Encoder-Decoder Machine Learning Approach for Meteo-Oceanographic Timeseries Prediction

Authors: Gianina Chirosca[(1)], Maria Emanuela Mihailov[(2)*], Maria Ionela Tomescu Chivu[(3)], Alecsandru Chiroșca[(1)]

(1) Faculty of Physics - University of Bucharest, Atomistilor Street no.. 405, Magurele, Ilfov, Romania, gianina.chirosca@unibuc.ro

(2) Maritime Hydrographic Directorate, Fulgerului Street no.1, Constanta, Romania, emanuela.mihailov@dhmfn.ro

(3) National Meteorological Administration - Regional Meteorological Centre of Dobrogea, Mamaia Blvd. 300, Constanta, Romania, mariatmsc@yahoo.com

Abstract: Environmental state evaluation through its dynamic parameters plays a key role in assessment procedures to apply prediction models for early anomaly detection. This work presents the implementation of a Machine Learning Encoder-Decoder pattern used to environmental in-situ data (air and seawater temperature and wind speed) measured at Gloria offshore drilling platform - Romanian Black Sea Shelf. The station was chosen because there are no boundary interactions with the coastal region which enables the development of the multivariate, unidirectional time-series prediction algorithm. The model provided less than 5% mean absolute error (MAE) for 7 data points (months) forecast requiring the last 10 data points as input. The model accuracy enables the anomaly identification for meteo-oceanographic monthly average data. Future evolution of the seawater temperature extends the model for coastal areas with a less than 5% additional accuracy reduction. This model was developed mainly using available open-source frameworks permitting the integration with most of the visualisation platforms available today.

Keywords: Environmental health assessment, machine-learning prediction, environmental modelling, machine learning, time-series prediction

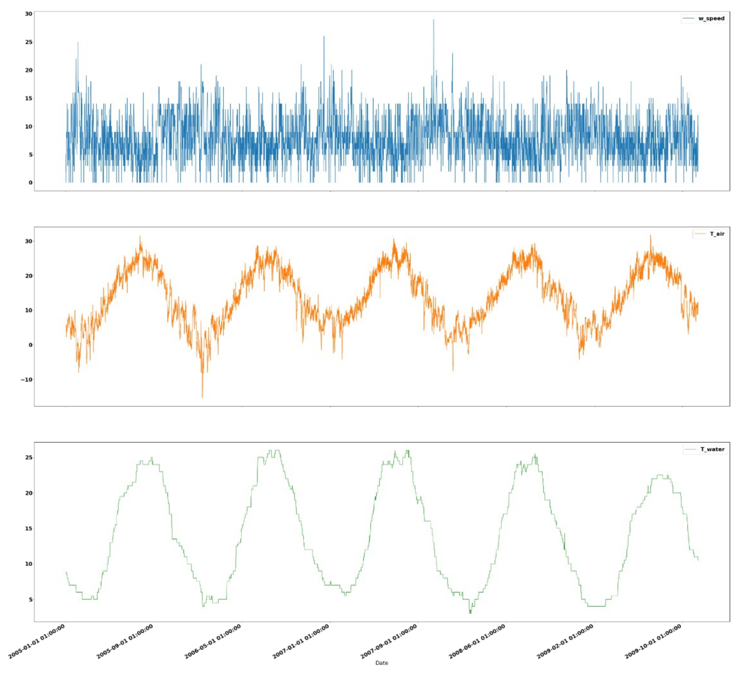

Fig. 1. Original Dataset for multivariate model training.

Data & Methods: The current study is based on in-situ observations (air and seawater temperature and wind speed), recorded at Gloria Platform (44.52°N, 29.57°E), 30km offshore Romanian Black Sea coast at 50m bathymetric line. The wind measurements are realised at the 36m above sea level. The available datasets cover five years of continuous measurements (2005 to 2010) with a recording interval at every 6 hours. Data was split into training and testing datasets (the training dataset contains the last 700 datapoints within the multivariate timeseries.

Our approach is based upon open source Data Science Tools applied for machine learning using the Python language for the model development, thus allowing for a better data evaluation and visualisation.

We start implementing the machine learning models starting with a multivariate pattern of 3 variables (wind speed, air temperature and water temperature) with a lookback of 12 values and a forward prediction of 3 values. We built several models and obtained the best results using a 4x4 model (The encoder has a depth of 4 LSTM layer, and the decoder is the same with four layers interconnected to the respective encoder layers). Using an iterative approach, we found that the overfitting for our models occurs above epoch 60 and chooses a total epoch size of 50 steps. The training parameters include 200 neurons for each LSTM layer.

```
Model: "model"
__________________________________________________________________________________________________
Layer (type)                    Output Shape         Param #     Connected to
==================================================================================================
input_1 (InputLayer)            [(None, 12, 3)]      0
__________________________________________________________________________________________________
lstm (LSTM)                     [(None, 12, 200), (N 163200      input_1[0][0]
__________________________________________________________________________________________________
lstm_3 (LSTM)                   [(None, 200), (None, 320800      lstm[0][0]
__________________________________________________________________________________________________
repeat_vector (RepeatVector)    (None, 3, 200)       0           lstm_3[0][0]
__________________________________________________________________________________________________
lstm_4 (LSTM)                   (None, 3, 200)       320800      repeat_vector[0][0]
                                                                 lstm[0][1]
                                                                 lstm[0][2]
__________________________________________________________________________________________________
lstm_1 (LSTM)                   [(None, 200), (None, 320800      lstm[0][0]
__________________________________________________________________________________________________
lstm_5 (LSTM)                   (None, 3, 200)       320800      lstm_4[0][0]
                                                                 lstm_1[0][1]
                                                                 lstm_1[0][2]
__________________________________________________________________________________________________
lstm_2 (LSTM)                   [(None, 200), (None, 320800      lstm[0][0]
__________________________________________________________________________________________________
lstm_6 (LSTM)                   (None, 3, 200)       320800      lstm_5[0][0]
                                                                 lstm_2[0][1]
                                                                 lstm_2[0][2]
__________________________________________________________________________________________________
lstm_7 (LSTM)                   (None, 3, 200)       320800      lstm_6[0][0]
                                                                 lstm_3[0][1]
                                                                 lstm_3[0][2]
__________________________________________________________________________________________________
time_distributed (TimeDistribut (None, 3, 3)         603         lstm_7[0][0]
==================================================================================================
Total params: 2,409,403
Trainable params: 2,409,403
Non-trainable params: 0
__________________________________________________________________________________________________
```

Fig. 2. Model implementation using Encoder-Decoder pattern with 4x4 LSTM model.

Fig. 3. Model estimation (red: recorded, blue: predicted) data for the testing dataset. The testing data is 10% from the actul data (Fig. 1).

Results & Discussion: Model training for the determined amount of epoch can be analysed using the loss function plot for both the training and testing dataset. Overfitting will increase testing loss (compared to the actual value in the specific dataset) while the training loss decreases.

After the model training, we evaluated the results using the remainder 10% of the data (the testing data).

Our first parameter, the wind speed, provides an estimation for the potential energy transferred to the water masses and usually, this parameter is considered an input parameter. The wind speed prediction appears to be accurate (within an absolute mean error - MAE of 2.09 m/s). An accurate prediction model with an MAE of 1.11°C was developed for the air temperature. The main output parameter, the water temperature, shows an MAE of 0.28°C for the testing dataset.

Conclusion: Our implementation is a proof of concepts for such models using wind speed, water and air temperature with no exogenous variables. The mean absolute error for the main parameter (the water temperature) provides an estimate for Machine Learning Recurrent Neural Network designs tuned for specific complex issues. For future data, not seen by the model we expect an increase in prediction accuracy within 10% leading to an accurate prediction model using Machine Learning techniques. The model can be extended for area prediction using another multivariate parameter encoding the station (position on the map). The authors recommend continuing to use the machine learning models and their derivatives.

Acknowledgements

This research has been carried out, with financial support, from the Sectorial Research-Development Plan of the Romanian Ministry of National Defense 2020 and 2021.