Signal to Noise Ratio, and Noise to Signal Ratio, the proposed method is compared with the existing method and the obtained results are given in table 3, to highlight its enhanced detection performance.

Table 3: Comparison of Error Detection Measures [Metal Commodity Used: Mcx – Nickel Data]

Statistical data from other predication Models	**Errors (indicated in powers of $\times 10^{-3}$)**			
	MSE	**N-MSE**	**Signal to Noise Ratio**	**Noise to Signal Ratio**
Multi-kernel LSTM (Enhanced with drop out regularization)	2.260	1.481	0.728	0.272
Prophet (fbprophet)	2.341	1.527	0.698	0.302
Integrated Neural Prophet with Anomaly and Noise detection	1.974	1.084	0.8882	0.1118

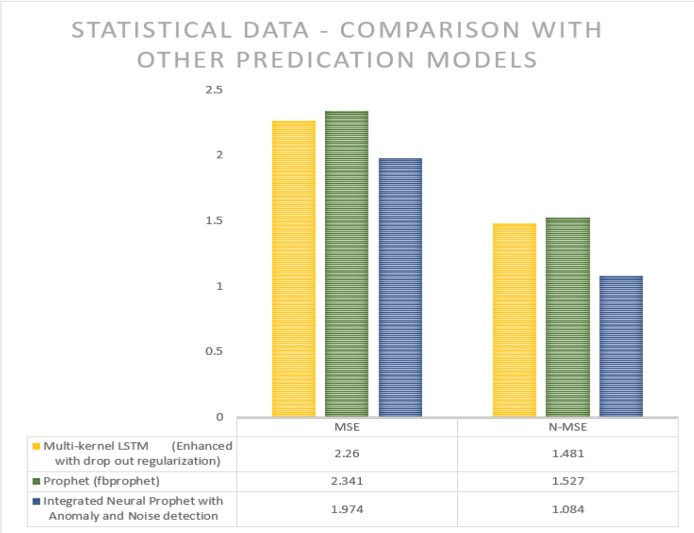
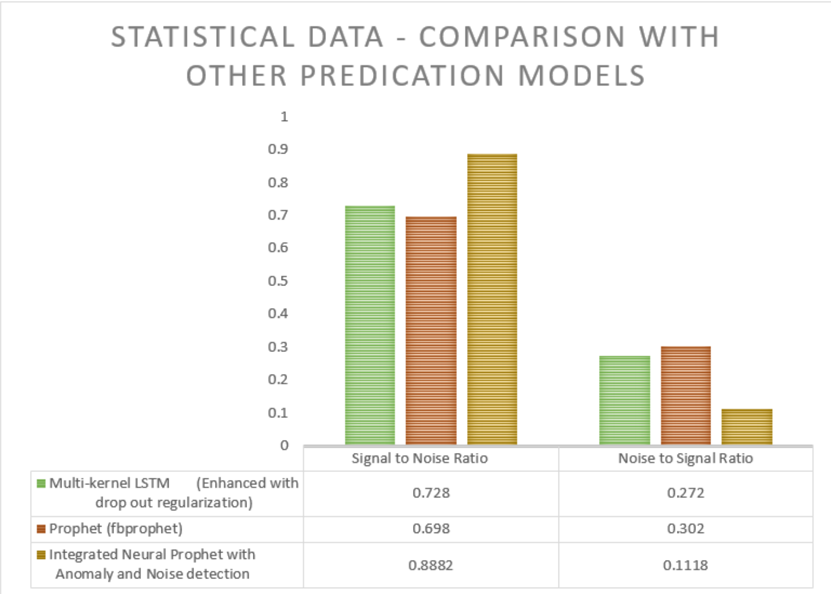


Fig. 9: Graphical representation of Error Detection Measures



Test metric	DataLoader 0
Loss_test RegLoss_test	0.11168357729911804 0.0

	Loss_test	RegLoss_test
0	0.111684	0.0

Fig. 10: Error Detection Measures rates for Nickel Price Prediction

Figure 9 depicts a bar graph indicating error detection measures using various approaches in comparison to another based on transactional data for last three cycles of 3 months period and consolidated the performance of the mentioned models. It is observed that the Mean Square Error (MSE) using Enhanced Multikernel LSTM (MKLSTM) indicates 2.26 and with use of basic prophet (fbprophet library) – MSE is calculated as 2.341. Whereas with the proposed model, MSE indicates as 1.974. From the results calculated above, we can conclude that on average, forecasts made with proposed integrated approach are closer to the real data and error detection are lesser. This integrated approach bundles the benefits of the appropriate core systems. For best results overtime, varying parameters are to be considered in tune with market dynamics.

Figure 10 depicts graphical representation of Error Detection Measures rates for Nickel Price Prediction. This approach focuses on bundling the core of Neural Prophet with anomaly evasion using drop outs and white noise detection. There by, removal of outliers and handling white noise provides a significant impact in strengthening signal to noise ratio (SNR). As the SNR is strengthened, error rates are reduced and bring the accuracy of the prediction model, thereby making predictions much nearer to the observed data than traditional approaches.

6. Conclusion

Literature studies and research on predication models were commonly used for calculating error indicators such as of Mean Square Error using traditional and straight forward neural network models. Model prediction with use of LSTM increased with use of Gaussian models and various improvements were seen. Use of alternative approaches like deep multi kernel learning and other research work are analyzed for study on price volatility. Novelty and strength of this approach lies in using an integrated approach of using neural prophet API triggers (estimated with Fourier series) along with anomaly detection measures. This helps in evading outliers and cleansing the data. This paper recommends use of integrated Neural Prophet along with anomaly detection as a learning model to observe various market trends of commodity trading. Data across wide time span of Nickel data set is collated. This is used as base entity to build, use, observe and predict using this model. With data indicators and statistical measure consolidated in above section (IV), there are a lot of benefits and performance improvement gains in error detection. Other key indicators such as Signal-to-Noise ratio is also strengthened using this approach.

To conclude - use of configuring weights, dropout regularization integrated with neural prophet and anomaly detection brings high level of performance benefits over linear or traditional approaches. This study provides results that can be used as early indicators for investor to strategize decision making and enhance security of the model using regressors that can abstract outliers. It provides an unbiased view towards decision making and devising transaction strategies based this forecasting model.

Future study using this integrated model will focus on using regressors for specific time periods for observing and comparing behaviours. Stock pricing is hard to predict based on its dynamic traits and changing market conditions. A hybrid approach is combined using LSTM and butterfly optimization for accuracy. To withstand such situations and challenges elaborated, use of added regressors for certain time period like seasonal or holiday package to extract time carved result. This is a handy feature to evaluate impact market behaviour based on subsidy or if a specific marketing campaign needs to be considered over different time frames as a variant that determines parameters as an influence on outliers. In the context of the study using metal commodities, this feature with regressors will help traders understand trends specific to metal commodity exchange.

References

Chhajer, P., Shah, M., & Kshirsagar, A. (2022). The applications of artificial neural networks, support vector machines, and long–short term memory for stock market prediction. *Decision Analytics Journal,* 2, 100015.

Gers, F. A., Schmidhuber, J., & Cummins, F. (2000). Learning to Forget: Continual Prediction with LSTM, Neural Computation. *MIT Press Journal*, 12(10), 2451-2471.

Ghosh, I., Sanyal, M. K., & Jana, R. K. (2020, February). An ensemble of ensembles framework for predictive analytics of commodity market. In *2020 4th International Conference on Computational Intelligence and Networks (CINE)*, Kolkata, India (pp. 1-6). IEEE.

Hopman, C. (2007). Do supply and demand drive stock prices? *Quantitative Finance*, 7(1), 37-53.

Lahmiri, S. (2016). Intraday stock price forecasting based on variational mode decomposition. *Journal of Computational Science*, 12, 23-27.

Merkel, G. D., Povinelli, R. J., & Brown, R. H. (2017, June). Deep neural network regression for short-term load forecasting of natural gas. In *37th Annual International Symposium on Forecasting*, Cairns, Australia (pp. 246-255). IIF.

Othman, A. H. A., Kassim, S., Rosman, R. B., & Redzuan, N. H. B. (2020). Prediction accuracy improvement for Bitcoin market prices based on symmetric volatility information using artificial neural network approach. *Journal of Revenue and Pricing Management*, 19(5), 314-330.

Pham, V., Bluche, T., Kermorvant, C., & Louradour, J. (2014, September). Dropout Improves Recurrent Neural Networks for Handwriting Recognition. In *2014 14th International Conference on Frontiers in Handwriting Recognition*, Hersonissos, Greece (pp. 285-290). IEEE.

Rackauckas, C. (2018). Re: What is the advantage of using RNN and LSTM over traditional methods for time series of streaming data? *Retrieved from*: https://www.researchgate.net/post/What-is-the-advantage-of-using-RNN-and-LSTM-over-traditional-methods-for-time-series-of-streaming-data/5acff394615e279654383f81/citation/download.

Shah, D., Isah, H., & Zulkernine, F. (2019). Stock Market Analysis: A Review and Taxonomy of Prediction Techniques. *International Journal of Financial Studies*, 7(2), 26.

Shewalkar, A., Nyavanandi, D., & Ludwig, S. A. (2019). Performance Evaluation of Deep neural networks Applied to Speech Recognition: RNN, LSTM and GRU. *Journal of Artificial Intelligence and Soft Computing Research*, 9(4), 235-245.

Strobl, E. V., & Visweswaran, S. (2013). Deep Multiple Kernel Learning. In *2013 12th International Conference on Machine Learning and Applications*, Miami, FL, USA (pp. 414-417). IEEE.

Taylor, S. J., & Letham, B. (2018). Forecasting at scale. The American Statistician, 72(1), 37-45.

Venkateswararao, K., & Reddy, B. V. R. (2023). LT-SMF: long term stock market price trend prediction using optimal hybrid machine learning technique. *Artificial Intelligence Review*, 56(6), 5365-5402.

Weng, F., Chen, Y., Wang, Z., Hou, M., Luo, J. & Tian, Z. (2020). Gold price forecasting research based on an improved online extreme learning machine algorithm. *Journal of Ambient Intelligence and Humanized Computing*, 11(10), 4101-4111.

Zhou, C., Sun, C., Liu, Z., & Lau, F.C.M. (2015). A C-LSTM Neural Network for Text Classification. *arXiv*:1511.08630.

Zohar, A., & Rosenschein, J. S. (2008). *Mechanisms for information elicitation. Artificial Intelligence*, 172(16-17), 1917-1939.