IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

IEEE SYSTEMS, MAN AND CYBERNETICS SOCIETY SECTION

Received March 1, 2022, accepted March 12, 2022, date of publication March 16, 2022, date of current version March 22, 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3159797

Meta-Transfer Learning Using Wavelet Decomposition for Multi-Horizon Time Series Forecasting

MARIO MAYA AND WEN YU, (Senior Member, IEEE)
Departamento de Control Automatico, CINVESTAV-IPN (National Polytechnic Institute), Mexico City 07360, Mexico
Corresponding author: Wen Yu (yuw@ctrl.cinvestav.mx)

This work was supported in part by Project CONACyT-A1-S-8216, and in part by Project SEP-CINVESTAV-62.

ABSTRACT Multi-horizon time series forecasting is a very challenging task in many fields of research. In the field of machine learning, artificial neural networks have been used to carry out these tasks. However, there are still problems that are of general interest to researchers such as: Loss of data in data acquisition and long-term forecast. In this paper, we propose a hybrid Meta-Transfer Learning technique based on transfer-learning, meta-learning and signal detection by means of the discrete wavelet transform to solve the aforementioned problems in multi-horizon time series forecasting. Input-to-state stability analysis and the strong and weak convergence analysis for the proposed method are included. To validate the effectiveness of the method, the long-term prediction of earthquakes magnitude (M>4.5) in Italy is taken as a case of study, using information from Italy and Mexico. Simulations of classic methods for forecasting time series based on neural models are performed. The forecasting performance obtained is the minimum square error (MSE) is 0.091, while for the meta-transfer learning, the MSE is 0.032.

INDEX TERMS Deep learning, meta-transfer learning, wavelet decomposition, stable learning, time series forecasting.

I. INTRODUCTION

Time series forecasting is one of the most important tasks in the field of information engineering. Two main types of forecasting can be distinguished [1]: short-term prediction and long-term prediction, also called multi-step or multi-horizon prediction. Many popular methods are being used to solve this problem, such as Box and Jenkins' approach [2]. In [3], a review of the most common methods to resolve this issue are presented. There are linear and non-linear regression models that allow the modeling of a time series, for instance, the linear methods as ARX, ARMA, NARMA models [4], and neural networks for nonlinear modeling [5].

Unlike short-term time-series forecasts, long-term forecasts often present a challenge to research efforts as they have well-known problems, such as increased forecast error when forecasts are made over a period of time, since spatio-temporal conditions are normally unknown. In addition, there is uncertainty due to lack of information as a result of failures in data acquisition or failures in measurement instruments. Prediction based on linear models under these conditions tends to have poor performance.

Neural networks have been successfully applied in the problem of time series prediction [5]–[7]. The Multilayer Perceptron (MLP) is the most widely used type of neural network. The Backpropagation (BP) algorithm is adopted to minimize modeling error and update the weights accordingly. However, the MLP with BP, has two main problems: slow convergence and local minima [8].To avoid these problems, several approaches of machine learning have made considerable efforts to improve the results, such as Deep-Learning [9] and Meta-Learning (ML). The strong and weak convergence of ML and BP with momentum term are given in [10] and [11].

The methods of Meta-Learning and Transfer-Learning (TL) are relatively new ideas from psychology studies to explain how the learning process works by solving new problems based on knowledge and experience [12]. Moreover, the ML algorithm has been proposed to solve the problem of multi-step forecasting [13]. In general, the combination TL or ML has poor exploration in the advancement architectures

The associate editor coordinating the review of this manuscript and approving it for publication was Yiming Tang.

of neural networks; both concepts are used in a qualitative approach [14]–[16].

Strictly speaking, meta-learning is only capable of improving the learning process in the same domain or task. When the information is insufficient to complete the task, or there are many problems in the implementation of the solutions, the transfer-learning method can be used when the tasks and distributions used in the training and testing stages are different. Therefore, the neural model has been given the ability to learn from other problems or tasks. A full description of this topic is provided in [17]. It should be considered that for the transfer-learning method it is necessary to answer the question: What knowledge should be transferred? This represents a challenge for researchers, since determining the similarities or patterns between databases is not an easy task to perform.

The application of deterministic or stochastic methods is not enough for multi-horizon prediction. It is necessary to understand time series in a domain other than time. Better results can be obtained using Wavelet Transform (WT) analysis. An advantage of wavelet analysis is the ability to perform local space-time analysis of time series [18], [19]. The WT allows us to reveal aspects of signal that other analysis techniques overlook, such as trends, breakpoints, discontinuities, etc.

This multiple resolution can also be obtained using WT, called discrete wavelet transform (DWT) [20]. The DWT uses filter banks, while the discrete WT uses discrete versions of the scale and expansion axes. The DWT is a transformation that decomposes a given signal into a number of sets, this technique has been successfully implemented in [21], [22].

The complexity in the prediction of time series increases when dealing with chaotic systems, since the trends and behaviors do not follow the characteristics of seasonality and periodicity.

Since friction is a nonlinear phenomenon [23], earthquakes can be considered a chaotic deterministic system [24] with limited predictability. Therefore, the interpretation of earthquakes can be regarded as a stochastic process or as a deterministic chaotic process [25]. In general, there are two approaches for earthquake forecasting:

1) Earthquakes are considered a stochastic process, where the main shock intervals between events are stationary and typically follow a Poissonian distribution [26]. The earthquakes can be based on some renewal time model that mimics the theory of elastic rebound [27]. Although now nobody can predict exactly next earthquake, some parameters of next big earthquake, such as time interval and magnitudes, can be estimated in the sense of probability based on past seismicity.
2) Earthquakes are considered as the result of a deterministic process, such as the result of a stick slip friction slip [28]. The deterministic predictability of earthquakes remains a debated topic in seismology.

Theoretical and numerical studies based on deterministic equations indicate that stick slippage can be chaotic time series [29]. However, natural climatic earthquakes are explained by a chaotic deterministic time series. The chaotic behavior in regular earthquakes remains a challenge due to the short period of observation time [30].

Long-term forecast events are based on periodically arriving earthquakes, in general, a long-term event is too difficult to predict due to the limited information available. A complete earthquake prediction procedure should have three types of information: magnitude, location, and time of occurrence. Many methods are used to predict earthquakes, such as rule-based approach [31], shallow neural network [32], and deep learning [33]. Many methods use neural network models [34], which have great difficulties due to the rarity of the data, the quality of the historical earthquake data, the lack of pattern and the variability of the performance in different geological locations. The most important challenges are: forecast precision is limited to large magnitude, big forecast error in long-term prediction, the effect of environmental factors, and uncertainty in the factors.

In this work, a new method called Meta-Transfer Learning (MTL) with searching algorithm based on wavelet decomposition to solve the classical problems on multi-horizon forecasting of time series. The method of MTL is a hybrid of ML and TL methods. The Transfer-Learning modified is applied to solve the problem where there are not enough historical data in training domain and in combination with the wavelet decomposition it is possible to have a tool that allows determining what information to use within a set of secondary tasks, this knowledge will improve the accuracy in the prediction of a main task. The Meta-Learning modified helps us to solve the problems of local minima and slow convergence of neural networks.

Comparisons with other classical neural network models are proposed. The comparative analyzes show that: 1) Novel method has better modeling performances than the other algorithms in earthquake forecasting in order to minimize the MSE criterion; 2) The proposed method has a rapid convergence and is capable of achieving the assigned task.

In order to create an effective learning method for neural models, especially for long-term forecasting, we make the following contributions:

1) Meta-transfer learning and neural networks are applied for time series forecasting in the cases of multi-horizon and lacking data.
2) A modification is made for transfer learning with multiple resolution wavelet decomposition, such that the most important information are used to transfer.
3) A modification to meta-transfer learning is made to provide an output to a cycling problem within the algorithm.
4) Some important properties such as stability and the convergence (weak and strong) of the proposed meta-transfer learning are analyzed.
5) The proposed method is successfully applied to earthquake forecasting.

II. MULTI-HORIZON TIME SERIES FORECASTING USING NEURAL NETWORKS

The behavior of a time series $y(1), \ldots, y(N)$ can be described as a dynamic system as:

$$y(k+1) = F[y(k), y(k-1), y(k-2), \ldots, y(k-n^*)] \quad (1)$$

where $F(\cdot) \in C^{\infty}$ is an unknown nonlinear function, n^* is the number of past events needed to make the forecast. The multi-horizon forecasting of the time series $y(1), \ldots, y(N)$ is:

$$\hat{y}(k+1+d_\sigma) = F[y(k-d_\alpha)] \quad (2)$$

where $\hat{y}$ is the prediction value, d_σ is the prediction horizon, d_α is the recursive delay. Such that, $d_\sigma = \{0, 1, 2, \ldots, n_\sigma\}$, $d_\alpha = \{1, 2, \ldots, n_\alpha\}$, n_σ is the maximum horizon, n_α is the number of past events. The multi-horizon forecasting becomes:

$$[\hat{y}(k+n_\sigma), \ldots, \hat{y}(k+1)] = F[X(k))] \quad (3)$$

or:

$$\hat{y}(k+1+d_\sigma) = F[X(k)] \quad (4)$$

where:

$$X(k) = [y(k), \ldots, y(k-n_\alpha)] \quad (5)$$

Because $F(\cdot)$ is unknown, the following neural model is used to approximate it:

$$\hat{y}(k+1+d_\sigma) = NN[X(k)] \quad (6)$$

If $NN(\cdot)$ has a single-layer neural network the model is:

$$\hat{y}(k+1+d_\sigma) = \Gamma[W_k X(k)] \quad (7)$$

where $W_k \in R^n$ is the weight matrix, $\Gamma(\cdot)$ is the activation function.

If $NN(\cdot)$ has a two-layer neural network,

$$\hat{y}(k+1+d_\sigma) = V_k \Gamma[W_k X(k)] \quad (8)$$

where $W_k \in R^{m \times n}$ is the weight matrix of the hidden layer, $V_k \in R^{o \times m}$ is the weight matrix of the output layer.

If $NN(\cdot)$ is deep-neural network the model is:

$$\hat{y}(k+1+d_\sigma) = V_k \Gamma\{W_1 \Gamma_1[\cdots W_l X(k)]\} \quad (9)$$

where l is the number of hidden layers.

The scheme of the time series modeling using neural networks is shown in Figure 1. In this paper, we will use these three types of neural networks (7)-(9) for multi-horizon time series forecasting. The objective of the time series forecasting is to minimize the following modeling error:

$$e(k) = \hat{y}(k) - y(k) \quad (10)$$

For multi-horizon time series forecasting, the modeling error is:

$$e(k+1+d_\sigma) = \hat{y}(k+1+d_\sigma) - y(k+1+d_\sigma) \quad (11)$$

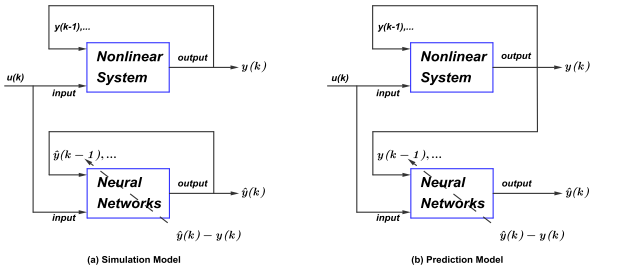

FIGURE 1. Scheme of the time series modeling using neural networks for the simulation model and prediction model. The difference of using real data decides the difficulty of time series forecasting.

The training object of the neural network models is to update the weights W_k and V_k, such that the modeling error is minimized:

$$NN(\cdot) = \arg\min_{W_k, V_k} e^2(k) \quad (12)$$

The following gradient method for (8) can minimize: (12)

$$W_{k+1} = W_k - \eta \frac{\partial J}{\partial W}, \quad V_{k+1} = V_k - \eta \frac{\partial J}{\partial V} \quad (13)$$

where η is the positive learning rate $\eta < 1$, $\frac{\partial J}{\partial V} = \Gamma[W_k X(k)] e(k)$ and $\frac{\partial J}{\partial W} = \Gamma' V_k e(k)$. This is the back-propagation algorithm. To increase convergence speed, the momentum term is added to (13):

$$\begin{aligned} W_{k+1} &= W_k - \eta \frac{\partial J}{\partial W} + \alpha \Delta W_k \\ V_{k+1} &= V_k - \eta \frac{\partial J}{\partial V} + \alpha \Delta V_k \end{aligned} \quad (14)$$

where $\Delta W_k = W_k - W_{k-1}$, α is a constant $0 < \alpha < 1$.

III. NEURAL NETWORK WITH WAVELET DECOMPOSITION

Meta-learning is used to avoid local minima, while transfer-learning is applied for the insufficient information in the training data of neural network models.

Meta-transfer learning brings together the properties and characteristics of the meta-learning and transfer-learning. Our method can be divided into two parts:

1) The modified Transfer-Learning method is responsible for determining what information is relevant to transfer between neural models, through synaptic weights W_s^*. For this, the searching algorithm is based on the Discrete Wavelet Transform using the multilevel decomposition, and with this determine the coefficients (σ_{cA}, σ_{cD}) to compute the deviation standard between different databases. If there is a low standard deviation then the databases have a strong correlation and it is possible to use that information. This stage aims to overcome the problem of lack of information in a time series due to failures in data acquisition.
2) The purpose of the Meta-Learning method is to avoid local minima. This is achieved once the modified Transfer-Learning method selects a matrix of weights

W_s^* called sub-optimal matrix, such that each iteration the weight matrix W_k converges to the sub-optimal weights W_s^* by means of the modified BP learning law due to the addition of terms $\beta_{w,k}^{s*}\hat{X}_{W,k}^{s*}$ associated with Meta-Learning.

A. WAVELET TRANSFORM

For non-stationary and multi-horizon forecasting of real world time series, meta-transfer learning cannot provide good prediction accuracy. We will use wavelet to solve these problems.

A wavelet function $\Psi \in L^2(\mathbb{R})$ is defined as

$$\Psi_{m,n}(x) = 2^{\frac{m}{2}}\Psi\left(2^m x - n\right), \quad \text{for all } m, n \in \mathbb{Z} \tag{15}$$

For a orthogonal basis for $L^2(\Re)$, the function Ψ is also called mother wavelet.

Considering the closed space Z_i, for all $i \in \mathbb{Z}$ the Wavelet base have the following properties:

1) Z_i space sequence is nested,

$$\cdots \subset Z_{-1} \subset Z_0 \subset Z_1 \subset \cdots \tag{16}$$

2)

$$\cap_{m\in\mathbb{Z}} = \{0\}$$

3)

$$f(x) \in Z_k$$

if and only if $f(2x) \in V_{k+1}$

4) $\cup_{k\in\mathbb{Z}} V_k = L^2(\mathbb{R})$.

Using (16), it is possible to build an orthogonal basis for L^2. where W_n is an orthogonal complement from Z_m whit respect from Z_{m+1}:

$$W_m \oplus Z_m = V_{m+1}, \quad W_m \perp Z_m. \tag{17}$$

Thus:

$$L^2 = \oplus_{m\in\mathbb{Z}} W_m, \quad W_m \perp W_{m'} \tag{18}$$

and can be rewritten as:

$$\Psi_n(x) = \Psi(x - n), \quad n \in \mathbb{Z}. \tag{19}$$

then the system $\{\Psi_n\}_{n\in\mathbb{Z}}$ is an orthogonal basis of W_0. Consequently the system $\left\{\Psi_{m,n}(x)\right\}_{n,k\in\mathbb{Z}}$ is an orthogonal basis of the space W_m, therefore, it is an orthogonal basis of L^2.

Any continuous function $f \in L^2[0, 1]$, can be expanded by the series:

$$f = \sum w_{mn}\Psi_{mn}, \tag{20}$$

where the coefficients $w_{m,n}, m, n \in \mathbb{Z}$, can be calculated by the inner product:

$$w_{m,n} = < f, \Psi_{mn} > . \tag{21}$$

As described above the subspace formed by the base:

$$Z_j = \oplus_{n<j} W_n, \tag{22}$$

can be reduced to the trivial space $j \longrightarrow -\infty$, and the series can be written as follows:

$$f = \sum_{m\geq j}\sum_{n\in\mathbb{Z}} w_{mn}\Psi_{mn}. \tag{23}$$

B. WAVELET DECOMPOSITION

The wavelet decomposition is actually the application of the discrete wavelet transform (DWT), but for different scale factors [18]. The DWT can be represented as

$$W_{m.n}^{wav} = 2^{-\frac{m}{2}}\sum_{i=1}^{L} f_i\Psi[2^{-m}i - n] \tag{24}$$

where m represents the scale index, n is the translation variable, Ψ is the wavelet mother, L is the length of the series or the function f.

Haar wavelet [35] is the simplest discrete wavelet transforms. Haar wavelet is the most commonly used. When we need a model which can eliminate the high-frequency noise and avoid the distribution of the rest of the signal, the disadvantages of Haar wavelet are that it is discontinuous, and it does not approximate continuous signals very well.

The Haar wavelet is produced from the Haar mother function:

$$\Psi(x) = \begin{cases} 1 & 0 \leq x < \frac{1}{2} \\ -1 & \frac{1}{2} \leq x < 1 \\ 0 & others \end{cases} \tag{25}$$

where the input has 2^n numbers, it may be considered to simply pair up input values, it operates on data by calculating the sums and differences of adjacent elements. This function is capable to capture the frequency and temporal contents. A typical Haar wavelets is:

$$WH_{m,n}(x) = 2^{m/2}\psi(2^m x - n), \quad n < 2^m$$

where m and n are integers, ψ is defined by (25). However, it is necessary the discrete-time wavelet Haar:

$$WH_{m,n}(x) = W(x - n), \quad n \in \mathbb{Z}$$

the system $\{WH_n\}_{n\in\mathbb{Z}}$ is an orthogonal basis of w_0. Moreover, the system $\{WH_{m,n}(x)\}_{n,k\in\mathbb{Z}}$ is a normal basis of the space w_m, therefore is an orthogonal basis of L^2. Any continuous function $f \in L^2$ can be rewritten by the series:

$$\hat{f} = \sum w_{mn}WH_{mn}$$

where the coefficients $w_{m,n}$ with $m, n \in \mathbb{Z}$, are calculated by the inner product,

$$w_{m,n} = \langle \phi_{m,n}, WH_{m,n} \rangle$$

here, $\phi_{m,n}$ is the Haar wavelet transform, it starts with 2^n array, and performs a process with n iterations of the basic

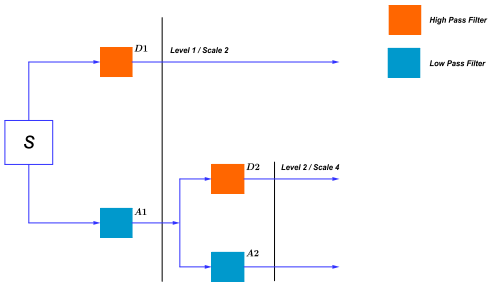

FIGURE 2. Multi resolution wavelet transform decomposition for time series by means of high pass filter and low pass filter. This can help us to use the most important data.

transform. For each index $l \in \{1, \ldots, n\}$, the array structure consists of in coefficients for $2^{n-(l-1)}$ step functions:

$$\phi_{mn}(i) = \begin{cases} 1 & \text{if } 2^{n-m}k \leq 2^{n-m}(k+1) \\ 0 & \text{otherwise} \end{cases}$$

where ϕ_{mn} is also called scaling function. The base Haar can be formed into a subspace:

$$V_j = \oplus_{n<j} WH_n$$

The Haar series in to the trivial space $j \to -\infty$

$$\hat{f}(x) = \sum_{m \geq j} \sum_{n \in \mathbb{Z}} w_{mn} WH_{mn}(x)$$

Without losing of generality, we can assume that $j = 0$, the Haar series is:

$$\hat{f}(x) = \sum_{m=0}^{\infty} \sum_{n \in \mathbb{Z}} w_{mn} WH_{mn}(x)$$

Even though the mathematical wavelet transform concept is applied, it consists of a set of low and high-pass filters [18]. Figure 2 shows an example on how a decomposition of scale 4 (or level 2) for a signal is done. The wavelet decomposition is applied to each Ψ domain.

The wavelet transform uses a broad range of compact orthogonal supporting analyzing wavelets. Orthogonality in DWT causes that the information deduced at a certain scale m, which is disjoint from the information at other scales:

$$\sigma_{wav}(m) = \left[\frac{1}{N-1} \sum_{n=1}^{N} (W_{m.n}^{wav} - \langle W_{m.n}^{wav} \rangle)^2 \right]^{\frac{1}{2}} \quad (26)$$

where N represents the number of wavelet coefficients at a given scale m, $W_{m.n}$ is the average among the coefficients.

The following equations described how to compute the deviation standard using the wavelet coefficients to a pair of domains:

$$\sigma_{cA} = \sqrt{\frac{1}{N-1} \sum_{i=0}^{N} (cA_i - W_{m_1})^2}$$
$$\sigma_{cD} = \sqrt{\frac{1}{N-1} \sum_{i=0}^{N} (cD_i - W_{m_2})^2} \quad (27)$$

where "cA" represents the lowest frequency of the signal, and "cD" is the highest frequency of the signal, $W_{m1} = mean(cA)$, $W_{m2} = mean(cD)$.

IV. NEURAL NETWORK WITH META-TRANSFER LEARNING

A. TRANSFER LEARNING

According to the fundamental property of knowledge transfer, which states that it is possible to use the previously acquired knowledge in an auxiliary task Λ_a and thus help in the performance of the main task Λ_p. Let us define two sets Ψ_a and Ψ_p. The domain Ψ_a is for learning task Λ_a. The principal domain Ψ_p is for the principal learning task Λ_p.

Transfer-Learning for neural model aims to improve the time series forecasting with Λ_p in Ψ_p using the knowledge of Ψ_a and Λ_a. The auxiliary domain data Ψ_a and Ψ_p are:

$$\Psi_a = X(k+1+d_\sigma)_a = [y_a(k-1), \ldots, y_a(k-n)]$$
$$\Psi_p = X(k+1+d_\sigma)_p = [y_p(k-1), \ldots, y_p(k-n)] \quad (28)$$

There is a fundamental problem within the Transfer-Learning technique: how to select previous knowledge acquired by an auxiliary task to improve the performance of a defined task?

In this work, we propose the following method to find the optimal information through the Wavelet Transform, this allows to find information that helps to determine a correlation between the two domains Ψ_a and Ψ_p. In general, there can be n domains Ψ_n across which comparisons can be made to find the best set that guarantees the improvement of the results obtained by the main task.

There can be two ways to interpret the standard deviation: strong correlation and weak correlation, depending on the nature of the time series. We create a domain Ψ_{TF} to generate a hybrid database between Ψ_a and Ψ_b, which is used to the meta-learning and find the optimal weights $W^*$of neural network. In weak correlation, local minima in the main task Λ_p may be avoided.

We assume the time series has a definite time Ts. We use the following function to generate Ψ_{TF}:

$$f\left(\tau_{i_{\Lambda_p}}, \tau_{i_{\Lambda_a}}\right) = \begin{cases} \tau_{i_{\Lambda_p}} & if \geqslant \gamma \\ \tau_{i_{\Lambda_p}} \oplus \tau_{i+\beta_{\Lambda_a}} \oplus \tau_{i+\beta+1_{\Lambda_p}} & if < \gamma \end{cases} \quad (29)$$

The model (29) has information of the time series from the tasks Λ_p and Λ_a. Ψ_{TF} mixes the data from both sets. It depends on the nature of the phenomenon that has been described in the time series, in addition, the selected characteristic is based on the previous knowledge of the researcher in the problem.

B. META LEARNING

The time between events is an important characteristic of time series. We use wavelet transformation for multiple solutions databases Λ_p and Λ_a, then we use Meta-Transfer Learning to consider other characteristics.

Algorithm 1 Transfer-Learning Modified

1: Choose a task defined by Λ_p
2: Propose several tasks Λ_p. This set could be a correlated or not with Λ_p
3: Apply the model (24) to each Λ_p and Λ_{ai}
4: Apply the model (27) to obtain coefficients cA and cD
5: Compare the factor correlation with the model (26)
6: Select the pair Λ_p and Λ_p with more or less correlation. This criterion is chosen by test.
7: Apply the (29) to combine the data-sets of Λ_p and Λ_p
8: Use the model (9)
9: Return the W^*

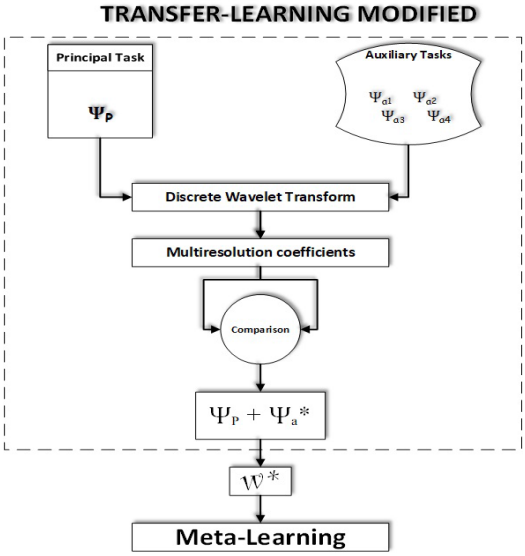

FIGURE 3. Selection of the sub-optimal weights W^* from transfer learning and the search method based on the wavelet transformation. It shows the benefit of the combination.

We use the following method for the inter-event time: when the inter-event time of Λ_p is smaller than γ, the data information of Λ_a it is saved, such that we can know where is the information of Λ_a is, adding to Λ_p. β which is a parameter that indicates the number of data of Λ_a added to Λ_p, see Algorithm 1. Figure 4 shows how to use transfer-learning to find W^*.

After wavelet transformation and transfer learning, we use Meta-Learning and back-propagation to train the neural network models. This is our modified meta-transfer learning, see Algorithm 2.

In order to improve the forecasting accuracy, the following modified back-propagation algorithm is applied, which uses the principal task Λ_p and the knowledge of W^*,

$$W_{k+1} = W_k - \eta \frac{\partial J}{\partial W} + \alpha \Delta W_k + \beta_{w,k} \hat{X}_{W,k}$$
$$\eta_k = \frac{\eta}{1 + \left\| \phi' X^T(k) \right\|^2}, \quad 0 \le \eta_k < \eta \le 1 \qquad (30)$$

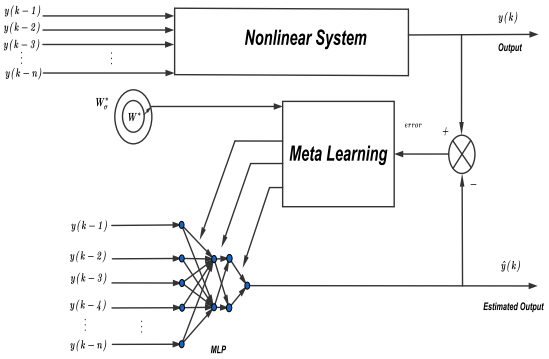

FIGURE 4. Scheme of implementation of the meta learning by the choice of sub-optimal weights for time series forecasting. The time series forecasting can be regarded as nonlinear system modeling.

where η is the positive learning rate $\eta < 1$, $\frac{\partial J}{\partial W} = \phi' \left[W_k \hat{X}(k) \right] e(k)$, $\Delta W_k = W_k - W_{k-1}$, α is a positive constant $\alpha < 1$, $\beta_{W,k}$ is a constant.

$\hat{X}_{W,k}$ is decided by

$$\hat{X}_{W,k} = \max_i \left[\theta_{W,i} \right], \quad \cos\left(\theta_{W,i}\right) = \frac{W^* - W_i}{\|W^* - W_i\|} \frac{\hat{X}_{W,k}}{\left\| \hat{X}_{W,k} \right\|}$$
$$\hat{X}_{V,k} = \max_i \left[\theta_{V,i} \right], \quad cos\left(\theta_{V,k}\right) = \frac{V^* - V_k}{\|V^* - V_k\|} \frac{\hat{X}_{V,k}}{\left\| \hat{X}_{V,k} \right\|}$$

where $\hat{X}_{W,k} \in R^{m \times n}$ is the vector which forces the angle between W_k and W^*, $\theta_{W,k}$, to arrive the maximum value, $\theta_{W,k}$ is the angle.

The Meta-Learning term $(\beta_{W,k} X_{W,k})$ can reduce the forecasting error in each step k for the neural model (2), and produce a fast convergence between the pairs (W_k, W^*). The weights W_k are projected to the sup-optimal weights W^*, *i.e.*, the current weights W_k go towards the desired weights W^* with the direction $\hat{X}_{W,k}$ and the step size $\beta_{W,k}$.

Figure 4 and Figure 3 shows how to apply the modified Meta-Transfer Learning for neural network training.

The following steps show the MTL methodology:

1) We train the neural model (9) with the classical gradient decent algorithm (13) using different initial weights V_0 and W_0, for Ψ_{TF}.
2) We select the best final weights, V^* and W^*,which can minimize the modeling error in the sense of (10). This idea is to extract some properties from previous knowledge.
3) We further train the neural model (9) with the ML algorithm (30). The step size $\beta_{W,k}$ reduces the distance between W_k and W^*,

$$\beta_{W,k} = \left(W^* - W_k\right) \frac{\hat{X}_{W,k}}{\left\| \hat{X}_{W,k} \right\|} \qquad (31)$$

This time-varying term ensures that the angle condition is fulfilled in each step. Similar

$$\beta_{V,k} = \left(V^* - V_k\right) \frac{\hat{X}_{V,k}}{\left\|\hat{X}_{V,k}\right\|}$$

However, to obtain $\beta_{i,k}$ the angular condition Ac is needed. When a deep-neural network model is applied, the size of the vectors $\hat{X}_{i,k}$ increases by dimension due to the number of weights. We need W_k and V_k converge to W^* and V^*. Normal algorithms need long times in the execution. We propose the following modified meta-learning to avoid the aforementioned problem.

Algorithm 2 Compute of $\hat{X}^{s*}_{V,k}$

1: Select an Initial $\hat{X}_{V,0} = 0$
2: Select a coefficient r (number of iterations)
3: **for** r times **do**
4: Choose a random vector $\hat{X}_{V,k}$ $0 \le \hat{X}_{V,k}(i) \le 1$ $i = 1, 2, \ldots, m$
5: Calculate $\left\|\hat{X}_{V,k}(k)\right\|$
6: Compute $cos\Theta_{V,i} = \frac{V^* - V_k}{\|V^* - V_k\|} \cdot \frac{\hat{X}_{V,k}(k)}{\left\|\hat{X}_{V,k}(k)\right\|}$
7: **end for**
8: Select the max$(cos\Theta_{V,i})$
9: Return $\hat{X}^{s*}_{V,k}$

According to the above, the models (31) and (30) can be rewritten as

$$\beta^{s*}_{W,k} = \left(W^* - W_k\right) \frac{\hat{X}^{s*}_{W,k}}{\left\|\hat{X}^{s*}_{W,k}\right\|} \tag{32}$$

And the modified meta-learning is given by,

$$\begin{aligned} W_{k+1} &= W_k - \eta \frac{\partial J}{\partial W} + \alpha \Delta W_k + \beta^{s*}_{w,k} \hat{X}^{s*}_{W,k} \\ \eta_k &= \frac{\eta}{1 + \left\|\phi' X^T(k)\right\|^2}, 0 \le \eta_k < \eta \le 1 \end{aligned} \tag{33}$$

V. CONVERGENCE ANALYSIS

To show the effectiveness of our meta-transfer learning for time series forecasting, we will give strong and week convergence properties of the proposed algorithm.

We first give the following stability result of the meta-transfer learning.

The following theorem gives convergence of the modified meta-learning.

Theorem 1: If the meta-learning algorithms (33)-(32) are applied, the training processes of the neural networks (7)-(9) are stable in the sense of L_∞

$$|e(k)| < \infty \tag{34}$$

Proof 1: For the single layer neural network (7), the meta-learning is

$$\beta^{s*}_{w,k} \hat{X}^{s*}_{W,k} = -\tilde{W}_k \frac{X_p^T X_p}{\|X_p\|} = -\gamma \tilde{W}_k \tag{35}$$

We define the following Lyapunov candidate function,

$$\begin{aligned} L_k = 2\left\|\tilde{W}_k\right\|^2 + 3\left\|\alpha \tilde{W}_{k-1}\right\|^2 + 7\left\|\alpha \tilde{W}_k\right\|^2 \\ + \left\|\gamma \alpha \tilde{W}_{k-1}\right\|^2 + \left\|\gamma \tilde{W}_k\right\|^2 + \tau\left\|\tilde{W}_k\right\|^2 \end{aligned}$$

From the meta-learning update law (33)

$$\tilde{W}_{k+1} = \tilde{W}_k - \eta_k e(k) \varphi' X^T(k) - \alpha \Delta \tilde{W}_k - \gamma \tilde{W}_k$$

The difference term $\Delta V_k = V_{k+1} - V_k$ is

$$\begin{aligned} \Delta L_k &\le (1+\alpha)\left\|\eta_k e(k) \varphi' X(k)\right\|^2 \\ &\quad - \eta_k (1 + \alpha - \gamma) e^2(k) + \eta_k (1 + \alpha - \gamma) \zeta^2(k) \\ &= -\eta_k \left[(1+\alpha) - (1+\alpha-\gamma)\eta \frac{\left\|\varphi' X^T(k)\right\|^2}{1+\left\|\varphi' X^T(k)\right\|^2}\right] e^2(k) \\ &\quad + \eta_k (1 + \alpha - \gamma) \zeta^2(k) \\ &\le -\pi e^2(k) + \eta_k (1 + \alpha - \gamma) \zeta^2(k) \end{aligned}$$

where

$$\pi = (1+\alpha) - (1+\alpha-\gamma)\eta \frac{K}{1+K} > 0$$

$K = \sup_K \left\|\varphi' X^T(k)\right\|^2$. Because

$$\text{Because, } n \min\left(\tilde{w}_i^2\right) \le V_k \le n \max\left(\tilde{w}_i^2\right)$$

where $n \min\left(\tilde{w}_i^2\right)$ and $n \max\left(\tilde{w}_i^2\right)$ are functions of κ_∞, as well as $\pi e^2(k)$ which and $\eta_k \zeta^2(k)$ are κ functions. The Lyapunov function L_k is the function of $e(k)$ and $\zeta(k)$, then L_k is a smooth ISS-Lyapunov function. So, the dynamics of the identification error is an Input-State Stable.

For the multi layer neural networks, we use the following positive defined matrix L_k:

$$L_k = \left\|\tilde{W}_k\right\|^2 + \left\|\tilde{V}_k\right\|^2 \tag{36}$$

From the update law (33):

$$\begin{aligned} \tilde{W}_{k+1} &= \tilde{W}_k - \eta_k e(k) \varphi' P^{0T} X^T(k) \\ &\quad - \alpha \Delta W_k - \gamma W_k \\ \tilde{V}_{k+1} &= P_k - \eta_k e(k) \varphi' X^T(k) - \alpha \Delta V_k - \gamma V_{kk} \end{aligned}$$

Similar development with a single layer neural network:

$$\begin{aligned} &\Delta L_k \\ &\le \eta_k \left[(1+\alpha) - (1+\alpha-\gamma)\eta \frac{\left\|\varphi' P^{0T} X^T(k)\right\|^2}{1+\left\|\varphi' P^{0T} X^T(k)\right\|^2}\right] e^2(k) \\ &\quad + \eta_k (1 + \alpha - \gamma) \zeta^2(k) \\ &\le -\pi e^2(k) + \eta_k (1 + \alpha - \gamma) \zeta^2(k) \end{aligned}$$

Furthermore,

$$\begin{aligned} & n\left[\min\left(\tilde{w}_i^2\right)+\min\left(\tilde{v}_i^2\right)\right] \\ & \leq L_k \\ & \leq n\left[\max\left(\tilde{w}_i^2\right)+\max\left(\tilde{v}_i^2\right)\right] \end{aligned}$$

where

$$n\left[\min\left(\tilde{w}_i^2\right)+\min\left(\tilde{v}_i^2\right)\right]$$

and

$$n\left[\max\left(\tilde{w}_i^2\right)+\max\left(\tilde{v}_i^2\right)\right]$$

are κ_∞ functions, $\pi e^2(k)$ is a κ_∞ function, $\eta_k\left(1+\alpha-\gamma\right)\zeta^2\left(k\right)$ is a κ function. L_k admits a smooth ISS-Lyapunov function, moreover is the function $e(k)$ and $\zeta\left(k\right)$. If the "input" $\zeta\left(k\right)$ is bounded, then the dynamics of the "state" $e(k)$ is bounded.

The weak convergence of the proposed Meta-Transfer learning is given by the following theorem.

Theorem 2: The Meta-Transfer Learning (32-33) are the weak convergence,

$$\lim_{k\to\infty}\left(\tilde{W}_k\right)^2<\infty,\quad \lim_{k\to\infty}\left(\tilde{V}_k\right)^2<\infty \tag{37}$$

i.e., the increments of the weights are bounded, here

$$\tilde{W}_k=W_{k+1}-W_k,\quad \tilde{V}_k=V_{k+1}-V_k$$

Proof 2: For the single layer neural network (7), we use the following Lyapunov function:

$$\begin{aligned} L_k=&2\left\|\tilde{W}_k\right\|^2+3\left\|\alpha\tilde{W}_{k-1}\right\|^2+7\left\|\alpha\tilde{W}_k\right\|^2 \\ &+\left\|\gamma\alpha\tilde{W}_{k-1}\right\|^2+\left\|\gamma\tilde{W}_k\right\|^2+\tau\left\|\tilde{W}_k\right\|^2 \end{aligned}$$

For the multi layer neural networks, we use the following positive defined matrix L_k:

$$L_k=\left\|\tilde{W}_k\right\|^2+\left\|\tilde{V}_k\right\|^2 \tag{38}$$

From the stability proof of Theorem 1:

$$\Delta L_k\leq-\pi e^2\left(k\right)+\eta_k\left(1+\alpha-\gamma\right)\zeta^2\left(k\right) \tag{39}$$

where

$$\pi=(1+\alpha)-(1+\alpha-\gamma)\,\eta\frac{K}{1+K}>0$$

$K=\sup_K\left\|\varphi' X^T(k)\right\|^2$. The update law is input-to-sate stable, and (37) is established. We use the following Lyapunov function:

$$\begin{aligned} L_k=&2\left\|\tilde{W}_k\right\|^2+3\left\|\alpha\tilde{W}_{k-1}\right\|^2+7\left\|\alpha\tilde{W}_k\right\|^2 \\ &+\left\|\gamma\alpha\tilde{W}_{k-1}\right\|^2+\left\|\gamma\tilde{W}_k\right\|^2+\tau\left\|\tilde{W}_k\right\|^2 \end{aligned}$$

From the stability proof of Theorem 1

$$\Delta L_k\leq-\pi e^2\left(k\right)+\eta_k\left(1+\alpha-\gamma\right)\zeta^2\left(k\right) \tag{40}$$

where π and K are defined in (39). Since the modeling error $e(k)$ and L_k are bounded, the gradient term associated with each of the layers due to the ML learning law is bounded with time going to infinity. Therefore, the increment defined by $\tilde{W}_k$ and $\tilde{V}_k$ are bounded, and (37) is established.

The week convergence continues to be fulfilled as long as there is a sufficient number of iterations. The following theorem gives the strong convergence of the proposed meta-transfer learning.

Theorem 3: There exist a $W_{ai}^*\subset\Omega$ such that $W^{s*}\subset W_\sigma$. The meta-learning (33) leads the strong convergence with proper initial conditions and rich input signals,

$$\lim_{k\to\infty}W_k=W^{s*} \tag{41}$$

Proof 3: We define W^{s*} as the sub-optimal weight of W_k at the k. The projection angle θ of the two vectors W_k and W^{s*} is:

$$cos\theta=\frac{W_k\cdot W^{s*}}{\parallel W_k\parallel\parallel W^{s*}\parallel} \tag{42}$$

We also define l as, see Figure 5:

$$\begin{aligned} l&=\parallel W^{s*}\parallel cos\theta=l_x+l_x \\ &=\frac{W^{s*}+W_k}{\parallel W_k\parallel}+\frac{W^{s*}+W_k}{\parallel W_k\parallel}=\frac{W_k\cdot W^{s*}}{\parallel W_k\parallel} \end{aligned}$$

Using the triangular inequality:

$$\frac{W^{s*}W_k}{\|W^{s*}\|\|W_k\|}\leq\frac{W^{s*}}{\|W^{s*}\|\|W_k\|}-\frac{W_k}{\|W^{s*}\|\|W_k\|} \tag{43}$$

Using the updated law (33), the increment is:

$$\Delta W=\frac{W^{s*}\cdot W_{k+1}}{\parallel W^{s*}\parallel\parallel W_{k+1}\parallel}-\frac{W^{s*}\cdot W_k}{\parallel W^{s*}\parallel\parallel W_k\parallel}\geq 0$$

where $k=1,2,\ldots$.Using the Cauchy-Bunyakovsky-Schwarz inequality, the increment of ΔW and Lemma 1, the increment along the sequence is:

$$\begin{aligned} 0\leq&\left\|\frac{W^{s*}W_k}{\|W^{s*}\|\|W_k\|}\right\|^2\leq\sum_{i=k+1}^{\infty}(\Delta W_i^{s*})^2 \\ &\leq\sum_{i=k}^{k+1}\Delta W_i^2\leq\sum_{i=0}^{k}\Delta W_i^2 \end{aligned}$$

Because:

$$\begin{aligned} 0&\leq\|W^{s*}-W_{k+1}\|\leq\|W_{k+1}-W_k\| \\ &\leq\|W_k-W_0\|\leq\|W_0\| \end{aligned} \tag{44}$$

and modeling error is bounded as (34), then (41) is fulfill. From Lemma 1 and

$$\frac{W^*W_k^{s*}}{\|W^*\|\|W_k^{s*}\|}\leq\frac{W^*}{\|W^*\|\|W_k^{s*}\|}-\frac{W_k^{s*}}{\|W^*\|\|W_k^{s*}\|} \tag{45}$$

The difference on the right side of the above inequality is:

$$\delta=\frac{W^*}{\|W^*\|\|W_k^{s*}\|}-\frac{W_k^{s*}}{\|W^*\|\|W_k^{s*}\|} \tag{46}$$

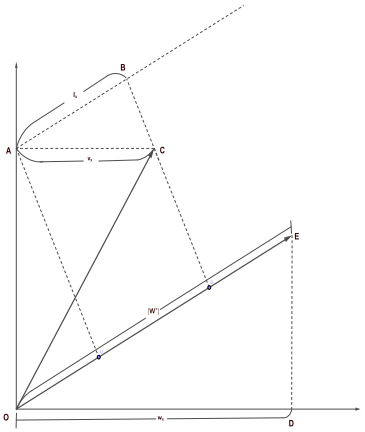

FIGURE 5. Geometric interpretation of the angle between two vectors. It can be explained as the Euclidean norm between both.

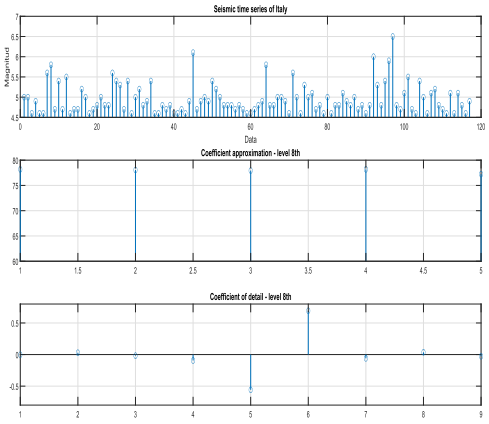

FIGURE 6. Multi-resolution wavelet transform for the seismic data with magnitudes of about 4.5 of the Italian catalogue. We can see the important data.

So

$$\lim_{k\to\infty} W_k^{s*} = W^*$$

Let the vector $\hat{X}_{V,k}$ lead the weights V_k towards V^*. Then the vector $\hat{X}_{V,k}^{s*}$ will drive the weights V_k towards V^{s*}. Therefore, if the number of iterations is sufficient according to the Algorithm 2, then $V^{s*} = V^*$, otherwise the weights $V^{s*} \in \Delta$, where Δ is a closed compact of radius δ such that $\delta < \infty$, Δ being is a neighborhood close to V^* and

$$V^* = V^{s*} + \delta$$

VI. EARTHQUAKE PREDICTION USING META-TRANSFER LEARNING

We use the proposed method to forecast the earthquakes in Italy (M>4.5) by using the data both from Italy and Mexico. The data of Italy are extracted by the publicly available database in ''cnt.rm.ingv.it/en/iside'', the data of Mexico are extracted by the publicly available database in ''http://www.ssn.unam.mx''.

The motivation of using both datasets of Italy and Mexico for Italy is the available earthquake data of Italy are not sufficient for neural network models. We add Mexico earthquake data to the time series of Italy. Normally, it is not reasonable, because these two time series are corresponding to different models. However, we successfully combine three techniques: wavelet decomposition, meta-learning, and transfer-learning, such that these two earthquake datasets can be applied to train one neural network model.

The time series for the M>4.5 data of the Italian seismic catalog contains a quantity of 104 elements during 1970-2018. This available information may become insufficient to make a multi-horizon prediction, if Figure 9 is analyzed, through visual inspection it can be seen that there is no trend in the prediction, since the neural models classics can get it right due to a test datum and in the immediately subsequent datum fail due to a considerable error. The idea of modified TL is to generate a data set that shares the information of the Mexican and Italian catalogs, through which the weights of W_s^* necessary for the ML method can be determined. To choose the information, a search algorithm based on the multi resolution of the DTW is proposed. The WT allows to identify spatio-temporal features within a time series, thus extracting relevant information from a time series that cannot be easily obtained with the simple analysis of discrete-time data. In [18], they use the standard deviation to compare the levels obtained from the multi-resolution of the DTW. since lower scales are associated with higher frequency oscillations, the increase in $\sigma_{wav}(m)$ with scale indicates that higher frequency fluctuations are less strong than lower frequency ones. Similarly, in this work they allow to determine the correlation between databases and thus determine what information should be transferred by the MTL method. Until this moment of the investigation there is no way to determine which values σ_{cA}, σ_{cD}) of the standard deviation should be chosen to obtain a good performance in the method. For this paper, the closer the deviance coefficients of the Mexican and Italian datasets are, the higher their correlation will be, and therefore there is a high similarity in characteristics or properties between the time series.

A. WAVELET DECOMPOSITION FOR MULTI-HORIZON TIME SERIES

In other words, the standard deviation, in this case, let us know which set of seismic data of Mexico is similar to the seismic information of Italy. If similar information in terms of standard deviation is added to the data set of Italy, it will be fed with similar events, but for correctly training a neural network, diversity in the signal is necessary. Therefore, the time series of Mexico that is selected is the one with a major standard deviation from the seismic information of Italy.

The eighth level of decomposition was achieved. In Figure 6 the decomposition of the seismic data of Italy is shown. In Figure 7, the two sets of data with the higher standard deviation from the information of Italy are shown. The information of Mexico in 2016 is selected.

	Desv(cA)	Desv(cD)
Italia:	0.338842	0.045574
México 2001:	1.152003	0.144298
México 2002:	1.847408	0.307796
México 2003:	2.248482	0.282674
México 2004:	3.640420	0.488308
México 2005:	1.315854	0.184348
México 2006:	3.050858	0.433966
México 2007:	1.057269	0.373970
México 2008:	1.035213	0.153787
México 2009:	0.637653	0.095475
México 2010:	1.368199	0.201624
México 2011:	1.205139	0.135587
México 2012:	0.622432	0.122014
México 2013:	2.083414	0.297961
México 2014:	0.624838	0.055370
México 2015:	0.201995	0.034680
México 2016:	0.296783	0.072113
México 2017:	0.733177	0.086301
México 2018:	1.133294	0.227045
México 2019:	0.364960	0.118925
México 2020:	2.990503	0.430463

FIGURE 7. Comparison of coefficients of the standard deviation of Italy (green) and Mexico (red) seismic data. It is the motivation of transfer learning.

We will show the standard deviation of the wavelet coefficients from two data sets, the inter-event time of both data sets, as well as the created functions from two time series.

1) **The inter-event**, time allows us to graphically find a condition to add the seismic information of Mexico in 2016 in the information of Italy. The inter-event time has the information of the time intervals between successive seismic events. When the inter-event time of Italy is smaller than an inter-event called γ, the magnitude information of Mexico is saved in an array, which will be full of not only the
2) **magnitude** of the earthquakes registered but also zeros. Then, the position is detected where the magnitude information is saved in the last array so that this parameter allows us to know where to add the seismic information of Mexico in the information of Italy. After that, a β parameter indicates the number of seismic data of Mexico added to the data set of Italy. Finally, the new signal to train the neural network is ready and is constructed according the model (20) and the Algorithm 2.

B. META-TRANSFER LEARNING FOR USING THE DATA OF MEXICO TO TRAIN THE MODEL OF ITALY

1) TRANSFER-LEARNING

$$f\left(\tau_{i_{Italy}}, \tau_{i_{Mexico}}\right) = \begin{cases} \tau_{i_{Italy}} & if \geqslant \gamma \\ \tau_{i_{Italy}} \oplus \tau_{i+\beta_{Mexico}} & \\ \oplus \tau_{i+\beta+1_{Italy}} & if < \gamma \end{cases} \quad (47)$$

2) META-LEARNING

Table 1 shows the comparisons of the developed algorithm with some well-known methods. Only 10 events are taken for the testing stage. The experimentation takes into account the information available for a magnitude window $M > 4.5$

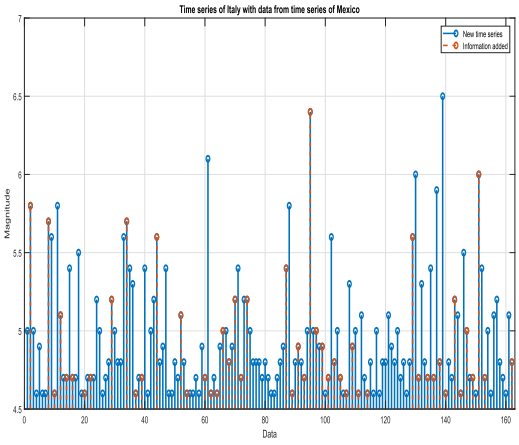

FIGURE 8. Result of the application of the modified transfer learning and the search algorithm based on wavelet transformation. The subsequent mixture to form the time series is associated with Ψ_{TF}.

TABLE 1. Comparison of modeling errors for the different neural models. Here our meta-transfer learning has the best performance in the sense of mean squared error(MSE).

Neural Model	MSE
MLP	0.36
Deep NN 1	0.35
Deep NN 2	0.33
Transfer-Learning NN	0.098
Meta-Learning NN	0.091
Meta-Transfer Learning	0.032

The neural networks are proposed with the following characteristics:

- The neural model MLP and Meta-Transfer Learning has three layers: $W \in \Re^{40\times 15}$ and $V \in \Re^{40\times 5}$.
- The neural model Deep NN 1 has four layers: $W \in \Re^{60\times 15}$, $R \in \Re^{60\times 60}$ and $V \in \Re^{60\times 5}$.
- The neural models Deep NN 2, Meta-Learning NN and Transfer-Learning NN has five layers: $W \in \Re^{90\times 15}$, $S \in \Re^{90\times 60}$, $R \in \Re^{60\times 60}$ and $V \in \Re^{60\times 5}$.

The time series for M>4.5 from Italy only has 115 events in in a period of 50 years. For the training stage, 100 events have been used. The neural model: MLP, Deep NN 1, Deep 2 and Transfer-Learning NN they need epochs of training, in this case 2000 were used. On the other hand, the neural models Meta-Learning NN and Meta-Transfer Learning they do not need epochs in training stage. For all experiments, the learning constants $\eta = 0.35$ and $\alpha = 0.1$ were used.

The experiments are repeated at least 15 times to reach the repeatability of the results from the same conditions in the aforementioned hyper-parameters. With this way, we can avoid random errors in the predictions. Also we use the mean squares error (MSE) as performance index.

C. RESULTS DISCUSSION AND FINAL REMARKS

1) As shown in Figure 9 and in Table 1, the performance of the classical methods (MLP, Deep NN_1 Deep NN_2, Transfer-Learning and Meta-Learning) is not

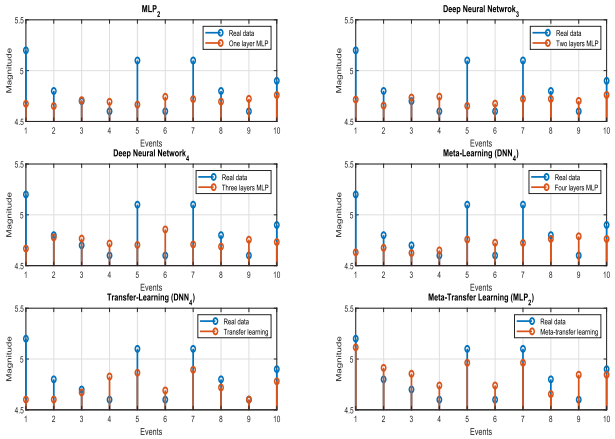

FIGURE 9. Testing stage of the time series with magnitudes of about 4.5 in Italy. The comparison is between the classical neural models and the proposed meta-transfer learning.

satisfactory, because the amount of information contained in the historical database for earthquakes with $M > 4.5$ from Italy is insufficient. In general, when forecasting the magnitude of earthquakes, a deviation between the actual data and the estimate implies a release of more or less energy due to the nonlinear nature of the earthquakes. Our meta-transfer learning minimizes the MSE performance to 0.060. It is much better than the other methods.

2) A advantage of meta-transfer learning are that it gives neural network the ability to use knowledge from another data set, and thus improve the accuracy of time series forecasting. The proposed method, through the search method based on the multi resolution wavelet transform, poses a parity between the data set on which the prediction of the time series is intended to be made, and the data set are extracted. Properties and characteristics are transformed into knowledge and experience. Therefore, it intends to analyze the selected time series in a space-time domain to determine a correlation that cannot be determined in another domain. With the above method, W_s^* is found.
3) The meta-transfer learning method allows taking advantage of the experience acquired between neural models, to improve the accuracy in the forecast stage of a task. This is achieved because the synaptic weights W of the ANN converge to the suboptimal synaptic weights through the projection generated by the meta-learning-based learning law on W_s^*.
4) There are two main limitations of the proposed approach:
   - This method requires a large number of similar data for meta-training which is costly
   - Each neural model is a low complexity base learner, such as shallow neural network, to avoid model over-fitting. So it is unable to use deeper and more powerful architectures.
5) There are also some implementation aspects:
   - The selected databases must have an intrinsic relationship, however, to determine it, it is necessary to have adequate knowledge about the phenomenon that is intended to be predicted and thus make a selection of information with logic sense. Otherwise there can be no connection between the information and therefore the results can be worse than the classical methods.
   - The hyper-parameters such as the number of hidden layers, the learning constants and the number of neurons per layer are the ones to be chosen, so it is necessary to do tests to determine the appropriate ones for the task.
   - The computational cost can increase if you have too much information about it for the secondary database from which the best in W^* is obtained, since the databases must be compared individually together with the original database.
   - To determine the amount of information to be added according to equation (47) it is necessary to experiment until an acceptable response is obtained in the forecast of the time series.

VII. CONCLUSION

In this paper, the multi-horizon time series forecasting is realized by a deep neural networks with Meta-Transfer Learning and wavelet decomposition. We successfully solved the common problem in time series forecasting with missing data and long-term prediction. The proposed method is applied to predict earthquake magnitude with two different data sets. The future works will focus on studying its adaptation properties of the meta-transfer method for the automatic control, to improve the results of identification and control systems based on neural networks.

REFERENCES

[1] P. Nath, P. Saha, A. I. Middya, and S. Roy, "Long-term time-series pollution forecast using statistical and deep learning methods," *Neural Comput. Appl.*, vol. 33, no. 19, pp. 12551–12570, Oct. 2021.
[2] G. E. P. Box, G. C. Reinsel, G. M. Ljung, and G. M. Jenkins, *Time Series Analysis: Forecasting and Control*. Hoboken, NJ, USA: Wiley, 2015.
[3] A. Tealab, "Time series forecasting using artificial neural networks methodologies: A systematic review," *Future Comput. Informat. J.*, vol. 3, no. 2, pp. 334–340, Dec. 2018.
[4] L. Ljung, *System Identification Theory for User*. Englewood Cliffs, NJ, USA: Prentice-Hall, 1987.
[5] M. F. Mushtaq, U. Akram, M. Aamir, H. Ali, and M. Zulqarnain, "Neural network techniques for time series prediction: A review," *JOIV, Int. J. Informat. Visualizat.*, vol. 3, no. 3, pp. 314–320, Aug. 2019.
[6] S. Ben Taieb, A. Sorjamaa, and G. Bontempi, "Multiple-output modeling for multi-step-ahead time series forecasting," *Neurocomputing*, vol. 73, nos. 10–12, pp. 1950–1957, Jun. 2010.
[7] M. P. Clements, P. H. Franses, and N. R. Swanson, "Forecasting economic and financial time-series with non-linear models," *Int. J. Forecasting*, vol. 20, no. 2, pp. 169–183, Apr. 2004.
[8] W. Bi, "Avoiding the local minima problem in backpropagation algorithm with modified error function," *IEICE Trans. Fundamentals Electron., Commun. Comput. Sci.*, vol. E88-A, no. 12, pp. 3645–3653, Dec. 2005.
[9] B. Lim and S. Zohren, "Time series forecasting with deep learning: A survey," *Phil. Trans. Roy. Soc. A*, vol. 379, Apr. 2020, Art. no. 20200209.

[10] W. Wu, H. Shao, and D. Qu, "Strong convergence of gradient methods for BP networks training," in *Proc. Int. Conf. Neural Netw. Brain*, Oct. 2005, pp. 332–334.
[11] W. Wu, N. Zhang, Z. Li, L. Li, and Y. Liu, "Convergence of the gradient method with momentum for back-propagation neural networks," *J. Comput. Math.*, vol. 26, no. 4, pp. 613–623, Jul. 2008.
[12] Y. Ganin, E. Ustinova, H. Ajakan, P. Germain, H. Larochelle, F. Laviolette, M. Marchand, and V. S. Lempitsky, "Domain-adversarial training of neural networks," *J. Mach. Learn. Res.*, vol. 17, no. 59, pp. 1–35, 2016.
[13] M. Maya, W. Yu, and L. Telesca, "Multi-step forecasting of earthquake magnitude using meta-learning based neural networks," *Cybern. Syst.*, pp. 1–18, Oct. 2021, doi: 10.1080/01969722.2021.1989170.
[14] A. R. Ali, B. Gabrys, and M. Budka, "Cross-domain meta-learning for time-series forecasting," *Proc. Comput. Sci.*, vol. 126, pp. 9–18, Jan. 2018.
[15] C. Lemke and B. Gabrys, "Meta-learning for time series forecasting and forecast combination," *Neurocomputing*, vol. 73, nos. 10–12, pp. 2006–2016, Jun. 2010.
[16] R. G. Mantovani, A. L. D. Rossi, E. Alcobaça, J. Vanschoren, and A. C. P. L. F. de Carvalho, "A meta-learning recommender system for hyperparameter tuning: Predicting when tuning improves SVM classifiers," *Inf. Sci.*, vol. 501, pp. 193–221, Oct. 2019.
[17] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Trans. Knowl. Data Eng.*, vol. 22, no. 10, pp. 1345–1359, Oct. 2009.
[18] L. Telesca, V. Lapenna, and N. Alexis, "Multiresolution wavelet analysis of earthquakes," *Chaos, Solitons Fractals*, vol. 22, no. 3, pp. 741–748, Nov. 2004.
[19] P. Addison, *The IllustratedWavelet Transform Handbook*. Boca Raton, FL, USA: CRC Press, 2002.
[20] J. Francoise, G. Naber, and T. T., *Wavelets: Applications*. New York, NY, USA: Academic, 2006.
[21] S. Padma, G. Hariharan, and S. Castellucci, "A new spectral method applied to immobilized biocatalyst model arising in biochemical engineering," *Contemp. Eng. Sci.*, vol. 10, no. 1, pp. 291–304, Jan. 2017.
[22] G. Hariharan, *Analysis—An Overview. In: Wavelet Solutions for Reaction–Diffusion Problems in Science and Engineering*. Springer, 2019.
[23] J. H. Dieterich, "Modeling of rock friction: 1. Experimental results and constitutive equations," *J. Geophys.*, vol. 84, pp. 2161–2168, May 1978.
[24] B. K. Shivamoggi, *Nonlinear Dynamics and Chaotic Pheonmena: An Introduction*. New York, NY, USA: Springer, 2014.
[25] A. Gualandi, J.-P. Avouac, S. Michel, and D. Faranda, "The predictable chaos of slow earthquakes," *Sci. Adv.*, vol. 6, no. 27, Jul. 2020, Art. no. eaaz5548.
[26] J. K. Gardner and L. Knopoff, "Is the sequence of earthquakes in Southern California, with aftershocks removed, Poissonian?" *Bull. Seismol. Soc. Amer.*, vol. 64, no. 5, pp. 1363–1367, 1974.
[27] E. Field, G. Biasi, P. Bird, T. Dawson, K. Felzer, D. Jackson, K. Johnson, T. Jordan, C. Madden, A. Michael, K. Milner, M. Page, T. Parsons, P. Powers, B. Shaw, W. Thatcher, R. Weldon, and Y. Zeng, "Long-term time-dependent probabilities for the third uniform California earthquake rupture forecast (UCERF3)," *Bull. Seismol. Soc. Amer.*, vol. 105, pp. 511–543, Apr. 2015.
[28] W. F. Brace and J. D. Byerlee, "Stick-slip as a mechanism for earthquakes," *Science*, vol. 153, no. 3739, pp. 990–992, Aug. 1966.
[29] N. Kato, "Earthquake cycles in a model of interacting fault patches: Complex behavior at transition from seismic to a seismic slip," *Bull. Seismol. Soc. Amer.*, vol. 106, pp. 1172–1787, Aug. 2016.
[30] J. Huang and D. L. Turcotte, "Evidence for chaotic fault interactions in the seismicity of the san Andreas fault and Nankai trough," *Nature*, vol. 348, no. 6298, pp. 234–236, Nov. 1990.
[31] B. Rahmat, F. Afiadi, and E. Joelianto, "Earthquake prediction system using neuro-fuzzy and extreme learning machine," in *Proc. Int. Conf. Sci. Technol. (ICST)*, 2018, pp. 452–458.
[32] A. Mignan and M. Broccardo, "Neural network applications in earthquake prediction (1994–2019): Meta-analytic insight on their limitations," 2019, *arXiv:1910.01178*.
[33] Y. Wang, Z. Wang, Z. Cao, and J. Lan, "Deep learning for magnitude prediction in earthquake early warning," 2020, *arXiv:1912.05531*.
[34] M. H. A. Banna, K. A. Taher, M. S. Kaiser, M. Mahmud, M. S. Rahman, A. S. M. S. Hosen, and G. H. Cho, "Application of artificial intelligence in predicting earthquakes: State-of-the-art and future challenges," *IEEE Access*, vol. 8, pp. 192880–192923, 2020.
[35] I. Daubechies and C. Heil, "Ten lectures on wavelets," *Comput. Phys.*, vol. 6, no. 6, p. 697, 1992.

MARIO MAYA received the B.S. degree in control and automation engineering from the Instituto Politécnico Nacional (IPN), CDMX, Mexico, in 2015, and the M.Sc. degree in automatic control from the Center for Research and Advanced Studies, Instituto Politécnico Nacional (CINVESTAV-IPN), CDMX, Mexico, in 2017. He is currently pursuing the Ph.D. degree in automatic control with CINVESTAV-IPN.

WEN YU (Senior Member, IEEE) received the B.S. degree in automatic control from Tsinghua University, Beijing, China, in 1990, and the M.S. and Ph.D. degrees in electrical engineering from Northeastern University, Shenyang, China, in 1992 and 1995, respectively. From 1995 to 1996, he worked as a Lecturer with the Department of Automatic Control, Northeastern University. Since 1996, he has been with CINVESTAV-IPN (National Polytechnic Institute), Mexico City, Mexico, where he is currently a Professor with the Departamento de Control Automatico. From 2002 to 2003, he held a research position with the Instituto Mexicano del Petroleo. He was a Senior Visiting Research Fellow with Queen's University Belfast, Belfast, U.K., from 2006 to 2007, and a Visiting Associate Professor with the University of California at Santa Cruz, Santa Cruz, from 2009 to 2010. He has been holding a visiting professorship with Northeastern University, China, since 2006. He serves as an Associate Editor of IEEE TRANSACTIONS ON CYBERNETICS, *Neurocomputing*, and *Journal of Intelligent and Fuzzy Systems*. He is a member of the Mexican Academy of Sciences.