



# Introducing Temporal Correlation in Rainfall and Wind Prediction From Underwater Noise

Andrea Trucco , Senior Member, IEEE, Annalisa Barla , Roberto Bozzano , Sara Pensieri , Alessandro Verri, and David Solarna, Member, IEEE

**Abstract**—While in the past the prediction of wind and rainfall from underwater noise was performed using empirical equations fed with very few spectral bins and fitted to the data, it has recently been shown that regression performed using supervised machine learning techniques can benefit from the simultaneous use of all spectral bins, at the cost of increased complexity. However, both empirical equations and machine learning regressors perform the prediction using only the acoustic information collected at the time when one wants to know the wind speed or the rainfall intensity. At most, averages are made between spectra measured at subsequent times (spectral compounding) or between predictions obtained at subsequent times (prediction compounding). In this article, it is proposed to exploit the temporal correlation inherent in the phenomena being predicted, as has already been done in methods that forecast wind and rainfall from their values (and sometimes those of other meteorological quantities) in the recent past. A special architecture of recurrent neural networks, the long short-term memory, is used along with a data set composed of about 16 months of underwater noise measurements (acquired every 10 min, simultaneously with wind and rain measurements above the sea surface) to demonstrate that the introduction of temporal correlation brings significant advantages, improving the accuracy and reducing the problems met in the widely adopted memoryless prediction performed by random forest regression. Working with samples acquired at 10-min intervals, the best performance is obtained by including three noise spectra for wind prediction and six spectra for rainfall prediction.

**Index Terms**—Acoustical meteorology, machine learning (ML), rainfall intensity prediction, regression, temporal correlation, underwater noise, wind speed prediction.

Manuscript received 23 May 2022; revised 23 September 2022; accepted 8 November 2022. Date of publication 10 January 2023; date of current version 14 April 2023. This work was supported in part by the Italian framework Programma Nazionale di Ricerca in Antartide under Grant PNRA18\_00154, in part by the European Commission within the Horizon 2020 Programme (2014–2020) under Grant 101008724 (MINKE—Metrology for Integrated marine maNagement and Knowledge-transfer nEtwork), and in part by EMSO-ERIC. (Annalisa Barla, Roberto Bozzano, Sara Pensieri, Alessandro Verri, and David Solarna contributed equally to this work.) (Corresponding author: Andrea Trucco.)

**Associate Editor: T. Ross.**

Andrea Trucco and David Solarna are with the Department of Electrical, Electronic, Telecommunications Engineering, and Naval Architecture (DITEN), University of Genoa, 16145 Genoa, Italy (e-mail: andrea.trucco@unige.it; david.solarna@gmail.com).

Annalisa Barla and Alessandro Verri are with the Department of Informatics, Bioengineering, Robotics, and System Engineering (DIBRIS), University of Genoa, 16146 Genoa, Italy (e-mail: annalisa.barla@unige.it; alessandro.verri@unige.it).

Roberto Bozzano and Sara Pensieri are with the National Research Council of Italy, Institute for the Study of Anthropic Impact and Sustainability in the Marine Environment (CNR-IAS), 16149 Genoa, Italy (e-mail: roberto.bozzano@cnr.it; sara.pensieri@cnr.it).

Digital Object Identifier 10.1109/JOE.2022.3223406

## I. INTRODUCTION

THE prediction<sup>1</sup> of wind speed and precipitation intensity using underwater noise measurements has received considerable attention over the years. Setting up sensor networks to gather these meteorological quantities, with high spatial and temporal resolution and without the need for instruments installed above the sea surface, is becoming increasingly important [1], [2], with a view to preventing environmental risks and monitoring climate change. For places with harsh working conditions for surface instrumentation and poor satellite coverage, such as in polar waters, measuring underwater noise may even be the only way to infer wind speed and rainfall intensity [3], [4], [5].

For many years, this prediction was carried out using empirical equations linking the desired quantity, be it wind or rain, to the noise intensity measured at a precise frequency [3], [4], [5], [6], [7], [8], [9], [10], [11], [12], [13] or, in rare cases, at a few frequencies [14]. Due to their simplicity, these equations have proven to be effective tools in predicting wind and rain with good accuracy, especially in the most frequent ranges. Recently, it was proved that solutions based on supervised learning techniques, exploiting the information contained in all frequencies of the measured noise spectrum, can improve prediction accuracy and make it more robust against interferences (e.g., passing ships). In particular, Taylor et al. [15] proposed random forest (RF) and CatBoost to predict wind and rainfall from hourly averaged noise spectra. RF [16], [17] was also used in [18] for rainfall detection from hourly averaged noise spectra, as well as in [19] for wind prediction. In the latter case, the averaging of predictions obtained from instantaneous noise spectra (i.e., prediction compounding) was adopted because it provided better performance than the traditional averaging of spectra (i.e., spectral compounding) before the prediction.

<sup>1</sup>According to the machine learning (ML) community, this article reserves the term *estimation* to the choice of the best parameters involved in the empirical equation or ML model, using a set of training data. Instead, the term *prediction* is adopted to indicate the calculation of the output variable using the already built model and a vector of input variables. In this article, the wind speed or rainfall intensity at a time  $T$  is predicted using a given equation or model and the underwater noise measurements, gathered at time  $T$  or until  $T$  (i.e., from  $T - \Delta t$  to  $T$ ), as input data. The term *forecast* is adopted to indicate the future evolution of wind and rain, i.e., the computation of the wind speed or rainfall intensity at a time  $T$ , using a given model and physical quantities gathered until  $T_0$ , with  $T_0 < T$ , as input data. Forecasting is a particular type of prediction. Although the method proposed in this article is not concerned with forecasting, some of the literature cited about the temporal correlation of wind and rain deals with forecasting.

Both empirical equations and ML regression techniques predict wind and rainfall at the time the underwater noise is measured, possibly averaging over a given time interval (the most common is 1 h) to improve accuracy and robustness. The innovative contribution of this article is to introduce temporal correlation in the wind and rainfall prediction: to obtain the prediction at a given time interval, the underwater noise samples collected in several previous intervals are individually exploited. To model the temporal correlation of wind speed and rainfall intensity, and to set the regressor that takes into account the current and previous noise samples, it was decided to use the ML tools already adopted for wind and rainfall forecasting in the meteorological field.

To the best of the authors' knowledge, no one has previously tried to use the temporal memory of wind and rainfall to improve predictions performed by using underwater noise. Instead, the temporal correlation has been exploited to forecast wind and rain using the present and past values of these two quantities and, in some cases, the values of other meteorological quantities. Concerning the hourly average of wind speed, the autocorrelation functions estimated at seven different locations are shown in [20]. Based on the observation that the correlation coefficient (CC) generally remains above 0.8 in the first hour and above 0.5 in the first 4 h, several techniques have been proposed [21], [22], [23], [24], [25], [26], [27] for forecasting wind speed, from a few minutes to a few hours ahead, using the past values of the wind speed itself. Both methods for the statistical analysis of time series, such as autoregressive moving-average, and ML models, such as support vector machines and artificial neural networks, have been tested and compared. In recent years, the long short-term memory (LSTM) architecture, a recurrent neural network (RNN) used in the field of deep learning, has attracted particular attention [25], [26], [27]. In addition, the advantage of averaging subsequent forecasts (at 10-min intervals) to obtain the hourly average forecast has been reported [23].

Similarly, for rainfall intensity, both statistical analysis of time series and ML models have been proposed to forecast precipitation over successive days [28], [29], [30]; successive hours [31], [32], [33], [34], [35]; or sequences of shorter intervals [36]. For short-term forecasting (i.e., limited to a few hours ahead), the input variables generally consist of the rainfall intensity, and sometimes other meteorological quantities, recorded over a period going from the last 45 min to the last 3 h [31], [32], [33], [34], [35], [36]. As for wind, the most recent contributions [29], [30], [34], [35] focus on the LSTM network. According to the results reported in previous papers, Barrera-Animas et al. [35] state that forecast models based on LSTM networks "outperform other models in the task of forecasting rainfall on an hourly, daily, and monthly basis."

This article exploits a sequence of underwater acoustic noise spectra covering a period of approximately 16 months with a 10-min time interval, acquired at a depth of 36 m on a 1200-m seabed. The acoustic data are synchronized with measurements of wind speed and rainfall intensity taken at the same location, using a sonic anemometer and a rain gauge mounted 10 m above sea level. This data set has already been used to predict hourly averages of wind speed and rainfall intensity [11], [15], [18], [19]

by applying both empirical equations and ML techniques, in both cases without considering temporal memory. The best results for wind prediction are those reported in [19] where the RF regression [16], [17] and the average of consecutive predictions (namely, six predictions at 10-min intervals were averaged to yield the hourly average) were adopted. For rainfall prediction, the best results in [15] were obtained with an RF regressor fed by the hourly averaged spectra, using only a fraction of the mentioned data set. Both in [15] and [19], as well as in all other papers using underwater noise, only noise samples collected during the interval in which the wind or rainfall is predicted are exploited. In contrast, the LSTM network [37], [38] is used in this work to model the temporal correlation of the two meteorological phenomena under consideration and exploit antecedent noise samples for their prediction.

The assessment of LSTM networks for wind and rainfall prediction was performed both in chronological order (i.e., using the first noise samples for the model training and the remaining samples to test it) and by  $K$ -fold cross-validation (i.e., the data set is split into  $K$  subsets.  $K-1$  subsets are used for the model training and the remaining one to test it; this is repeated until each subset has been used to test the model [17]). For comparison, the same testing approach is applied to an RF regressor, which is considered by the literature to be one of the best options [15], [19] when temporal memory is not considered. With both LSTM and RF, six consecutive predictions are averaged [19] to compute the hourly average of wind speed and rainfall intensity.

The importance of introducing temporal correlation lies in a significant improvement in the prediction accuracy, both in general and in cases where wind speed is particularly high. The main limitation reported for the RF regression [19] is the underprediction for high wind speeds, probably due to the low probability of such values occurring. The LSTM network significantly reduces this problem. In addition to increased accuracy, robustness is also an important feature of the method proposed in this article: all acoustic signals acquired are used to predict wind and rainfall, with no samples discarded because they are altered by noise from passing ships, the presence of animals, or other sources. In this way, the temporal resolution in wind and rainfall prediction is maintained and any possible error in the prior classification of acoustic samples (e.g., discarding samples that are not really blurred by ship noise) is avoided.

However, the dependence of the results on the location where the acoustic system is deployed remains to be verified. If the acoustic propagation and meteorological conditions that characterize the location are kept almost unaltered, it is reasonable to assume that the trained system may retain performance similar to that reported in this article. If, on the other hand, the propagation and meteorological conditions change significantly, partial or total retraining will be necessary and performance fluctuations cannot be excluded.

The rest of this article is organized as follows. Section II describes the experimental data set, the models for wind and rainfall prediction encompassing the temporal correlation, and the performance assessment. Section III reports the results obtained and discusses the importance of modeling the temporal correlation. Finally, Section IV concludes this article.

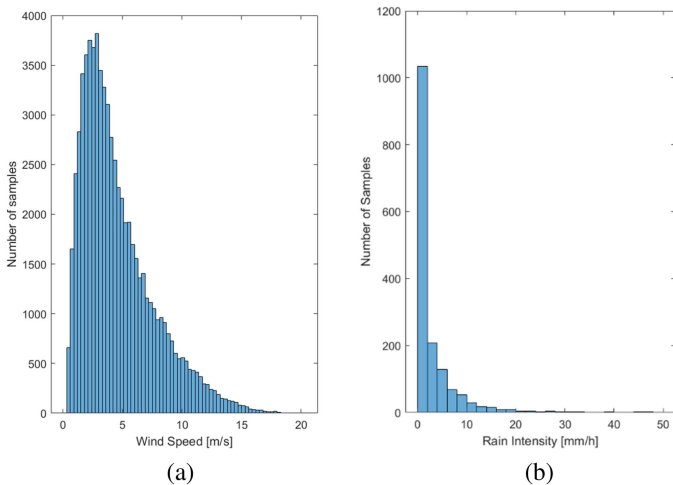


Fig. 1. Distribution of (a) wind speed and (b) rainfall intensity in the period from June 17, 2011 to October 10, 2012. Only rainfall samples greater than 0.1 mm/h are considered.

## II. MATERIAL AND METHODS

### A. Experimental Measurements

Underwater noise, wind speed, and rainfall intensity were gathered from June 17, 2011 to October 10, 2012. In this time window, data were collected without a fixed time step, common to all instruments, but in such a way that it is possible to organize them in 10-min intervals. The data acquisition process experienced only a few short interruptions, usually for system maintenance, the longest of which was about one day (i.e., May 10, 2012). The total number of samples recorded at 10-min intervals is 69 120. All the sensors were installed on the WIM3A meteo-oceanographic observatory part of the EMSO-ERIC network of Eulerian stations [39]. This buoy was moored on a deep seabed of 1200 m, about 80 km off the Ligurian coast, in the northwestern part of the Mediterranean Sea (see details in [40]).

The wind speed was measured by a WindSonic 2-D sonic anemometer, mounted 10 m above sea level on the buoy trellis. At the same height, the rain gauge Vaisala Raincap Sensor was mounted, comprised of a Vaisala Weather Transmitter WXT520, used to measure the rainfall intensity [11], [41]. Anemometer measurements acquired at 5-s intervals were averaged within the 10-min intervals in which the shared time grid is organized. Analogously, rain gauge measurements acquired at 5-s intervals were accumulated within the 10-min intervals and expressed in millimeters per hour. These measurements are assumed to be the ground truth and will be referred to as the actual wind speed and rainfall intensity values. With reference to the whole period considered, Fig. 1 shows the distribution of wind speed measurements, ranging from 0.4 to 20.7 m/s, and that of rainfall intensity measurements above 0.1 mm/h (i.e., the output resolution of the deployed sensor), with an average of 2.5 mm/h and a maximum of 51.5 mm/h.

The underwater acoustic noise was acquired by a dedicated oceanic recorder, based on passive aquatic listener technology [12], [41], [42], clamped to the body of the platform at a depth of 36 m. The recorder was powered by an internal battery assuring long-term operation and equipped with a low-noise wideband hydrophone (Hi-Tech-92WB) with a sensitivity of  $-160$  dB

relative to  $1 \text{ V}/\mu\text{Pa}$ . The acoustic system was set up to acquire an acoustic noise snapshot every 10 min, but when significant noise changes were automatically detected, the system reduced the interval between successive snapshots [41], [42]. For this reason, in the available data set, there are on average seven acoustic noise snapshots per hour. Each snapshot consists of a time series of 4.5 s, sampled at 100 kHz, which is processed on board to obtain a spectrum composed of 64 frequency bins, with a resolution of 0.2 kHz from 0.1 to 3 kHz and 1 kHz from 3 to 50 kHz. When more than one acoustic snapshot is contained within a given 10-min interval, the spectrum associated with that interval is obtained by averaging the available data and is referred to as an instantaneous spectrum. Examples of the collected spectra, depending on wind and rain, are shown and discussed in [18] and [19].

As the anemometer and rain gauge data are acquired every 5 s and averaged/cumulated, respectively, over the 10-min interval, wind and rain variations on a time scale of less than 10 min are filtered out and are no longer detectable in the ground truth data. In contrast, acoustic noise is acquired for a duration of 4.5 s only once (or at most very few times) in a 10-min interval. Unlike the anemometer and rain gauge, phenomena with a time scale of less than 10 min can strongly influence noise measurements. From a spatial point of view, anemometer and rain gauge data are acquired exactly at the location of the buoy. In contrast, the noise collected by the hydrophone at a depth of 36 m is mainly generated inside a circle on the sea surface (centered on the buoy) with a radius of approximately 100 m [11].

Therefore, while the data from surface sensors (i.e., ground truth data) come from a precise location but have undergone temporal average, the noise data have a precise temporal origin but have undergone spatial average. These discrepancies may contribute to the poor agreement between predictions and ground truth data that has been reported when working on single 10-min intervals [19] and may explain why averaging operations over longer time intervals (typically, 1 h) achieve significantly better agreement [19].

### B. Recurrent Neural Networks (RNNs) and LSTM Networks

Deep-learning architectures are inspired by the processing schemes of biological neural networks. They are generally structured in multiple layers to progressively extract higher-level information from the raw input data. Each layer is composed of a collection of units, called artificial neurons, that are connected to the units of the adjacent layers. In analogy with their biological counterparts, artificial neurons perform local processing on the signals that are received from the upstream units and then transmit such processed signals to the downstream neurons in the network. The local processing is parametrized by a set of weights and bias terms.

Specifically, each unit performs a weighted average of the input data and applies a nonlinear activation function after adding a bias term. Let  $\mathbf{x} = [x_1, x_2, \dots, x_n]$  be an input vector and  $g(\cdot)$  be the nonlinear activation function, then the output  $y_u$  of a single neuron is computed according to

$$y_u = g \left( \sum_{i=1}^n w_i x_i + b \right) = g(\mathbf{w} \cdot \mathbf{x} + b) \quad (1)$$

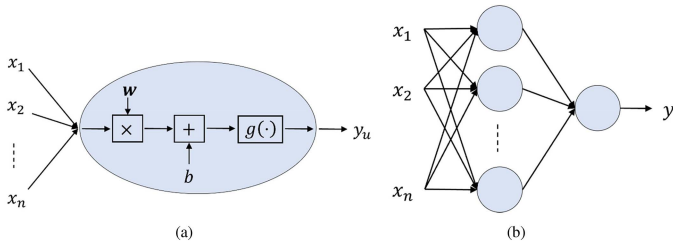


Fig. 2. (a) Single processing unit. (b) Overall architecture of a feed-forward network.

where  $\mathbf{w} = [w_1, w_2, \dots, w_n]$  is a vector containing all the weights  $w_i$  and  $b$  is the bias term. In a multilayer (deep) network, both the weights and the bias terms of all the units in all the layers (namely  $W$  and  $B$ ) are optimized during the training phase by minimizing a loss function  $Q$  based on a labeled training set

$$(W^*, B^*) = \arg \min_{(W, B)} Q(\mathbf{y}^*, \mathbf{y}) \quad (2)$$

where  $\mathbf{y}^*$  is the collection of labels in the training data set and  $\mathbf{y}$  is the collection of output provided by the network when each element of the training set is fed as input. In particular, each element  $y^*$  is the label that, according to the training set, corresponds to the input  $\mathbf{x}$  and  $y$  is the output of the network when  $\mathbf{x}$  is fed as input. Fig. 2 shows an example of a feed-forward network with details on the single processing unit and overall network architecture.

RNNs [43] are a family of neural networks that have been specifically designed to process sequential data. Just as convolutional networks [44] can scale to images with large sizes, RNNs can scale to long sequences, enabling applications that would be impractical for standard networks without sequence-based specialization. The main foundation lies in the parameter-sharing strategy. In case separate parameters are used for each time step within the input sequence, it would not be possible to generalize to lengths that are not present in the training data, nor to share statistical strength across different sequence lengths or across different positions in time. Indeed, parameter sharing is particularly important when a specific piece of information can occur at many different positions [38].

Following the philosophy of dynamical systems, RNNs are based on the concept of state. In fact, even if they share parameters across time, they propagate the state information along the sequence. The state is then used to augment the output with information related to the whole input sequence. For the sake of simplicity, let us focus on a scalar input sequence  $x = \{x^1, x^2, \dots, x^n\}$  and a scalar output sequence  $y = \{y^1, y^2, \dots, y^n\}$ . At each time index  $t$ , the output is computed according to

$$h^t = g(w_x \cdot x^t + w_h \cdot h^{t-1} + b_x) \quad (3)$$

$$y^t = g(w_y \cdot h^t + b_y) \quad (4)$$

where  $h^t$  is the hidden state at the time index  $t$  and  $(w_x, w_h, w_y, b_x, b_y)$  are the weights and bias terms that are shared across the different time steps. Unlike the case of the feed-forward neural network above, the weights and bias terms are divided into multiple sets due to the increased number of

computations that are required to integrate the state information in the processing scheme. The extension to the case of nonscalar inputs and multiple layers is straightforward and only requires substituting the scalar weights with vectors as seen previously. A graphical representation of the simplified case is provided in Fig. 3, where both the unfolded sequence and the recursive representation are shown.

One of the main drawbacks of RNNs as seen so far is their difficulty in handling long-term dependencies [45]. LSTM networks are a special kind of RNN that can overcome this limitation by learning such long-term dependencies [37]. The general idea is similar to the RNN case, with the recurrent structure and the parameter-sharing strategy across different time steps. Nevertheless, LSTM networks have been designed to augment state information with the concept of memory. Indeed, they do propagate both the hidden state  $h^t$  and a cell state  $c^t$  from one time step to the following.

The architecture of an LSTM unit is depicted in Fig. 4, where a comparison with the simple RNN unit is shown. For the sake of simplicity, the weights are omitted in the representation of the LSTM unit. Each LSTM unit contains a cell state and three gates implemented via sigmoidal units  $\sigma(\cdot)$ : a forget gate  $f^t$  controlling the amount of memory to keep; an input gate  $i^t$  controlling the amount of memory that the current input can modify; and an output gate  $o^t$  deciding which part of the cell state to use to update the hidden state, and thus to compute the output. It is worth noting that the output of a sigmoidal unit ranges between zero and one, thus determining the amount of information that should be let through the resulting gate for each component.

For ease of notation and consistency with the RNN case above, let us consider a univariate input sequence  $x = \{x^1, x^2, \dots, x^n\}$  and a univariate output sequence  $y = \{y^1, y^2, \dots, y^n\}$ . The unit takes the input value at the current time step  $x^t$  and the hidden state at the previous time step  $h^{t-1}$  to update the internal hidden state and compute the output according to the following equations:

$$f^t = \sigma(w_f [h^{t-1}, x^t] + b_f) \quad (5)$$

$$i^t = \sigma(w_i [h^{t-1}, x^t] + b_i) \quad (6)$$

$$\tilde{c}^t = \tanh(w_c [h^{t-1}, x^t] + b_c) \quad (7)$$

$$c^t = f^t c^{t-1} + i^t \tilde{c}^t \quad (8)$$

$$o^t = \sigma(w_o [h^{t-1}, x^t] + b_o) \quad (9)$$

$$h^t = o^t \tanh(c^t) \quad (10)$$

$$y^t = g(w_y \cdot h^t + b_y) \quad (11)$$

where  $(w_f, w_i, w_c, w_o, w_y)$  and  $(b_f, b_i, b_c, b_o, b_y)$  are the learnable weight vectors and bias terms, and  $[h^{t-1}, x^t]$  represents a vector containing the previous hidden state  $h^{t-1}$  and the input  $x^t$ . Also, in this case, the extension to the multivariate case is straightforward. The rationale of said processing scheme is that, through the forget gate  $f^t$ , the LSTM unit decides which part of the cell state (i.e., the memory) to forget. Then, a new candidate memory  $\tilde{c}^t$  is computed based on the hidden state and input value. Therefore, the cell state is updated according to the

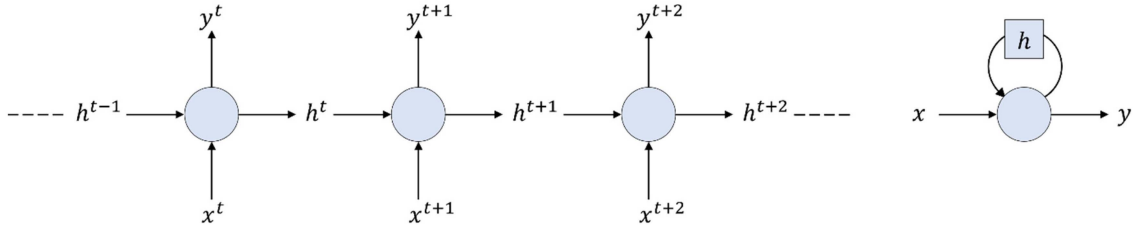


Fig. 3. Graphical representation of an RNN: unfolded (left) and compact (right).

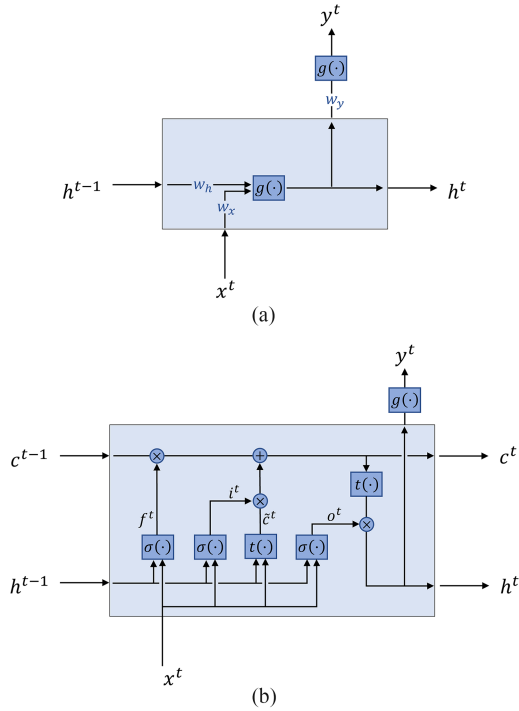


Fig. 4. Comparison between (a) standard RNN unit and (b) LSTM unit.

filtering behavior of the input gate  $i^t$ . Finally, the hidden layer is updated based on the updated cell state  $c^t$  and according to the output gate  $o^t$ .

### C. LSTM Architecture for Wind and Rainfall Prediction

In [19], it was demonstrated that the prediction obtained by averaging a sequence of successive predictions, each generated by feeding the instantaneous noise spectrum into the regressor, is better than the prediction generated by feeding the average noise spectrum into the regressor. Namely, the wind speed on an hourly basis was computed by averaging six predictions obtained from instantaneous spectra acquired every 10 min, as shown in Fig. 5. In this scheme, the model (e.g., an RF regressor) generates predictions without considering the spectra of previously recorded noise.

To integrate time correlation into the prediction process, it is possible to design an LSTM architecture in such a way that predictions made every 10 min can be generated using not only the most recent instantaneous spectrum but also a given number of previous instantaneous spectra. The average of successive predictions (all obtained taking into account the temporal

correlation of the phenomenon) will provide the compounded prediction, for instance on an hourly basis.

Fig. 6 shows an example of a many-to-one LSTM network designed to predict hourly average wind speed or rainfall intensity using instantaneous underwater noise spectra, organized in 10-min intervals. With this kind of LSTM network, an input sequence of a given length produces a single instantaneous output that, thanks to the state information, depends on the whole input sequence. Using  $t$  to identify a current time index in the 10-min grid, each instantaneous prediction  $\hat{y}^t$  is computed based on  $L$  instantaneous spectra: the current spectrum and the last  $L - 1$  spectra. Let  $x$  denote a sample, and let also  $x = [S(f_1), \dots, S(f_{N_F})]$  be a vector composed of the  $N_F$  spectral bins  $S(f_k)$ ,  $k = 1, \dots, N_F$  (in this study,  $N_F = 64$ ). Denoting with  $x^t$  the sample collected at time  $t$ , let  $\mathbf{X}^t = [x^{t-L+1}, \dots, x^t]$  be the collection of  $L$  consecutive samples with  $t$  as last time index. Then, the instantaneous prediction  $\hat{y}^t$  is computed on the basis of the sequence  $\mathbf{X}^t$ . Specifically, Fig. 6 provides a graphical representation of the proposed network in case the compounding is done on an hourly basis and the input sequence has length  $L = 3$ .

It is worth noting that two main hyperparameters are influencing the architecture of the proposed network: the number of LSTM units  $N_h$  and the number of dense units  $N_d$ . The former specifies the dimension of the compressed representation  $h^t$ , whereas the latter specifies the number of hidden neurons in the feed-forward neural network that is applied to the output of the LSTM network to generate the prediction. Indeed, the output of the LSTM network is not scalar, whereas the target of the regression task is the estimation of a scalar quantity (i.e., the wind speed or rainfall intensity). Therefore, a simple feed-forward network is applied to the output of the LSTM to reduce the dimensionality and compute the prediction. Fig. 7 depicts the complete architecture (Fig. 6 proposed a simplified representation for the sake of readability). The LSTM output is fed to a dense hidden layer composed of  $N_d$  units, whose output is processed by a single neuron to produce the scalar output.

### D. RF Regression

In [15], [18], and [19], it was shown that detection and regression by ML techniques, exploiting all the spectral bins, achieve better performance than those obtained from empirical rules and equations, which use a very limited set of spectral bins. Of the many ML techniques tested for wind and rainfall prediction, RF has always provided satisfactory results, often the best among those collected. For this reason, in this article the results obtained from the LSTM networks, exploiting temporal

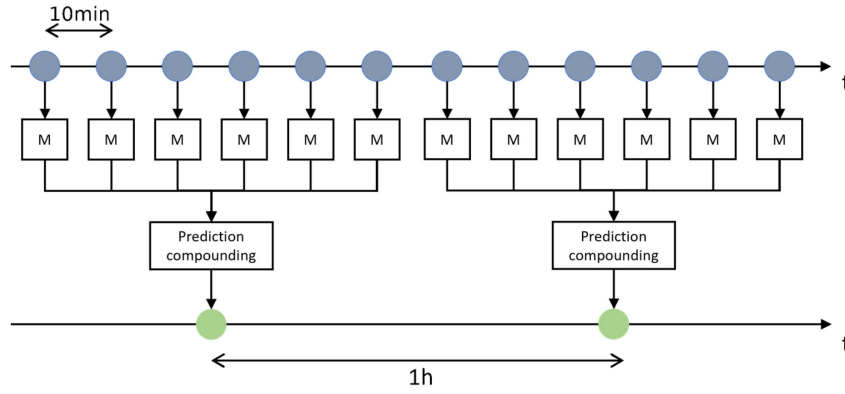


Fig. 5. Prediction scheme in which the hourly average (green dot) is obtained compounding six predictions generated by the model M (e.g., an RF regressor) fed by the instantaneous spectrum (blue dot). Temporal memory is not taken into account.

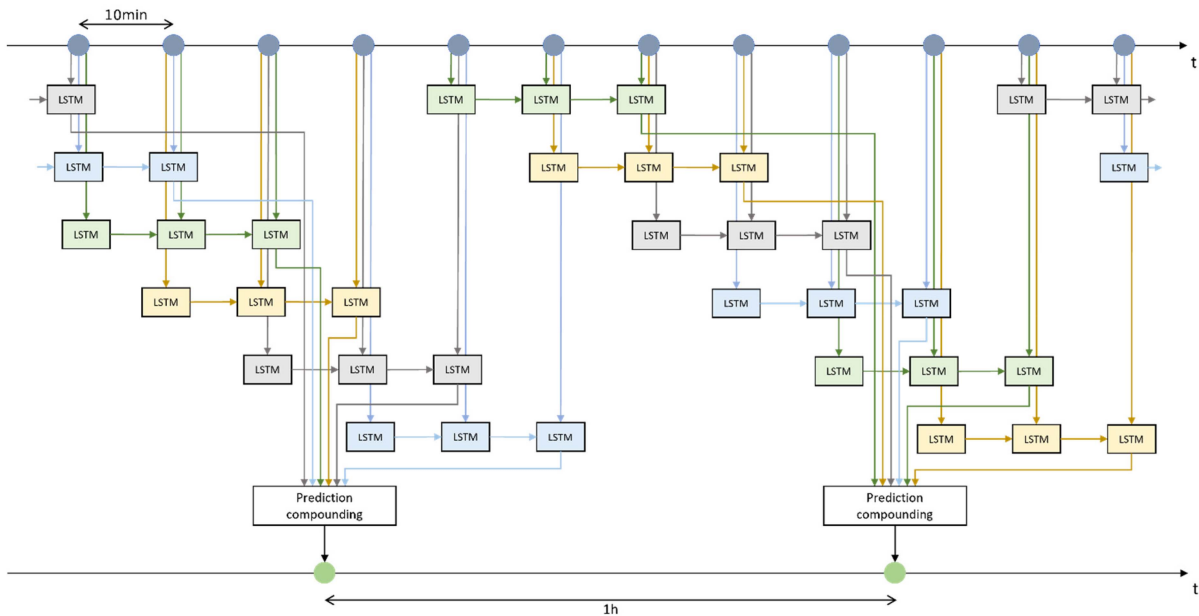


Fig. 6. Prediction scheme in which the hourly average (green dot) is obtained compounding six predictions generated by an LSTM network fed by the last three instantaneous spectra (blue dots). Although they are the same network, LSTM networks applied for prediction at successive time indexes have different colors for ease of visualization.

correlation (as shown in Fig. 6), will be compared with those provided by an RF regressor, trained with the same samples, without considering any memory (as shown in Fig. 5). An introduction to RF regression in relation to wind prediction can be found in [19]; a more methodological, rigorous, and comprehensive RF description can be found in [16] and [17].

As with LSTM, with RF the hourly average wind speed or hourly average rainfall intensity is also calculated from the average of six instantaneous predictions at 10-min intervals (i.e., through prediction compounding), as shown in Figs. 5 and 6.

*E. Performance Assessment*

Four different metrics are adopted to assess the accuracy and bias of the prediction approaches mentioned above: root mean squared error (RMSE), mean absolute error (MAE), Pearson CC, and mean error (ME). Denoting with  $y_i$  the  $i$ th hourly

average of the actual wind speed or rainfall intensity,  $\hat{y}_i$  the corresponding prediction (obtained by averaging a given number of instantaneous predictions  $\hat{y}^t$ ; namely, six predictions in this study),  $H$  the total number of hourly samples,  $\mu_y$  the average of the  $H$  actual values  $y_i$ , and  $\mu_{\hat{y}}$  the average of the  $H$  predictions  $\hat{y}_i$ , the four metrics are computed as follows:

$$RMSE = \sqrt{\frac{1}{H} \sum_{i=1}^H (\hat{y}_i - y_i)^2} \tag{12}$$

$$MAE = \frac{1}{H} \sum_{i=1}^H |\hat{y}_i - y_i| \tag{13}$$

$$CC = \frac{\sum_{i=1}^H (\hat{y}_i - \mu_{\hat{y}})(y_i - \mu_y)}{\sqrt{\sum_{i=1}^H (\hat{y}_i - \mu_{\hat{y}})^2 \sum_{i=1}^H (y_i - \mu_y)^2}} \tag{14}$$

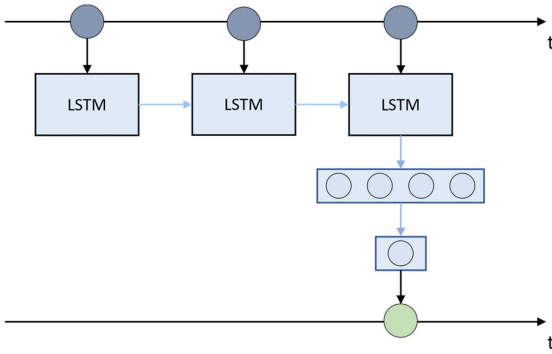


Fig. 7. Complete and unfolded architecture of the proposed LSTM-based model for wind speed and rainfall intensity prediction.

$$\text{ME} = \frac{1}{H} \sum_{i=1}^H \hat{y}_i - y_i. \quad (15)$$

The accuracy is assessed through RMSE and MAE, which measure, in two different ways, the distance between actual values and the corresponding predictions. The prediction bias is measured through the ME. Finally, CC is a normalized metric that evaluates the tendency of the actual quantity and prediction to be simultaneously greater than, or simultaneously less than, their respective mean values. These metrics are used when the model assessment is performed both in chronological order and by  $K$ -fold cross-validation [17]. In the latter case, the metrics are computed on the whole data set, but never involve training data in the prediction test. Because the data set is split in  $K$  subsets nonoverlapped and of approximately equal size (referred to as folds), it is possible to evaluate the dispersion of the metrics: the RMSE, MAE, CC, and ME can be computed  $K$  times, one for each fold used for the test, and, in the end, their averages and standard deviations will be available.

### III. RESULTS AND DISCUSSION

Although the total number of data set samples is 69 120, acquired at 10-min intervals, the number of hours populated by 6 samples is 11 509, involving 69 054 samples. Due to the need to work with homogeneous hourly samples, the remaining 66 samples are discarded. Thus, 69 054 instantaneous noise spectra, 11 509 hourly averaged wind speed actual values, and 11 509 hourly cumulative rainfall intensity actual values are used for the experimentation of the prediction models. It is worth noting that of the 69 054 instantaneous spectra, none are discarded due to the presence of noise from passing ships, animals, or other sources.

When the chronological order is adopted to split the data set into a training set and a test set, the samples collected during the first 9400 h (i.e., approximately 13 months) are used to compose the training set and those collected during the last 2000 h, ranging from July 19, 2012 to October 10, 2012, are used to compose the test set.

All the experiments are run on a PC with an Intel Core i7-4790 CPU and 24 GB of RAM memory. The programming language

is Python, and the software is run in an Anaconda virtual environment. The Scikit-learn library is used for the experiments related to the RF regression, whereas the TensorFlow library is used for the LSTM implementation. In this configuration, the training times of the LSTM and the RF models are approximately 20 min and 15 s, respectively.

#### A. Wind Speed Prediction

The hyperparameters of the LSTM network have been optimized following a grid search strategy and taking into consideration the chronological order scenario. In particular, the hyperparameters that have been considered are as follows:

- 1) the length of the input sequence for the LSTM model,  $L$ , which is selected among 2, 3, 6, and 9;
- 2) the batch size, selected between 32 and 64;
- 3) the number of LSTM units,  $N_h$ , between 64 and 128;
- 4) the number of dense units,  $N_d$ , between 32 and 64.

Concerning  $N_h$  and  $N_d$ , the choices are meant to test the tradeoff between a simpler architecture aimed at keeping overfitting under control and a more complex architecture that may prevent high bias error [38]. Values outside the considered ranges were experimented during a preliminary analysis of the hyperparameters and resulted in high-bias and high-variance models, with poor performance. Conversely, the batch size controls the way in which the training data are handled during the training process and influences the dynamics of the learning algorithm. The backpropagation process allows one to differentiate the loss function and compute the gradients used to update the weights of the neural network. The estimation of such gradients improves with the number of training samples used in each update, i.e., the batch size. While a larger batch size improves the estimate of the gradients, a smaller batch size helps reduce the time and memory requirements of the training procedure. Indeed, the weights are updated more frequently (i.e., once every  $bs$  samples, with  $bs$  being the batch size, not once every time the entire data set is processed) and only a subset of the samples (i.e.,  $bs$  samples) is loaded in memory to compute the loss function and estimate the gradients.

Table I reports the performance of the combinations of the hyperparameters that provide the 5 lowest RMSE values: the configuration providing the best accuracy is given by a sequence length  $L = 3$ , a batch size of 64, several LSTM units  $N_h = 64$ , and a set of  $N_d = 32$  dense units. Setting  $L = 3$  means performing the instantaneous prediction using the noise samples recorded at the current time, 10 min before and 20 min before. The validity of the choice is confirmed in Fig. 8, in which the RMSE and CC as a function of  $L$  are shown, averaged over all values of the other hyperparameters (i.e., averaging over all possible models, once the length of the sequence  $L$  is fixed). This result is consistent with the conclusions of [23] where, working with wind speed measurements in 10-min intervals, the best forecast (the one 10 min ahead) was obtained using the last two or three measurements.

Concerning the hyperparameters of the RF model, according to the analysis performed in [19] using the same data set, 50 trees are used and 32 features (out of the 64 spectral bins) are randomly

TABLE I  
LSTM NETWORK CONFIGURATIONS THAT PROVIDED THE LOWEST RMSE VALUES WHEN THE WIND PREDICTION IS ASSESSED IN CHRONOLOGICAL ORDER

| Sequence length, $L$ | Batch size | No. of LSTM units, $N_h$ | No. of dense units, $N_d$ | RMSE [m/s] | CC    | MAE [m/s] | ME [m/s] |
|----------------------|------------|--------------------------|---------------------------|------------|-------|-----------|----------|
| 3                    | 64         | 64                       | 32                        | 0.810      | 0.949 | 0.641     | 0.006    |
| 2                    | 64         | 128                      | 64                        | 0.817      | 0.949 | 0.657     | -0.126   |
| 3                    | 32         | 64                       | 32                        | 0.819      | 0.947 | 0.650     | -0.016   |
| 2                    | 32         | 128                      | 32                        | 0.820      | 0.948 | 0.655     | -0.087   |
| 6                    | 64         | 128                      | 64                        | 0.820      | 0.948 | 0.658     | -0.071   |

The corresponding CC, MAE, and ME values are also reported.

TABLE II  
WIND PREDICTION METRICS FOR THE LSTM NETWORK AND RF REGRESSOR, ASSESSED IN CHRONOLOGICAL ORDER AND BY  $K$ -FOLD CROSS-VALIDATION

|                         | RMSE [m/s]        | CC                | MAE [m/s]         | ME [m/s]          |
|-------------------------|-------------------|-------------------|-------------------|-------------------|
| LSTM (chron. order)     | 0.810             | 0.949             | 0.641             | 0.006             |
| RF (chron. order)       | 0.847             | 0.945             | 0.659             | 0.008             |
| LSTM (cross-validation) | $0.843 \pm 0.068$ | $0.955 \pm 0.013$ | $0.665 \pm 0.050$ | $0.036 \pm 0.136$ |
| RF (cross-validation)   | $0.896 \pm 0.102$ | $0.952 \pm 0.013$ | $0.696 \pm 0.062$ | $0.028 \pm 0.168$ |

In the latter case, the average  $\pm$  the standard deviation is reported.

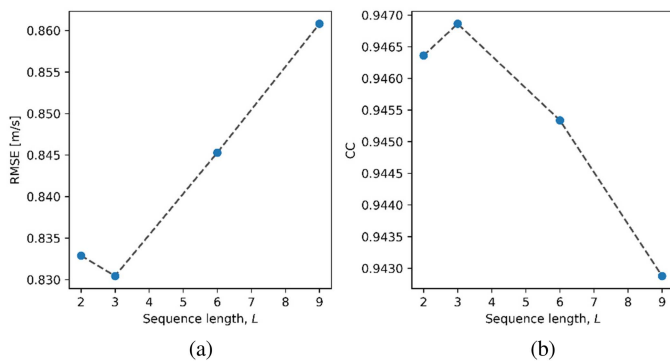


Fig. 8. (a) RMSE and (b) CC for the wind prediction in chronological order, as a function of the extent of the temporal correlation considered, i.e., the sequence length  $L$ . The values of RMSE and CC are averaged over all values of the other hyperparameters.

selected at each node. As demonstrated in [19] a change in these values, even over wide intervals, does not significantly affect the performance.

When the chronological order scenario is applied (training with the first 9400 h and test with the last 2000 h), the performance achieved by the LSTM network and RF model, configured as described above, are summarized in Table II. RF performance is almost identical to that obtained in [19], working on the same training and test data. The values of the four metrics for the LSTM regressor are better than those obtained from the RF regressor. Fig. 9 provides visual confirmation of this fact, showing the scatter plots for the two prediction models, along with the lines that best fit the predictions (i.e., the regression lines) and the quadrant bisectors (black lines).

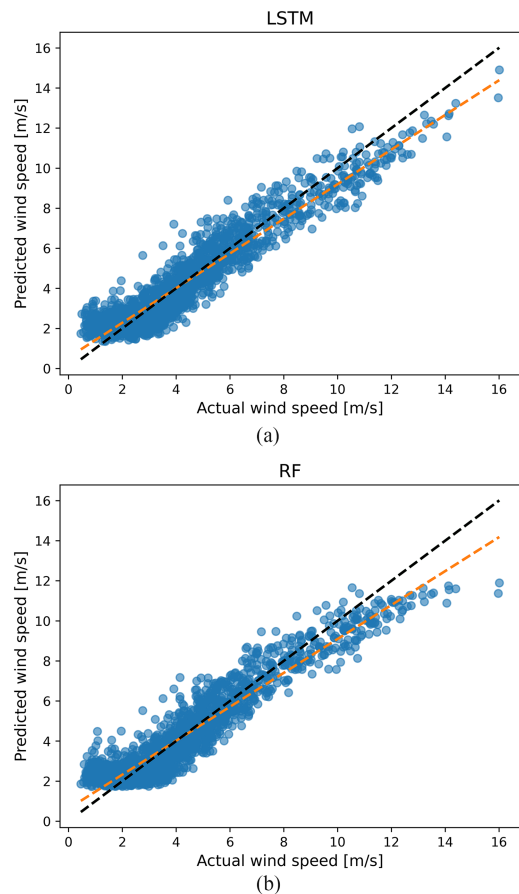


Fig. 9. Scatter diagrams of the wind predictions generated by (a) LSTM network and (b) RF regressor in the chronological order assessment. The lines that best fit the predictions are inserted (golden lines) in comparison with quadrant bisectors (black lines).



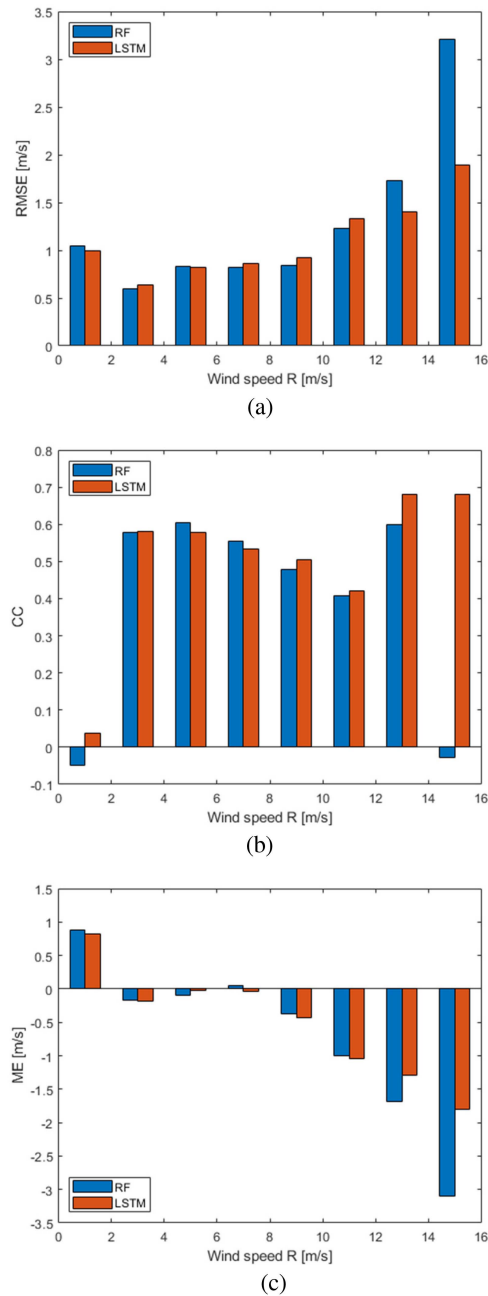


Fig. 10. Performance metrics for the chronological order assessment, computed considering the samples whose actual wind speed is in  $[R, R+2]$ ,  $R = 0, 2, 4, \dots$  m/s. (a) RMSE. (b) CC. (c) ME.

The main problems encountered in [19] with RF regression are the failure to predict the wind when its speed is below 2 m/s and the considerable underprediction for wind speeds above 12 m/s (resulting in increased RMSE for the corresponding samples). Fig. 10 shows the RMSE, CC, and ME for samples whose actual wind speed is included in  $[R, R+2]$ ,  $R = 0, 2, 4, \dots$  m/s, comparing the predictions performed by RF and LSTM. On the one hand, the LSTM regression does not solve the problem in the interval 0 to 2 m/s: in fact, it is well known that wind produces underwater noise only when its speed is greater than a threshold value, between 2 and 3 m/s [4], [8],

[9], [11], [12], [19], below which acoustic prediction becomes impossible. On the other hand, the introduction of the temporal correlation through the LSTM network solves, to a large extent, the underprediction problem<sup>2</sup> above 12 m/s, especially in the interval 14 to 16 m/s. While the ME reduction implies a lower prediction bias, the RMSE decrease implies a better prediction accuracy. In addition, the LSTM regression provides a CC close to 0.7 for speeds greater than 12 m/s, whereas the CC of the RF regressor in the interval 14 to 16 m/s was approximately 0.

To test the two prediction models using all the samples composing the data set (thus testing the models in all seasons of the year), the  $K$ -fold cross-validation strategy can be adopted, as already mentioned. To keep the temporal evolution of the observed phenomenon unaltered, the data set is not shuffled before the split in  $K$  subsets. In this way, the subset used for the test represents a segment of the data set containing a fraction of the samples (those not used for training) in their original temporal sequence. It is possible that the subset used for the test contains a rare event, not present in the subsets used for training, making prediction particularly difficult. Shuffling the data set avoids these criticalities, but it is not applied here to better simulate a real-case scenario, where the model is called to predict a set of consecutive samples.

The values of the four metrics obtained for the LSTM network and RF regressor when  $K$  is set equal to 10 are entered in Table II, including the average and standard deviation for each metric. The scatter plots of the predicted values are shown in Fig. 11, where all samples of the data set are present (i.e., including the predictions made on each subset during the  $K$  training and test cycles). Finally, in analogy with Fig. 10, the analysis of RMSE, CC, and ME as a function of the wind speed interval, considering the entire data set, is shown in Fig. 12.

The results of the  $K$ -fold cross-validation assessment, reported in Table II, make some observations possible.

- 1) LSTM prediction outperforms RF prediction in terms of RMSE, MAE, and CC, observing both average and standard deviation.
- 2) The predictions show stable performance over the different subsets (i.e., folds) as the standard deviations of RMSE, MAE, and CC are small compared to the respective average. The ME is an exception: although the average is less than 0.04 m/s, the standard deviation is larger, revealing that the individual folds are affected by some bias. In any case, the bias is limited as the standard deviation is less than 0.17 m/s for the RF prediction and less than 0.14 m/s for the RMSE prediction. In this case, LSTM is also advantageous compared to RF.
- 3) For both LSTM and RF, the performance obtained in the  $K$ -fold cross-validation case is slightly worse than the respective performance obtained in the chronological order case. Only CC shows a minor improvement.
- 4) The results obtained in [19] by using the RF regressor and applying the  $K$ -fold cross-validation on the same data, but

<sup>2</sup>As noted in [19], this problem is probably due to the small number of samples in the dataset for which the wind speed exceeds 12 m/s.

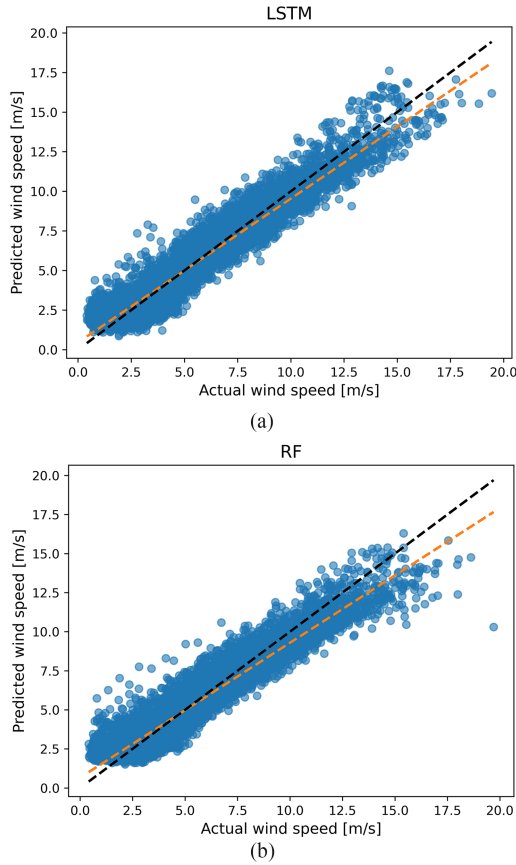


Fig. 11. Scatter diagrams of the wind predictions generated by (a) LSTM network and (b) RF regressor in the  $K$ -fold cross-validation assessment. The lines that best fit the predictions are inserted (golden lines) in comparison with quadrant bisectors (black lines).

after the data set shuffle operation, show better accuracy: RMSE was 0.81 m/s and MAE was 0.62 m/s. This confirms what was mentioned above: the shuffle eliminates the criticalities inherent in following the temporal evolution of the phenomenon, performing the prediction on temporal segments of which no sample was used in the training phase. Although it achieves lower accuracy, the test without shuffle (the results of which are reported in Table II) is more consistent with the real deployment of the prediction system at sea.

The analyses in Figs. 11 and 12 confirm that the introduction of the temporal correlation significantly reduces the underprediction problem above 12 m/s [19], both in terms of bias and accuracy. When, through the  $K$ -fold cross-validation strategy, all the data set samples are considered, the LSTM network keeps the ME above  $-0.6$  m/s and the RMSE below 1.65 m/s, also in the interval 14 to 16 m/s.

**B. Rainfall Intensity Prediction**

As for wind speed prediction, the LSTM network hyperparameters have been optimized following a grid search strategy and taking into consideration the chronological order scenario. The values considered are as follows:

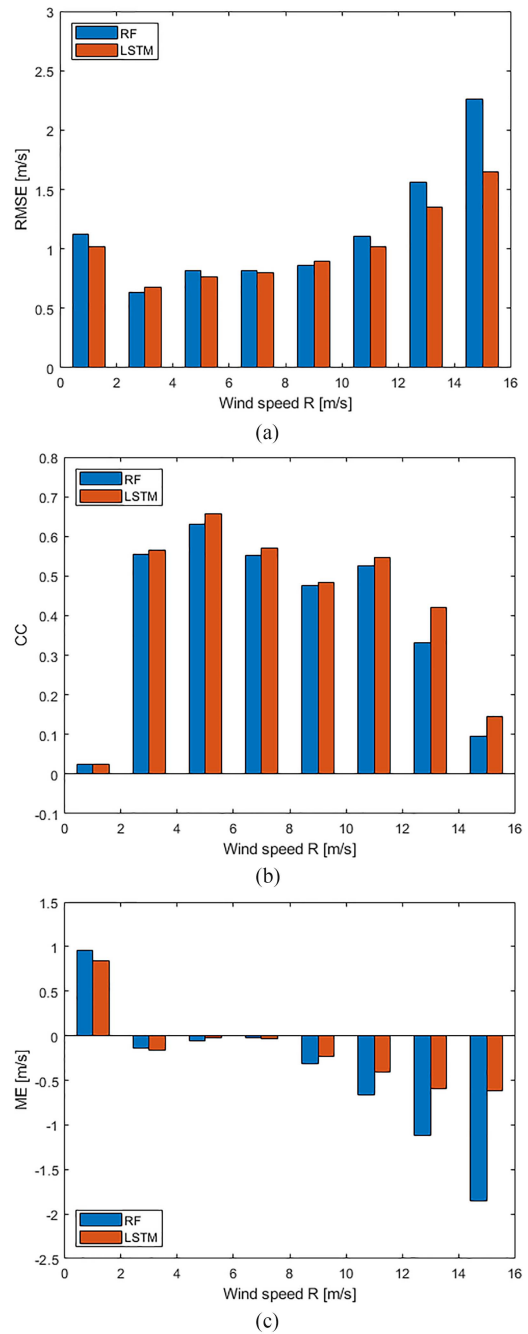


Fig. 12. Performance metrics for the  $K$ -fold cross-validation assessment, computed considering the samples whose actual wind speed is in  $[R, R+2]$ ,  $R = 0, 2, 4, \dots$  m/s. (a) RMSE. (b) CC. (c) ME.

- 1) the length of the input sequence,  $L$ , selected among 3, 6, and 9;
- 2) the batch size, selected between 1 and 2;
- 3) the number of LSTM units,  $N_h$ , between 32 and 64;
- 4) the number of dense units,  $N_d$ , among 16, 32, and 64.

It is worth noting that the batch size changes consistently from the wind prediction case. On the one hand, larger batch size is generally able to afford better regularization properties. On the other hand, since precipitation events are rare (587 rainy samples out of the 11 509 hourly samples), rainy samples negligibly

TABLE III  
LSTM NETWORK CONFIGURATIONS THAT PROVIDED THE LOWEST RMSE VALUES WHEN THE RAINFALL PREDICTION IS ASSESSED IN CHRONOLOGICAL ORDER

| Sequence length, $L$ | Batch size | N. of LSTM units, $N_h$ | N. of dense units, $N_d$ | RMSE [mm/h] | CC    | MAE [mm/h] | ME [mm/h] |
|----------------------|------------|-------------------------|--------------------------|-------------|-------|------------|-----------|
| 6                    | 2          | 32                      | 64                       | 0.154       | 0.957 | 0.033      | 0.011     |
| 6                    | 1          | 64                      | 64                       | 0.162       | 0.954 | 0.055      | 0.038     |
| 6                    | 1          | 64                      | 16                       | 0.174       | 0.946 | 0.060      | 0.038     |
| 3                    | 1          | 32                      | 16                       | 0.178       | 0.948 | 0.072      | 0.055     |
| 6                    | 2          | 64                      | 16                       | 0.182       | 0.943 | 0.088      | 0.067     |

The corresponding CC, MAE, and ME values are also reported.

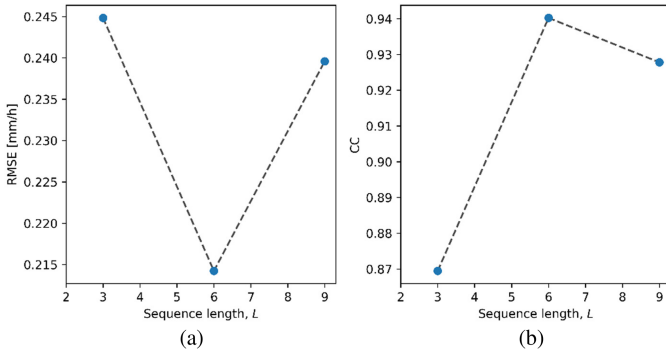


Fig. 13. (a) RMSE and (b) CC for the rainfall prediction, as a function of the extent of the temporal correlation considered, i.e., the sequence length  $L$ . The values of RMSE and CC are averaged over all values of the other hyperparameters.

influence the update of the network weights during the training process if a large batch size is used. Conversely, a small batch size ensures that the rainy samples, when processed, give a significant contribution to the update of the weights. Concerning  $N_h$  and  $N_d$ , as for the wind speed prediction case, the grid search boundaries have been chosen to study the tradeoff between the complexity of the model and the resulting generalization capabilities. In this case, again, values outside the selected ranges gave a poor performance, due to high bias or overfitting.

Table III reports the performance of the combinations of the hyperparameters that provide the 5 lowest RMSE values: the configuration providing the best accuracy is given by a sequence length  $L = 6$ , a batch size of 2, several LSTM units  $N_h = 32$ , and a set of  $N_d = 64$  dense units. Setting  $L = 6$  means performing the instantaneous prediction using the noise samples recorded from 50 min earlier to the current time. The validity of the choice is confirmed in Fig. 13, in which the RMSE and CC as a function of  $L$  are shown, averaged over all values of the other hyperparameters (i.e., averaging over all possible models, once the length of the sequence  $L$  is fixed). This result is consistent with the conclusions of [36], where, working with 15-min-interval rainfall measurements, the best forecast using data collected at a single spatial point is obtained using the last three measurements (i.e., a 45-min interval), and [34], where, working with 10-min-interval rainfall measurements, the best forecast is obtained using data collected in the last 60 min.

Concerning the hyperparameters of the RF model, as for the wind prediction, 50 trees are used and 32 features (out of the 64 spectral bins) are randomly selected at each node. The authors verified that this choice provides the lowest RMSE in the prediction of the rainfall intensity following the chronological order scenario. However, a change in these values, even over wide intervals, does not significantly reduce performance.

When the chronological order scenario is applied, the performance achieved by the LSTM network and RF model, configured as described above, are summarized in Table IV. The metrics are calculated on the whole test set and on two subsets of it: the samples where the actual rainfall intensity is greater than zero (rainy samples) and those where the actual rainfall is zero (nonrainy samples). The number of rainy samples is equal to 70, whereas the number of nonrainy samples is equal to 1930. Since for nonrainy samples the actual rainfall intensity is zero, according to (14), the CC is indeterminate and the corresponding boxes in Table IV are left empty. With the sole exception of the ME for rainy samples, the values of the four metrics for the LSTM network are significantly better than those obtained from the RF regressor. Fig. 14 provides visual confirmation of this fact, showing the scatter plots for the two prediction models, along with the lines that best fit the predictions (i.e., the regression lines) and the quadrant bisectors. It is worth noting that the LSTM network provides precipitation predictions with an RMSE of less than 0.77 mm/h when rainfall is actually present and less than 0.06 mm/h in the absence of rainfall.

The bias in the LSTM predictions for rainy samples (ME =  $-0.276$  mm/h), which is more significant than that reported by the RF regressor (ME =  $-0.187$  mm/h), is mainly caused by some rainy samples, with an intensity below 1.6 mm/h, for which the network predicts an intensity close to zero, as visible in Fig. 14. This bias does not prevent the accuracy of the predictions provided by the LSTM network for rainy samples from being significantly better than that of the RF regressor, both in terms of RMSE and MAE. The CC of the LSTM network is also considerably higher.

In [15], a portion of the data set used in this article and an RF regressor were employed to predict rainfall intensity, adopting spectral compounding instead of the prediction compounding adopted here (see Fig. 5). Rainy samples were split between the training set and the test set, resulting in an RMSE of 1.514 mm/h.

TABLE IV  
RAINFALL PREDICTION METRICS FOR THE LSTM NETWORK AND RF REGRESSOR, ASSESSED IN CHRONOLOGICAL ORDER

|                          | RMSE [mm/h] | CC    | MAE [mm/h] | ME [mm/h] |
|--------------------------|-------------|-------|------------|-----------|
| LSTM (all test samples)  | 0.154       | 0.957 | 0.033      | 0.011     |
| RF (all test samples)    | 0.259       | 0.867 | 0.048      | 0.025     |
| LSTM (rainy samples)     | 0.768       | 0.965 | 0.340      | -0.276    |
| RF (rainy samples)       | 1.210       | 0.908 | 0.493      | -0.187    |
| LSTM (non-rainy samples) | 0.059       | -     | 0.020      | 0.020     |
| RF (non-rainy samples)   | 0.130       | -     | 0.033      | 0.033     |

The metrics computed considering only rainy or only nonrainy samples are included.

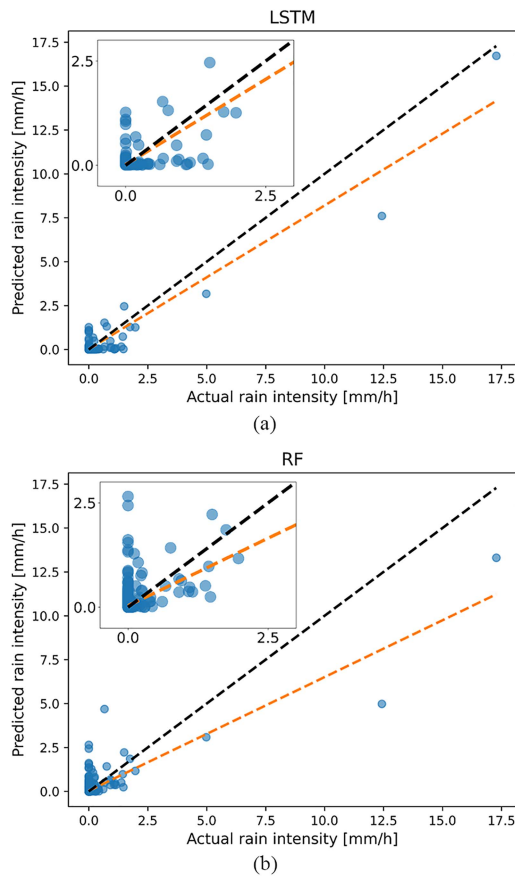


Fig. 14. Scatter diagrams of the rainfall predictions generated by (a) LSTM network and (b) RF regressor in the chronological order assessment. The lines that best fit the predictions are inserted (golden lines) in comparison with quadrant bisectors (black lines). A zoom of the region near the origin of the axes is inserted.

According to Table IV, the RMSE of the RF regressor developed in this article, considering only rainy samples, is 1.210 mm/h: the improvement over [15] is probably due to the advantages of prediction compounding, as demonstrated in [19] for wind prediction. In this perspective, the RMSE of 0.768 mm/h obtained through the LSTM network on the same samples assumes further significance.

To test the two prediction models on several rainy samples larger than the 70 samples present in the last 2000 h and in all

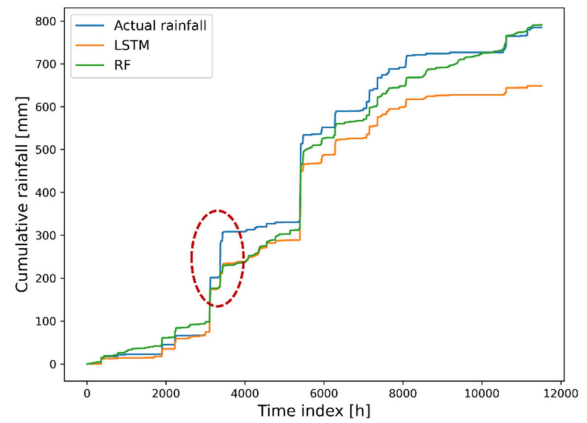


Fig. 15. Cumulative rainfall computed with actual measurements and predictions by the LSTM network and RF regressor. The red ellipse shows a time when the two prediction models fail to follow the actual rainfall accumulation.

seasons of the year, a  $K$ -fold cross-validation strategy, with  $K = 10$ , was adopted. For rainfall prediction, as in the case of wind, no shuffle is applied to the data set before splitting it into  $K$  subsets. Overall, 587 rainy samples out of 11 509 samples are available in the data set. The values of the four metrics for the LSTM network and RF regressor are entered in Table V, including the average and standard deviation. Since some of the ten subsets have almost no rainy samples, the CC for these subsets is meaningless and disturbs the average and standard deviation calculation. For this reason, only one CC value, calculated over the whole data set, was included in Table V.

A comparison between Tables IV and V reveals that by moving from the chronological order to the  $K$ -fold cross-validation assessment, the results deteriorate significantly, particularly those of the LSTM network. The inspection of the cumulative rainfall profiles (a quantity of special interest in climate science studies) allows one to identify the main problem. Fig. 15 shows these profiles for the entire data set, computed by setting equal to zero all the rainfall values below 0.1 mm/h (i.e., the output resolution of the deployed rain gauge). The two prediction models work properly (especially the LSTM network) until the hourly sample 3380, where the actual cumulative rainfall increases at great speed. Both the prediction models fail to follow the gradient. Subsequently, they resume following the actual trend

TABLE V  
RAINFALL PREDICTION METRICS FOR THE LSTM NETWORK AND RF REGRESSOR, ASSESSED BY *K*-FOLD CROSS-VALIDATION.

|                         | RMSE [mm/h]       | CC    | MAE [mm/h]        | ME [mm/h]         |
|-------------------------|-------------------|-------|-------------------|-------------------|
| LSTM (all test samples) | $0.264 \pm 0.179$ | 0.907 | $0.072 \pm 0.037$ | $0.020 \pm 0.025$ |
| RF (all test samples)   | $0.302 \pm 0.195$ | 0.887 | $0.061 \pm 0.032$ | $0.010 \pm 0.032$ |

For RMSE, MAE and ME the average  $\pm$  the standard deviation is reported.

TABLE VI  
RAINFALL PREDICTION METRICS FOR THE LSTM NETWORK AND RF REGRESSOR, ASSESSED BY *K*-FOLD CROSS-VALIDATION (REMOVING TWO FLOODING DAYS FROM THE DATA SET)

|                          | RMSE [mm/h]       | CC                | MAE [mm/h]        | ME [mm/h]          |
|--------------------------|-------------------|-------------------|-------------------|--------------------|
| LSTM (all test samples)  | $0.223 \pm 0.114$ | 0.929             | $0.070 \pm 0.024$ | $0.032 \pm 0.020$  |
| RF (all test samples)    | $0.295 \pm 0.163$ | 0.882             | $0.057 \pm 0.024$ | $0.012 \pm 0.022$  |
| LSTM (rainy samples)     | $1.349 \pm 0.304$ | $0.922 \pm 0.050$ | $0.777 \pm 0.173$ | $-0.426 \pm 0.207$ |
| RF (rainy samples)       | $1.776 \pm 0.532$ | $0.885 \pm 0.091$ | $0.870 \pm 0.234$ | $-0.607 \pm 0.259$ |
| LSTM (non-rainy samples) | $0.080 \pm 0.027$ | -                 | $0.047 \pm 0.012$ | $0.047 \pm 0.012$  |
| RF (non-rainy samples)   | $0.140 \pm 0.039$ | -                 | $0.031 \pm 0.006$ | $0.031 \pm 0.006$  |

The metrics computed considering only rainy or only nonrainy samples are included. Where applicable, the average  $\pm$  the standard deviation is reported.

(the LSTM network is particularly accurate in this respect) but the gap that occurred at the mentioned time remains. This heavy rain event corresponds to a particularly severe flood that struck the city of Genoa (on the Ligurian coast, approximately 80 km away from the buoy) on November 4th and 5th, 2011 causing damage and loss of lives. When the related acoustic samples are in the fold used for the test phase, both LSTM and RF are unable to predict such heavy rainfall because no similar event is present in the remaining folds (i.e., those used as the training set).

Due to the exceptional nature of the flooding, the tenfold cross-validation assessment is repeated using a data set from which all samples collected on the two days mentioned are removed. The values of the four metrics for the LSTM network and RF regressor are entered in Table VI, including the average and standard deviation, where applicable. The scatter plots of the predicted values are shown in Fig. 16, where all samples of the data set are comprised. Finally, Fig. 17 shows the cumulative rainfall profiles, computed by setting equal to zero all the rainfall values below 0.1 mm/h (i.e., the output resolution of the rain gauge).

The results of the *K*-fold cross-validation assessment, reported in Table VI, make some observations possible.

- 1) LSTM prediction outperforms RF prediction in terms of RMSE and CC, observing both average and standard deviation. For the rainy samples, this statement also holds in terms of MAE and ME.
- 2) Although the standard deviations of RMSE, MAE, and CC are lower than the respective averages, their magnitude is significant, greater than what was reported in wind prediction. This means that the prediction performance shows considerable fluctuations over the different folds

that may also comprise different types of precipitation (i.e., drizzle, stratiform, or convective).

- 3) For nonrainy samples, the RF regressor provides a lower overprediction than the LSTM network. However, the ME and MAE values for both prediction models are well below 0.1 mm/h, which is the output resolution of the deployed rain gauge. The advantage of RF is therefore of little importance. As the nonrainy samples are about 95% of all samples, the lower overprediction of the RF regressor is reflected on MAE and ME referred to all test samples: their values for the RF regressor are lower than those for the LSTM network.
- 4) For both LSTM and RF, the performance obtained in the *K*-fold cross-validation case is worse than the respective performance obtained in the chronological order case. The worsening is more significant for rainy samples and for the LSTM network.

The analyses in Figs. 16 and 17 confirm that the introduction of the temporal correlation significantly reduces the underprediction that characterizes the RF regressor for rainfall samples with an intensity higher than 5 mm/h. Indeed, the line that best fits the LSTM predictions is much closer to the quadrant bisector than that for the RF predictions. Furthermore, over about 16 months of rainfall observation, the cumulative precipitation profile computed using the LSTM predictions closely follows, with small discrepancies, the shape, and magnitude of that computed from actual data. The maximum discrepancy between the two profiles never exceeds 25 mm, despite the accumulated precipitation amounting to around 700 mm at the end of the period. In the case of the RF prediction, instead, the profile is smoother, rising where it should remain constant

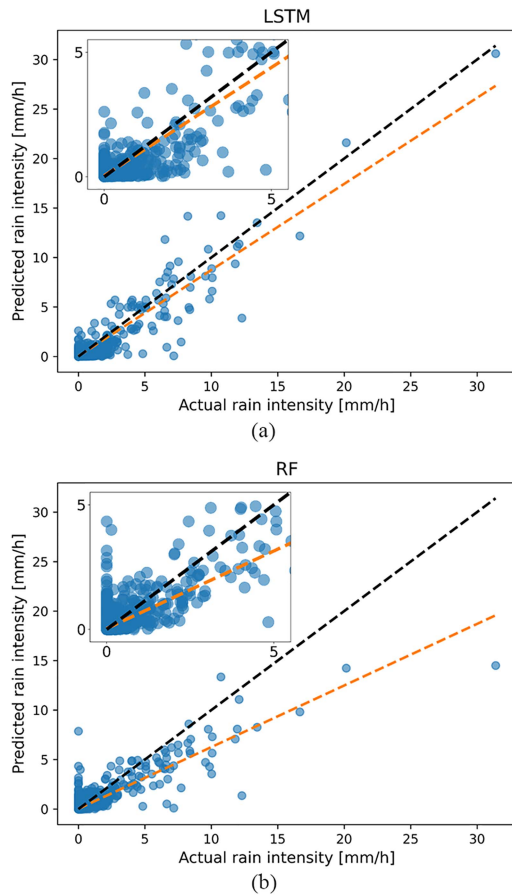


Fig. 16. Scatter diagrams of the rainfall predictions generated by (a) LSTM network and (b) RF regressor in the  $K$ -fold cross-validation assessment (removing two flooding days from the data set). The lines that best fit the predictions are inserted (golden lines) in comparison with quadrant bisectors (black lines). A zoom of the region near the origin of the axes is inserted.

and underestimating heavy rain events, in this way introducing greater discrepancies. Nevertheless, the ME value reported in Table VI for the “all test samples” case is very close to zero: this allows RF to provide a cumulative prediction that does not significantly diverge from the actual profile.

#### IV. CONCLUSION

This work demonstrates the value of considering temporal correlation in wind and rainfall prediction from underwater acoustic noise measurements. While previous articles have shown that ML techniques can improve the accuracy of memoryless prediction of wind and rainfall from underwater noise, compared to that obtained with empirical equations,<sup>3</sup> here it has been shown that the LSTM architecture is a supervised ML technique able to successfully model the temporal correlation which is inherent in the meteorological phenomena mentioned. The study focused on predicting hourly average wind speed and rainfall intensity from data recorded every 10 min over a period of about 16 months. The acoustic data were used to generate predictions,

<sup>3</sup>It is worth recalling that improvement is achieved through richer input information and at the cost of greater complexity.

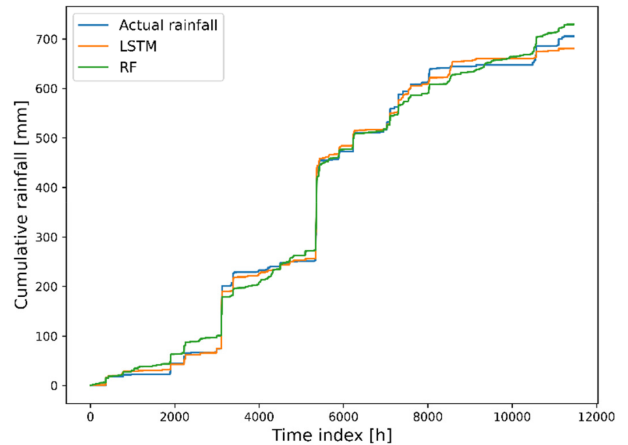


Fig. 17. Cumulative rainfall computed with actual measurements and predictions by the LSTM network and RF regressor (removing two flooding days from the data set).

with and without temporal memory, computing the hourly average for the wind speed and hourly cumulative for the rain intensity by compounding between all the instantaneous predictions made within 1 h. The predictions were compared with wind speed and rainfall intensity data measured above the sea surface, at the same location as the acoustic recording. While the LSTM network was used to include temporal correlation, memoryless predictions were performed by means of an RF regressor, which was particularly appreciated in the previous literature.

In wind prediction, the introduction of the two past spectra in addition to the current one (i.e., going back 20 min) resulted in better performance and, in particular, significantly reduced the underprediction that occurs with RF regression when wind speeds exceed 12 m/s. The assessment of ML models through tenfold cross-validation without data set shuffle<sup>4</sup> showed that LSTM networks predict wind speed with RMSE of 0.843 m/s, MAE of 0.665 m/s, and CC higher than 0.95. In the speed interval from 14 to 16 m/s, the RMSE is kept below 1.65 m/s. This performance was achieved without eliminating any acoustic samples: it is, therefore, robust against other sources of underwater noise, such as passing ships, and does not insert any time gaps in the prediction sequence.

Concerning rainfall prediction, the introduction of the five past spectra in addition to the current one (i.e., going back 50 min) resulted in performance significantly better than that achieved by the RF regression. After removing the samples collected during two days when a flood hit the city of Genoa (about 80 km away from the sensors) from the data set, the LSTM architecture, assessed through tenfold cross-validation without shuffle, provided rainfall intensity predictions with RMSE of 0.223 mm/h, MAE of 0.070 mm/h, and CC of about 0.93. Considering only the samples where rain is actually present, the metrics are RMSE = 1.349 mm/h, MAE = 0.777 mm/h, and CC = 0.922. Even for precipitation, acoustic samples affected by other noise sources were not discarded.

<sup>4</sup>Each fold used to test the model is a particular segment of the dataset, which preserves its original temporal order, and which was not used to train the model. Finally, the model is tested using, fold by fold, the entire dataset.

The use of temporal correlation not only improves the prediction accuracy, it also reduces the fluctuation in performance across the different folds that compose the data set. However, while a training set with a duration of about 14.4 months (i.e., nine-tenths of the entire data set) ensures considerable stability of performance in the case of wind, the same cannot be said for rainfall. As the latter is a phenomenon observed in only about 5% of the available samples, it is difficult to train the regressor for all possible rainy events. For example, despite training sets of considerable duration, the regressor was not prepared for the flood that occurred on the coast during the observed period. To incorporate rainfall-related temporal memory and achieve more stable performance, training sets of even longer duration would probably be necessary.

Therefore, while wind prediction using LSTM networks can be considered an advantageous option for the design of devices operating at sea, in the case of rainfall, the encouraging results obtained so far do not exclude the need for further investigation. In addition, the dependence of the results on the site where the acoustic system is installed remains to be verified. If the underwater acoustic propagation or meteorological conditions that characterize the site change significantly, it may be necessary to re-train the system and the prediction accuracy may vary. Future research could address these aspects. In particular, domain adaptation is an ML technique [46] that could allow the regression model, trained using a large amount of data, to be retained and adapted to a new installation site thanks to a limited number of samples acquired at the new site. The combined use of samples acquired at different installation sites could alleviate the need for registrations lasting many months to properly train the system.

## REFERENCES

- [1] F. J. Wentz et al., "Evaluating and extending the ocean wind climate data record," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 10, no. 5, pp. 2165–2185, May 2017.
- [2] A. Y. Hou et al., "The global precipitation measurement mission," *Bull. Amer. Meteorol. Soc.*, vol. 95, no. 5, pp. 701–722, 2014.
- [3] L. Zedel, L. Gordon, and S. Osterhus, "Ocean ambient sound instrument system: Acoustic estimation of wind speed and direction from a subsurface package," *J. Atmos. Ocean. Technol.*, vol. 16, no. 8, pp. 1118–1126, 1999.
- [4] J. A. Nystuen, S. E. Moore, and P. J. Staben, "A sound budget for the southeastern Bering Sea: Measuring wind, rainfall, shipping, and other sources of underwater sound," *J. Acoust. Soc. Amer.*, vol. 128, no. 1, pp. 58–65, 2010.
- [5] J. Yang, S. C. Riser, J. A. Nystuen, W. E. Asher, and A. T. Jessup, "Regional rainfall measurements: Using the passive aquatic listener during the SPURS field campaign," *Oceanography*, vol. 28, no. 1, pp. 124–133, 2015.
- [6] P. T. Shaw, D. R. Watts, and H. T. Rossby, "On the estimation of oceanic wind speed and stress from ambient noise measurements," *Deep Sea Res.*, vol. 25, no. 12, pp. 1225–1233, 1978.
- [7] D. L. Evans, D. R. Watts, D. Halpern, and S. Bourassa, "Oceanic winds measured from the seafloor," *J. Geophys. Res., Ocean.*, vol. 89, no. C3, pp. 3457–3461, 1984.
- [8] S. Vagle, W. G. Large, and D. M. Farmer, "An evaluation of the WOTAN technique of inferring oceanic winds from underwater ambient sound," *J. Atmos. Ocean. Technol.*, vol. 7, no. 4, pp. 576–595, 1990.
- [9] J. A. Nystuen and H. D. Selsor, "Weather classification using passive acoustic drifters," *J. Atmos. Ocean. Technol.*, vol. 14, no. 3, pp. 656–666, 1997.
- [10] B. B. Ma and J. A. Nystuen, "Passive acoustic detection and measurement of rainfall at sea," *J. Atmos. Ocean. Technol.*, vol. 22, no. 8, pp. 1225–1248, 2005.
- [11] S. Pensieri, R. Bozzano, J. A. Nystuen, E. N. Anagnostou, M. N. Anagnostou, and R. Bechini, "Underwater acoustic measurements to estimate wind and rainfall in the Mediterranean Sea," *Adv. Meteorol.*, vol. 2015, Art. no. 612512.
- [12] J. A. Nystuen, M. N. Anagnostou, E. N. Anagnostou, and A. Papadopoulos, "Monitoring Greek seas using passive underwater acoustics," *J. Atmos. Ocean. Technol.*, vol. 32, no. 2, pp. 334–349, 2015.
- [13] D. Cazau, J. Bonnel, and M. Baumgartner, "Wind speed estimation using acoustic underwater glider in a near-shore marine environment," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 4, pp. 2097–2106, Apr. 2019.
- [14] D. D. Lemon, D. M. Farmer, and D. R. Watts, "Acoustic measurements of wind speed and precipitation over a continental shelf," *J. Geophys. Res., Ocean.*, vol. 89, no. C3, pp. 3462–3472, 1984.
- [15] W. O. Taylor, M. N. Anagnostou, D. Cerrai, and E. N. Anagnostou, "Machine learning methods to approximate rainfall and wind from acoustic underwater measurements," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 4, pp. 2810–2821, Apr. 2021.
- [16] L. Breiman, "Random forests," *Mach. Learn.*, vol. 45, no. 1, pp. 5–32, 2001.
- [17] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Berlin, Germany: Springer-Verlag, 2009.
- [18] A. Trucco, R. Bozzano, E. Fava, S. Pensieri, A. Verri, and A. Barla, "A supervised learning approach for rainfall detection from underwater noise analysis," *IEEE J. Ocean. Eng.*, vol. 47, no. 1, pp. 213–225, Jan. 2022.
- [19] A. Trucco et al., "Compounding approaches for wind prediction from underwater noise by supervised learning," *IEEE J. Ocean. Eng.*, vol. 47, no. 4, pp. 1172–1187, Oct. 2022, doi: [10.1109/JOE.2022.3162689](https://doi.org/10.1109/JOE.2022.3162689).
- [20] A. C. Brett and S. E. Tuller, "The autocorrelation of hourly wind speed observations," *J. Appl. Meteorol. Climatol.*, vol. 30, no. 6, pp. 823–833, 1991.
- [21] Z. Huang and Z. S. Chalabi, "Use of time-series analysis to model and forecast wind speed," *J. Wind Eng. Ind. Aerodyn.*, vol. 56, no. 2/3, pp. 311–322, 1995.
- [22] A. Sfetsos, "A comparison of various forecasting techniques applied to mean hourly wind speed time series," *Renew. Energy*, vol. 21, no. 1, pp. 23–35, 2000.
- [23] A. Sfetsos, "A novel approach for the forecasting of mean hourly wind speed time series," *Renew. Energy*, vol. 27, no. 2, pp. 163–174, 2002.
- [24] J. Shi, J. Guo, and S. Zheng, "Evaluation of hybrid forecasting approaches for wind speed and power generation time series," *Renew. Sustain. Energy Rev.*, vol. 16, no. 5, pp. 3471–3480, 2012.
- [25] G. Memarzadeh and F. Keynia, "A new short-term wind speed forecasting method based on fine-tuned LSTM neural network and optimal input sets," *Energy Convers. Manage.*, vol. 213, 2020, Art. no. 112824.
- [26] A. Saeed et al., "Hybrid bidirectional LSTM model for short-term wind speed interval prediction," *IEEE Access*, vol. 8, pp. 182283–182294, 2020.
- [27] Q. Zhu et al., "Learning temporal and spatial correlations jointly: A unified framework for wind speed prediction," *IEEE Trans. Sustain. Energy*, vol. 11, no. 1, pp. 509–523, Jan. 2020.
- [28] E. Hernández, V. Sanchez-Anguix, V. Julian, J. Palanca, and N. Duque, "Rainfall prediction: A deep learning approach," in *Proc. Int. Conf. Hybrid Artif. Intell. Syst.*, 2016, pp. 151–162.
- [29] F. Kratzert, D. Klotz, C. Brenner, K. Schulz, and M. Herrnegger, "Rainfall-runoff modelling using long short-term memory (LSTM) networks," *Hydrol. Earth Syst. Sci.*, vol. 22, no. 11, pp. 6005–6022, 2018.
- [30] S. Poornima and M. Pushpalatha, "Prediction of rainfall using intensified LSTM based recurrent neural network with weighted linear units," *Atmosphere*, vol. 10, no. 11, 2019, Art. no. 668.
- [31] E. Toth, A. Brath, and A. Montanari, "Comparison of short-term rainfall prediction models for real-time flood forecasting," *J. Hydrol.*, vol. 239, no. 1–4, pp. 132–147, 2000.
- [32] P. C. Nayak, K. P. Sudheer, D. M. Rangan, and K. S. Ramasastri, "Short-term flood forecasting with a neurofuzzy model," *Water Resour. Res.*, vol. 41, no. 4, 2005, Art. no. W04004.
- [33] M. Qiu et al., "A short-term rainfall prediction model using multi-task convolutional neural networks," in *Proc. IEEE Int. Conf. Data Mining*, 2017, pp. 395–404.
- [34] D. H. Nguyen, J. B. Kim, and D. H. Bae, "Improving radar-based rainfall forecasts by long short-term memory network in urban basins," *Water*, vol. 13, 2021, Art. no. 776.
- [35] A. Y. Barrera-Animas, L. O. Oyedele, M. Bilal, T. D. Akinosho, J. M. D. Delgado, and L. A. Akanbi, "Rainfall prediction: A comparative analysis of modern machine learning algorithms for time-series forecasting," *Mach. Learn. Appl.*, vol. 7, 2022, Art. no. 100204.

- [36] K. C. Luk, J. E. Ball, and A. Sharma, "A study of optimal model lag and spatial inputs to artificial neural network for rainfall forecasting," *J. Hydrol.*, vol. 227, no. 1–4, pp. 56–65, 2000.
- [37] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.
- [38] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, MA, USA: MIT Press, 2016.
- [39] R. Bozzano et al., "The M3A network of open ocean observatories in the Mediterranean Sea," in *Proc. IEEE/MTS OCEANS Conf.*, Bergen, Norway, 2013, pp. 1–10.
- [40] E. Canepa, S. Pensieri, R. Bozzano, M. Faimali, P. Traverso, and L. Cavaleri, "The ODAS Italia 1 buoy: More than forty years of activity in the Ligurian Sea," *Prog. Oceanogr.*, vol. 135, pp. 48–63, 2015.
- [41] S. Pensieri, R. Bozzano, M. N. Anagnostou, E. N. Anagnostou, R. Bichini, and J. A. Nystuen, "Monitoring the oceanic environment through passive underwater acoustics," in *Proc. IEEE/MTS*, Bergen, Norway, 2013, pp. 1–10.
- [42] M. N. Anagnostou, J. A. Nystuen, E. N. Anagnostou, A. Papadopoulos, and V. Lykousis, "Passive aquatic listener (PAL): An adoptive underwater acoustic recording system for the marine environment," *Nucl. Instrum. Methods Phys. Res. A Accel. Spectrom. Detect. Assoc. Equip.*, vol. 626, pp. S94–S98, 2011.
- [43] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," *Nature*, vol. 323, no. 6088, pp. 533–536, 1986.
- [44] Y. LeCun et al., "Backpropagation applied to handwritten zip code recognition," *Neural Comput.*, vol. 1, no. 4, pp. 541–551, 1986.
- [45] Y. Bengio, P. Simard, and P. Frasconi, "Learning long-term dependencies with gradient descent is difficult," *IEEE Trans. Neural Netw.*, vol. 5, no. 2, pp. 157–166, Mar. 1994.
- [46] H. Daume III and D. Marcu, "Domain adaptation for statistical classifier," *J. Artif. Intell. Res.*, vol. 26, pp. 101–126, 2006.



**Andrea Trucco** (Senior Member, IEEE) was born in Genoa, Italy, in 1970. He received the M.Sc. and Ph.D. degrees in electronic and computer engineering from the University of Genoa, Genoa, Italy, in 1994 and 1998, respectively.

Since 1999, he has been with the University of Genoa, where in 2016, he was appointed as a Professor in telecommunications. From 2008 to 2012, he was the Vice-Director of the Department of Biophysical and Electronic Engineering, from 2011 to 2015, he was a Senior Researcher at the Italian Institute of

Technology, and from 2015 to 2020, he was the Vice-Rector for International Relations. He was the Principal Investigator in many research projects, funded by the EU, national agencies, and industrial companies. His research interests include acoustics, signal processing, pattern recognition, sonar systems, medical ultrasound, and fault diagnosis.

Dr. Trucco is currently an Associate Editor for IEEE JOURNAL OF OCEANIC ENGINEERING and IEEE TRANSACTIONS ON ULTRASONICS, FERROELECTRICS, AND FREQUENCY CONTROL. He is the author of more than 200 scientific papers (95 in international journals) and the inventor of 7 international patents. He won the student paper competitions organized by the 9th International Symposium Unmanned Untethered Submersible Technology (1995) and the MTS/IEEE Oceans'97 International Conference (1997).



**Annalisa Barla** received the master's degree in physics and the Ph.D. degree in computer science (working on kernel functions engineering for regularization methods in machine learning applied to image content understanding) from the University of Genoa, Genoa, Italy, in 2001 and 2005, respectively.

She is currently an Associate Professor of computer science with the University of Genoa. She is the author of more than 60 peer-reviewed journals and conference papers. Her main areas of interest are in the field of machine learning spanning from the study

of robust and reproducible variable selection methods with sparse regularization to the understanding and visualization of complex structured network data.



**Roberto Bozzano** received the M.Sc. degree in electronic engineering from the University of Genoa, Genoa, Italy, in 1993.

Since 1996, he has been working with the National Research Council of Italy. He is the principal investigator of the offshore observatory WIM3A, part of ERIC-EMSO and ERIC-ICOS. He is also an active member of regional and global alliances such as MonGOOS and OceanSITES. His theoretical and experimental research activity concerns the field of complex and multisensory technologies for environmental monitoring with particular reference to the marine-maritime sector. Specifically, the activity concerns the design of complex multifunctional measurement systems and their remote management, the development of acoustic methodologies for the measurement of environmental parameters, the study of the physical processes of energy exchanges at the interface-sea, and, in general, the development of systems for the acquisition and processing of signals on real-time embedded platforms and architectures.



**Sara Pensieri** received the M.Sc. degree in telecommunication engineering and the Ph.D. degree in electronic and computer engineering, robotics, and telecommunications from the University of Genoa, Genoa, Italy, in 2005 and 2013, respectively.

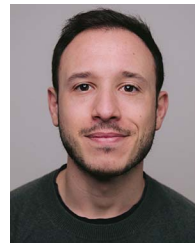
Since 2005, she has been a Researcher at the National Research Council of Italy in the Institute for the Study of Anthropogenic Impact and the Sustainability in the Marine Environment in Genoa. She is involved in the scientific and technological development of the WIM3A observatory. Her expertise spans from operational oceanography to underwater acoustics and the analysis of physical processes at the air-sea interface. She was involved in several European Projects (MERSEA, EuroSITES, MyOCEAN, PERSEUS, FixO<sup>3</sup>, and MINKE) aiming at the development of innovative methodologies for marine environmental monitoring from unmanned platforms. She has experience in data management and the design of field experiments at sea during oceanographic cruises.



**Alessandro Verri** was born in Genoa, Italy, in 1960. He received the Ph.D. degree in physics from the University of Genoa, Genoa, Italy, in 1989.

Since 2000, he has been a Professor of computer science with the University of Genoa. He visited MIT multiple times, ICSI, Heriot-Watt University, and INRIA-IRISA. He supervised more than 20 Ph.D. students and 120 M.Sc. students. He authored more than 150 papers in peer-reviewed journals and proceedings at international conferences. His research interests include learning theory and algorithms for

applications in the domain of computer vision and biomedical data analysis.



**David Solarna** (Member, IEEE) received the B.Sc. (*cum laude*), M.Sc. (*cum laude*), and Ph.D. degrees in telecommunication engineering from the University of Genoa, Genoa, Italy, in 2014, 2016, and 2021, respectively.

In 2016, he was a Visiting Researcher with the NASA Goddard Space Flight Center, Greenbelt, MD, USA. From 2017 to 2020, he was a Research Fellow with the NATO STO CMRE, La Spezia, Italy. His research interests include data fusion, with a particular focus on image registration techniques, machine learning and pattern recognition methods applied to remote sensing and underwater sonar data, and the combined use of modeling and simulation and autonomous systems in the underwater domain.

Dr. Solarna received the Interactive Symposium Best Paper Award of the 2016 IEEE International Geoscience and Remote Sensing Symposium, the award for the best Ph.D. thesis in the field of communication technologies organized by the GTTI group (Gruppo Telecomunicazioni e Tecnologie dell'Informazione) in 2021, and the award for the best Ph.D. thesis organized by the IEEE GRSS Chapters GRS29 Central-North Italy and GRS29 South Italy in 2022.