Advances in the Theory of Nonlinear Analysis and its Applications 7 (2023) No. 4, 89-98.
https://doi.org/10.17762/atnaa.v7.i4.285
Available online at www.https://atnaea.org/
Research Article

Advances in the Theory of Nonlinear Analysis and its Applications
ISSN: 2587-2648
Peer-Reviewed Scientific Journal

Structural structure of the STDR-MNN Hybrid Model with Application

Nihad S. Khalaf Aljboori[a]

[a] *Department of Mathematics, College of Education for Women, Tikrit University, Tikrit, Iraq.*

Abstract

Time series are always preoccupied with maintaining their stability to make precise forecasts about the future. So, a plethora of statistical models are emerged, by some focusing on mean stability and others on variance stability. Neural networks formation that resemble the structure of the nervous system in humans subsequently followed this. Now, in order to break down the time series into its constituent parts, our hybrid model will integrate the time series' structure. the neural network performs the sub-network prediction procedure after receiving the divided sub-series. These predictions are then concatenated to provide the original network prediction. Time series data display a wide range of patterns. It is frequently helpful to dissect a time series into its trend, seasonality, cyclic variation, and irregular components in order to identify underlying patterns. The time series model is in charge of classifying the data series into four patterns such as: seasonality, trend, dispersion, and the remainder which are expressed in the model by S, T, D, and R respectively. By this work, we will introduce a hybrid model called STDR-MNN that combines a time series with a neural network. The four segmented series are sent to the MNN neural network, which uses them to forecast each segmented series individually. Finally, we will use the MATLAB2022A program to test a realistic application on the generated hybrid model in order to gauge its effectiveness with Real data.

Keywords: Modular neural network; Remainder; Seasonality; Trend; Dispersion; STRD-decomposition; STRD-MNN hybrid model.
2010 MSC: 00-XX, 00-02

1. Introduction

Common techniques for deriving components from a time series are frequently employed to enhance comprehension of time series, but they may also be applied to raise prediction accuracy in the future. When

Email address: nihad.shreef16@tu.edu.iq (Nihad S. Khalaf Aljboori)

Received September 22, 2023; Accepted: November 26, 2023; Online: December 19, 2023.

a time series is broken down into its constituent parts, we are left with three parts: the trend cycle component, the seasonal component, and the residual component, which is made up of the other parts of the time series.

Regarding the trend component, where a trend is any long-term rise or decrease in the data; a trend need not follow a linear pattern. When a trend changes from a rising to a declining trend and then back to the seasonal component, it is sometimes referred to as a "change." When a time series is impacted by seasonal variables like the day of the week or the season, this is known as a seasonal pattern. Where Seasonality is always fixed and has a set periodicity.

Inspired by meteorology and astronomy, nineteenth-century economics proposed breaking down time series into unobservable components [1]. During that period, a great deal of study was conducted to identify the "cycles" that allowed economic crises to be predicted and explained. Poynting suggested using price averaging in 1884 as a means of removing seasonal and trend changes [2]. Later, other researchers developed this strategy, such as Copeland, who made the first attempt to extract the seasonal component [3], Persons, who identified the trend, cycle, seasonal, and irregular components of a time series, and suggested an algorithm for estimating them (the kin-joining method) [4], the enumeration technique II was created based on the Macauley method, and several of its versions, including X-11, X-11-ARIMA, X-12, X-12-ARIMA, X13, X-13ARIMA-SEATS, and TRAMO-SEATS, are still in widespread use today. An extensive analysis of these techniques [5], Next, Grzegorz introduced the Seasonal-Trend-Dispersion Decomposition model in 2015, creating a hybrid of additive and double decomposition by including dispersion in the double decomposition [6].

One kind of neural network that may be separated into several independent modules or subsystems operating on the input space is a modular neural network. These modules each focus on a distinct region of the input space, and by combining their unique answers, they are able to handle challenging computational problems. Additionally, modules that have acquired a function can be applied in other parts of the network [7], and each function is divided into smaller procedures that may be carried out independently in different modules without interfering with one another [8], followed by several later research, including the 2021 study by Wei et al. In order to create a transparent neural network architecture, the MNN-CH is a novel modular neural network that is built utilizing a hierarchy of investigated classes and certain patterns enforced in the associated modules [9], a modular input structured neural network architecture for large-scale neural network realization was presented by Elamparithy et al. in 2022 [10], Younès and Charles 2023 proposed a semi-supervised approach based on modular learning to classify dorsal septal instability [11], The neural network (NN) technique was introduced by Hongjun and others in the same year as an alternative to the computationally demanding Model Predictive Control (MPC) in modular transformer control. Dimensional [12].

2. STRD-Decomposition

The decomposition equation for time series is given in the form [5]:

$$\begin{aligned} x_t &= T_t + S_t + R_t && (1) \\ x_t &= S_t \times T_t \times R_t && (2) \end{aligned}$$

Where x_t is data series, T_t trend, S_t seasonal, and R_t remainder Assume that $\{x_t\}_{t=1}^{N}$ is a seasonal time series for a time period with n observations. Assuming that $\frac{N}{n} = K$ that means the series' length are longer than the seasonal time by twice as much. As a succession of progressively occurring seasonal sequences, the time series can be expressed as:

$$\left\{ \{x_{i,j}\}_{j=1}^{n} \right\}_{i=1}^{k} = \left\{ \{x_{1,j}\}_{j=1}^{n}, \{x_{2,j}\}_{j=1}^{n}, \ldots, \{x_{k,j}\}_{j=1}^{n} \right\} \tag{3}$$

So that j, i, and $t = j + (i-1)\,n$ are the index of time within the identified seasonal pattern, the running number of the seasonal cycle, and the global time index, respectively.

Therefore, the decomposition equation for seasonality, trend, and dispersion which is known by the abbreviation STD was proposed by Dudek where D_t represents the value of the dispersion component as follows [6]:

$$x_t = S_t \times D_t + T_t \tag{4}$$

In the same study, he also proposed a development of this equation by adding the remainders, which he defined by the abbreviation STDR, which are given by the equation:

$$x_t = \hat{S}_t \times D_t + T_t + R_t \tag{5}$$

Where $\hat{S}_t$ represent the average of S_t.

The values of the four components of the decomposition can be calculated as follows: The average amount of the seasonal sequence i-th is found according to Eq.(6), while the specific measure of diversity is defined according to Eq.(7):

$$\bar{x}_i = \frac{1}{n}\sum_{j=1}^{n} x_{i,j} \tag{6}$$

$$\hat{x}_i = \sqrt{\sum_{j=1}^{n} (x_{i,j} - \bar{x}_t)^2} \tag{7}$$

And seasonal component which is known as using an average seasonal pattern $\bar{S}_t$ as follows:

$$\bar{S}_j = \frac{1}{k}\sum_{j=1}^{k} S_{i,j} \tag{8}$$

Thus, we obtain the four components of the decomposed series with the following equations:

$$T_t = \underbrace{\{\{\bar{x}_i, \ldots, \bar{x}_i\}\}_{i=1}^{k}}_{\text{n-times}} \tag{9}$$

$$D_t = \underbrace{\{\{\hat{x}_i, \ldots, \hat{x}_i\}\}_{i=1}^{k}}_{\text{n-times}} \tag{10}$$

$$\hat{S}_t = \underbrace{\{\{\bar{S}_j, \ldots, \bar{S}_j\}\}_{j=1}^{n}}_{\text{n-times}} \tag{11}$$

$$R_t = x_t - \hat{S}_t \times D_t - T_t \tag{12}$$

3. Modular neural networks [13, 14]

A Modular neural network is usually represented by (MNN) is an artificial neural network that is made up of many separate neural networks that are overseen by different middlemen. As discrete building pieces, each autonomous neural network functions as a module, completing a specific subtask of the overall task the network is intended to complete with the help of distinct inputs. As a result, modules that are already proficient in one task may be applied to other tasks inside the network. Sub-operations within an individual function can be carried out independently in different modules without interfering with one another. It has been noted that the majority of well-known artificial neural networks lack any forced conventional structure. There is an issue with convergence with these networks. Mathematically, a modular neural network is a collection of linked units denoted by a set of seven-tuples.

$$N = \{I, C, r, P, O, d\} \tag{13}$$

Where I represents the number of inputs, C represents the number of classes divided by the number of units in the input unit, while r is the representation of the median, P is the permutation function, O represents the output layers of the model, and finally d represents the decision unit of the model. The MNN is divided into three subsections. This would form the basis for MNN models:

1. Problem division: The first main task is to divide the problem into parts. We mentioned that problem complexity is a major problem in using ANN and thus modules are required to solve this complexity problem. The first method commonly used in these systems is to delegate different modules to different types of inputs. Using this approach each ANN or module has a reasonably small complexity that it deals with. The other mechanism is to have all the different modules solve the same problem. In this mechanism there is of course no division of the problem.
2. Modules: Each of the modules in this approach is an independent ANN. This ANN has its own training procedures and training is performed independently of all other procedures and modules.
3. Integration: The last function that an integrator must perform is integration. In this mechanism the responses of different units are considered. It is on the basis of the outputs that the different modules create the decision regarding the final system output.

The training algorithm for MNN consists of two stages:

- The first stage: training the input layers, which includes choosing the training set from the original training set for each $i = 1, 2, \ldots, C-1$, and training all sub-models on the test set using the BPA backpropagation algorithm.

- The second stage: training the decision network, which includes calculating the r value for each of the first layers of the input vector k, building the training set from the decision network, and finally training the decision network on the decision network training set using BPA. Where the value of r and the value of the decision network training set TS_d can be calculated by Eqs.(14) and (15), respectively.

$$r^k = \varphi\left(y_1^k, y_2^k, \ldots, y_I^k\right) \tag{14}$$

$$TS_d = \left\{ {}^{(r^k; d^k)}/_{k = 1, 2, \ldots, t} \right\} \tag{15}$$

4. STRD-MNN hybrid model

STRD-MNN is a hybrid approach that enhances prediction performance by fusing the MNN neural network with the STRD model. Below are the steps for making this algorithm in detail:

1. Entering a data series.
2. Utilizing STRD, divide the original series into four substrings after setting the analysis period, they are seasonality, trend, remainders, and dispersion.
3. Each of the four series' subseries' data is split into a training set and a test set, and the prediction step is established before to the start of training. The MNN neural network is trained with the same parameters on the four subseries that were collected in step 2.
4. Compute the absolute error (MAE) and root mean square error (RMSE) by adding the subseries prediction results to the original data prediction results using Eq.(5).

$$MAE = \frac{1}{m}\sum_{i=1}^{m} |e_t - \widetilde{e}_t| \tag{16}$$

$$RMSE = \sqrt{\frac{1}{m}\sum_{i=1}^{m} (e_t - \widetilde{e}_t)^2} \tag{17}$$

Where $\widetilde{e}_t$ represents the predicted value, e_t is the real value at that moment t, and m the quantity of data included in the test set.

5. Modify the settings to determine the best MNN time step and decomposition frequency so that the indexes can.
6. It is possible to forecast the original series using the model that has the best indices on the test set.

The suggested hybrid model's algorithm diagram is shown below.

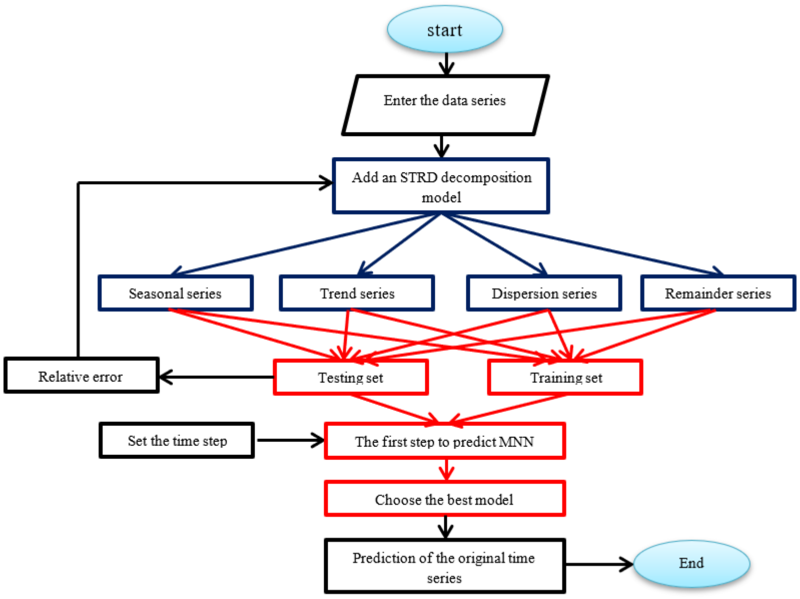

Figure 1: Diagram of the proposed hybrid model

5. Data analysis

5.1. Description of data

The data used in this analysis represented the monthly number (thousands) of foreign visitors in the United Kingdom for the period from January 1986 to February 2020, with 410 views obtained from the website https://www.kaggle.com/datasets, divided into two groups. The first group included the first 398 views, which were adopted in constructing the data. And time series analysis, while the second group included the last 12 observations of the data series, which were left to examine the predictive performance of the proposed hybrid model.

5.2. Analysis

First, we will draw the time series before performing any step in the analysis, as Figure.2 represents a drawing of the logarithm data series.

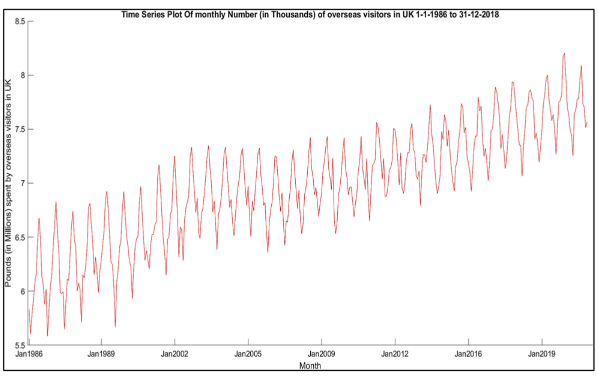

Figure 2: Drawing of the data series

Then we plot the functions ACF and PACF for 20 lags.

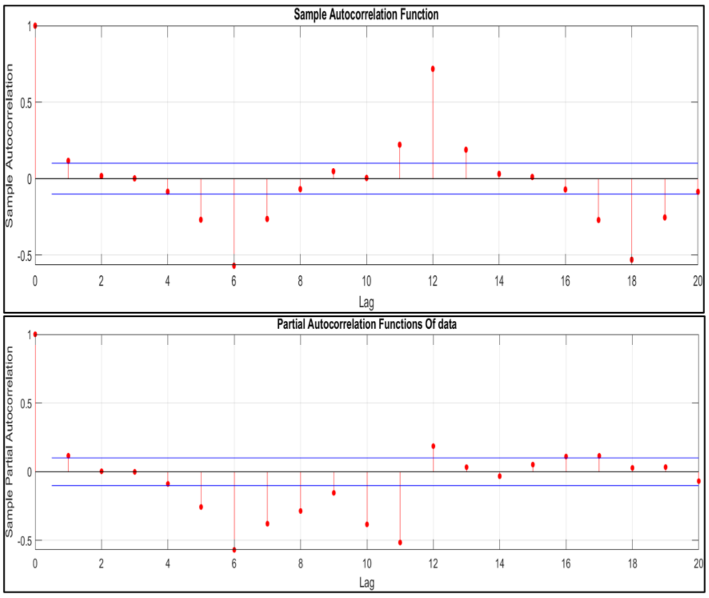

Figure 3: ACF and PACF for 20 lags

Second, We divide the time series into its four parts, which are)Seasonality, Trend, Dispersion, and the Remainder). Figure 4 represents the parts of the time series

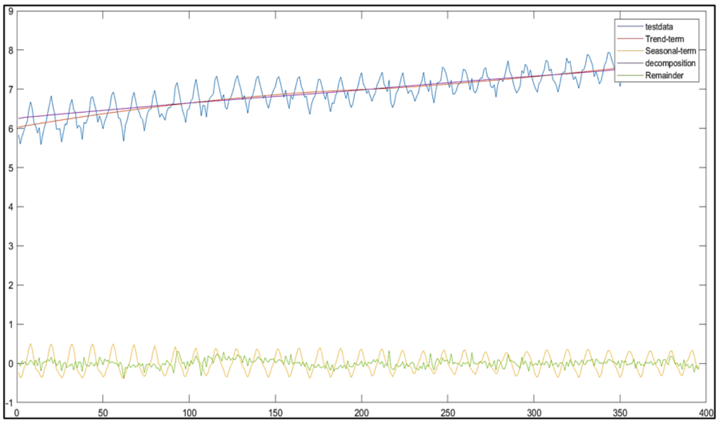

Figure 4: The Parts of the time series

It is noted from Figure 4 that the trend and dispersion components follow two almost straight lines aligned with the data series, while the seasonality components and the rest follow two zigzag lines far from the data series.

Third, each of the four components is divided into two groups, a test group and an evaluation group, to prepare the work of the Modular neural network (MNN), so that the MNN runs multiple times to analyze each component With 100 hidden layers. Where Table 1 represents the values of the training set

Table 1: Training set values.

Unit	Initial value	Stopped value	Target value
Epoch	0	9	1000
Elapsed Time	-	00:00:02	-
Performance	9.59	3.09e-07	0
Gradient	26.2	7.31e-05	1e-07
Mu	0.001	1e-10	1e+10
Validation Checks	0	6	6

Table 2: The training algorithm.

Data Division	Random
Training	Levenberg-Marquardt
Performance	Mean Squared Error
calculations	MEX

Where Figure 5 represents the drawing of the test set and the training and validation set, Figure.6 the drawing of the regression values, the Mu values, and the validation test values, and Figure.7 the regression drawing of the test set and the training and validation set.

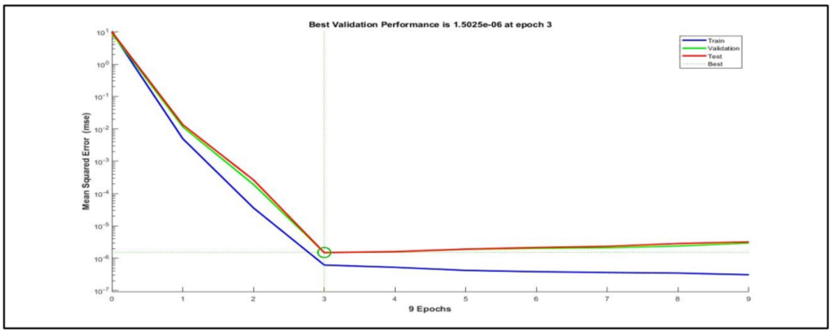

Figure 5: Represents the drawing of the test set and the training and validation set

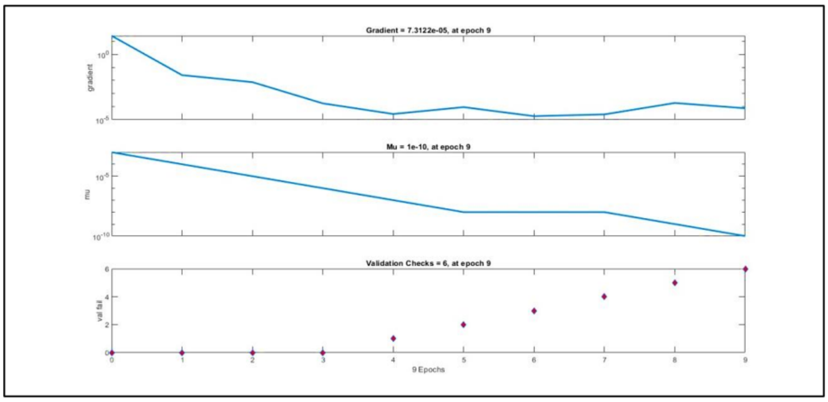

Figure 6: The regression values, the Mu values, and the validation test values

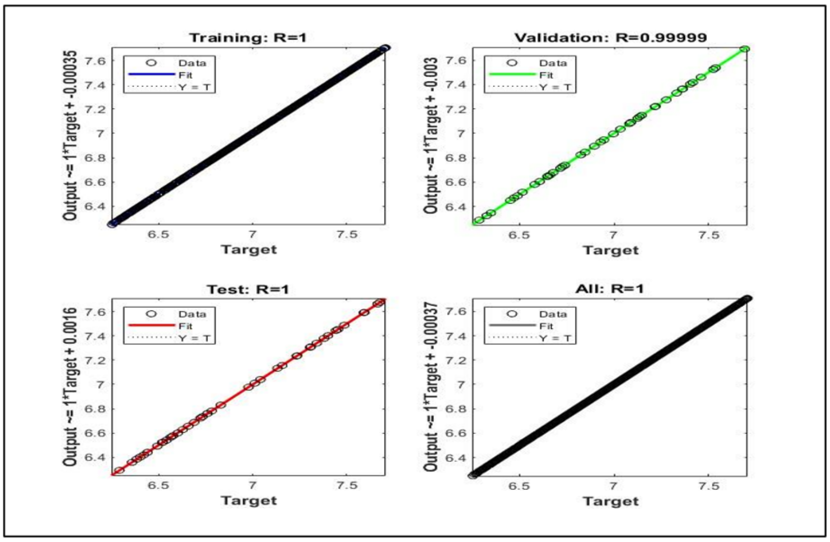

Figure 7: The regression drawing of the test set and the training and validation set

Fourth, after the neural network of the four components has completed its work in obtaining output values, the output series is drawn for the original series and the original series in addition to the sub-series. Figure.8 represents the output series for the original series and the original series in addition to the sub-sequences.

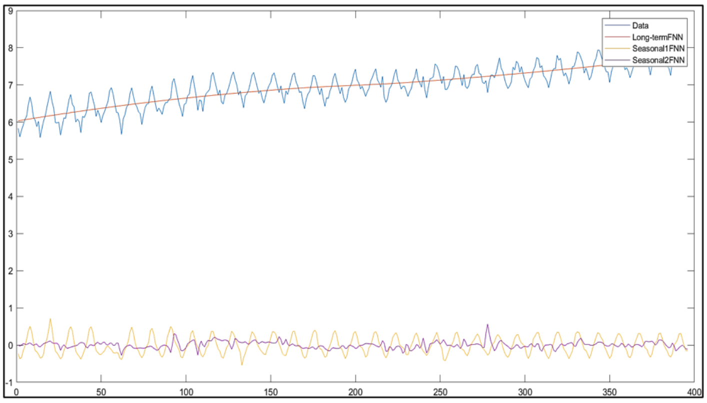

Figure 8: The output series for the original series and the original series in addition to the sub-sequences

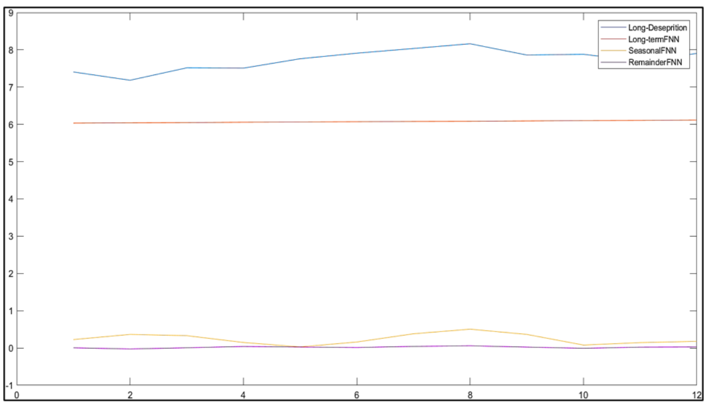

Figure 9: The outputs of the four subnetworks

Fifthly, the decomposition Eq.(5) is used to collect the outputs of the neural network to obtain a prediction of the values of the original time series. Figure 10 represents the predictions of the STRD-MNN hybrid model for the last 12 observations of the data series that were set as an evaluation set for the model prediction.

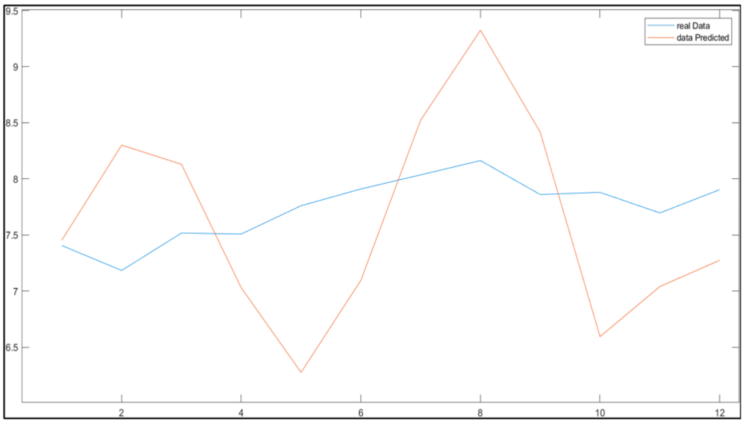

Figure 10: The predictions of the STRD-MNN hybrid model for the last 12 observations of the data series

Table 3 represents the values of the last 12 observations in tens of thousands of the time series values, in addition to the hybrid model prediction values, as well as the MAE and RMSE values.

Table 3: The last 12 observations of the time series values, Predictions of STRD-MNN hybrid model, as well as the MAE and RMSE values.

Real Data	Predictions of STRD-MNN hybrid model	Predictions of STRD
1646	1636	1786
1318	1352	4027
1840	1863	3396
1824	1807	1129
2347	2297	531
2725	2706	1206
3090	3107	5034
3510	3549	11219
2593	2611	4515
2645	2617	730
2201	2186	1141
2708	2694	1443
MAE	23.666	2045.9166
RMSE	26.2726	2688.0879

6. Conclusions

The proposed STRD-MNN hybrid model proved highly efficient in terms of prediction accuracy, especially in this type of time series that have a large sample size and high fluctuations as a result of their significant interaction in terms of seasonality. The error rate for MAE and RMSE was very small compared with the original STL model before merging, the amount of error was very high when compared to the hybrid model, which confirms that the connection A time series analyzed into parts with a neural network gives results with high accuracy and efficiency.

References

[1] C. H. D. Buys-Ballot, The periodic changes in temperature, depending on the nature of the sun and the moon, related to the prognosis of the weather, éducts from Dutch observations from 1729 à 1846, Kemink, Utrecht, 1847.

[2] S. Makridakis, S. C. Wheelwright, R. J. Hyndman, Forecasting methods and applications, John wiley & sons, Hoboken, New Jersey, 2008.

[3] A. Dokumentov, R. J. Hyndman, STR: Seasonal-Trend Decomposition Using Regression, INFORMS Joural on Data Science 1 (1) (2022) 50–62.

[4] W. M. Persons, An index of general business conditions, Review of Economic Statistics 1 (1) (1919) 111–205.

[5] E. B. Dagum, S. Bianconcini, Seasonal adjustment methods and real time trend-cycle estimation, Springer, Berlin, Heidelberg, 2016.

[6] G. Dudek, Std: A seasonal-trend-dispersion decomposition of time series, IEEE Transactions on Knowledge and Data Engineering 35 (10) (2023) 10339–10350.

[7] T. Hrycej, Modular learning in neural networks: a modularized approach to neural network classification, John Wiley & Sons, Inc., Hoboken, New Jersey, 1992.

[8] B. L. Happel, J. M. Murre, Design and evolution of modular neural network architectures, Neural networks 7 (6-7) (1994) 985–1004.

[9] W. Han, C. Zheng, R. Zhang, J. Guo, Q. Yang, J. Shao, Modular neural network via exploring category hierarchy, Information Sciences 569 (2021) 496–507.

[10] M. Elamparithy, R. Chithra, A. James, Imagesplit: Modular memristive neural network, in: The 22nd International Conference on Nanotechnology (NANO), IEEE, Palma de Mallorca, Spain, 2022, pp. 535–538.

[11] N. B. Salem, Y. Bennani, J. Karkazan, A. Barbara, C. Dacheux, T. Gregory, Modular neural network approaches for surgical image recognition, arXiv preprint arXiv:2307.08880 (2023).

[12] H. Wang, Y. Yue, B. Sun, H. Zhao, Neural network-based model predictive control approach for modular multilevel converters, in: 2023 4th International Conference on Electronic Communication and Artificial Intelligence (ICECAI), IEEE, Guangzhou, China, 2023, pp. 324–330.

[13] M. Kalaiyarasi, R. Janaki, A. Sampath, D. Ganage, Y. D. Chincholkar, S. Budaraju, Non-additive noise reduction in medical images using bilateral filtering and modular neural networks, Soft Computing (2023) 1–10.

[14] N. Winter-Hjelm, Å. B. Tomren, P. Sikorski, A. Sandvig, I. Sandvig, Structure-function dynamics of engineered, modular neuronal networks with controllable afferent-efferent connectivity, Journal of Neural Engineering 20 (4) (2023) 046024.