proceedings

Earthquakes magnitude prediction using recurrent neural networks

Jesús González[1], Wen Yu[2] and Luciano Telesca[3]

[1] Departamento de Control Automático CINVESTAV-IPN; contacto@jesusgonzalez.mx
[2] Departamento de Control Automático CINVESTAV-IPN; yuw@ctrl.cinvestav.mx
[3] Istituto di Metodologie per l'Analisi Ambientale, CNR; luciano.telesca@imaa.cnr.it

† Presented at the title, place, and date.

Received: date; Accepted: date; Published: date

Abstract: Seismological research importance around the globe is very clear, therefore new tools and algorithms are needed in order to predict magnitude, time and geographic location, as well as found out relationships that allow us to understand better this phenomenon and thus be able to save countless human lives. However, given the highly random nature of the earthquakes and the complexity in obtaining an efficient mathematical model, until now the efforts are insufficient and new methods capable of contributing to this challenge are needed. In this work a novel earthquakes magnitude prediction method is proposed, which is based on the composition of a known system whose behavior is governed according to the measurements of more than two decades of seismic events and is modeled as a time series using Machine Learning, specifically a network architecture based on LSTM cells.

Keywords: Earthquake Prediction; LSTM; Time series; Machine Learning;

1. Introduction

Natural disasters are without any doubt a latent danger and become very devastating and threaten the entire ecosystem of one region, that's why the prediction of earthquakes plays such an important role since its goal is to specify the magnitude and geographical and temporary location of future earthquakes with enough precision and anticipation to issue a warning. Despite the efforts to make mechanical or computational models of the earthquake process, these still do not achieve a real predictive power. Given the highly random nature of the earthquakes with relative high magnitude, their occurrence can only be analyzed with a statistical approach, but any synthetic model must show the same characteristics with respect to its distribution in size, time and space, which is very hard to achieve [1].

The earthquakes prediction can be separated into three main categories, short-term, intermediate-term, and long-term prediction, whose difference is in the type of analysis and the time considered to make the prediction. When we talk about short-term category the so-called precursors, which are phenomena or anomalies that precede the earthquake, are the main parameters used for making predictions. Tsuneji Rikitake [2], compiled almost 400 precursors that could give clues of a possible large magnitude earthquake.

The intermediate-term and long-term prediction look for trends or patterns in the seismic related signals recorded during periods that go from 1 to 10 years and from 10 years and above, respectively. There are different techniques for intermediate-term prediction, such as the CN algorithm, MSc algorithm and M8 algorithm, while for long-term predictions, despite the serious effort and the several developed models, no efficient technique has been established yet [3]. Thus, our work inherits in this context.

Nowadays, with the increasing computational power and the existing data processing tools, several techniques have been proposed such as that developed by Wang et al. [4], who used long short-term memory (LSTM) networks to learn the spatio-temporal relationship between earthquakes in different locations and make predictions on the base of such relationship. For the prediction of the magnitude of an earthquake in the region of Hindukush, Asim [5] used machine learning techniques, including pattern recognition neural network, recurrent neural network, random forest and linear programming boost classifier, formulating the problem as a binary classification task, and making prediction of earthquakes with magnitude greater than or equal to 5.5 in a time interval of 1 month. Narayanakumar1 and Raja [6] evaluated the performance of BP neural network techniques in predicting earthquakes occurring in the region of Himalayan belt with the use of different types of input data.

In the present work, we propose a short-term prediction of earthquake magnitude in Italy using a database of seismic events spanning over more than 20 years, by using a recurrent neural network model. The short-term earthquake prediction is very challenging, because large earthquakes cannot be reliably predicted for specific regions over time scales less than decades [7].

2. Methods

2.1. Time series modeling with the recurrent neural networks LSTM

Unlike traditional neural networks, the recurrent neural networks LSTM is an extremely efficient tool when the information is sequential. The basic condition of LSTM modeling is that all inputs and outputs are independent of each other. However, the dynamic non-linear system has the following form

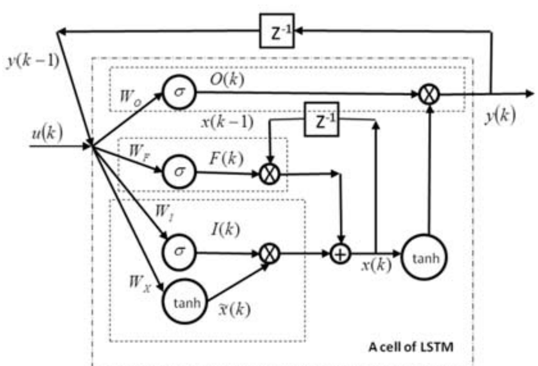
$$y(k) = \Phi[y(k-1), \cdots, y(k-n_y), u(k), \cdots, u(k-n_u)] \quad (1)$$

$\Phi(\cdot)$ is an unknown non-linear difference equation representing the plant dynamics, $u(k)$ and $y(k)$ are measurable scalar input and output, n_y and n_u are the last values of the output and input, respectively, to be considered for the system dynamics. Then the time series can be identified by the following prediction model:

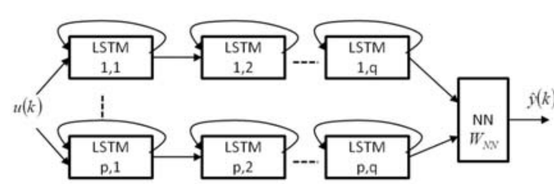
$$y(k) = N[y(k-1), \cdots, y(k-m)] \quad (2)$$

where m is the regression order for the output $y(k)$.

Connecting previous information to the present task depends of many factors, but the recurrent neural networks LSTM can learn to use the past information. In theory any recurrent neural network "RNN" can handle such long-term dependencies by picking certain parameters but in practice it does not seem to be able to learn them, but LSTM networks uses gates cells to remember them. The key gate to LSTMs is the cell state. LSTM cell has three gates to protect and control the cell state: forget gate, input gate, and output gate as shown in Figure 1(a).



(**a**)



(**b**)

Figure 1. LSTM cell (a), LSTM neural model(b).

The object of time series modeling using LSTM is to update the weights W_F, W_I, W_X, and W_O, such that the output of the LSTM neural network converges to the system output $y(k)$ in (1)

$$argmin_{W_F,W_I,W_X,W_O}[y(k) - y(k)]^2 \quad (11)$$

In this paper, we combine the classical neural networks with LSTM. This neural model [8] is shown in Figure 1 (b). Here we use $p \times q$ LSTMs, which are connected in simple feedforward form. The final p LSTMs are fully connected to a multilayer perceptron.

2.2. Earthquakes magnitude prediction as a time series modeling problem

In the present work we analyze the Italian seismic catalog of earthquakes with magnitude equal or larger than 1.5 from 1995 to 2018.

For each seismic event variables such as latitude, longitude and depth of the hypocenter, time of occurrence and magnitude are treated as a function $E(P(k))$ that represents a set of samples over time, where $k = 1, \ldots, N$ with N is the total number of samples taken in that period of time, and $P(k)$ is a vector of parameters derived from those variables.

The goal is to find a relationship between past and future events in order to predict the magnitude of upcoming events, with an acceptable error using the actual information. Given the seemingly random nature of the problem, it is difficult to find such relationship between the different variables and their derivatives; moreover, it is also difficult to know to what extent they influence the magnitude of a future event.

Therefore the following basic model to be learned is proposed:

$$y(k) = N[y(k-1), \cdots, y(k-n)] \quad (12)$$

where $y(k)$ represents the magnitude of an event at time k. Figure 2 represents the model shown in equation (12) for training (a) and prediction (b); such model will be used throughout the present work.

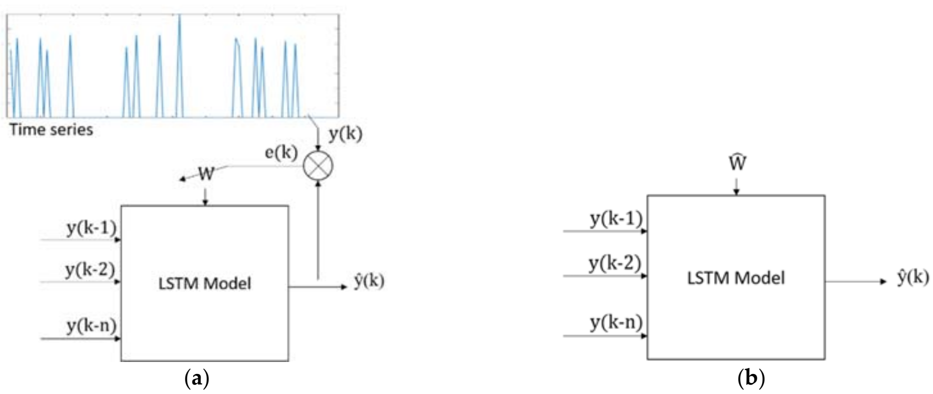


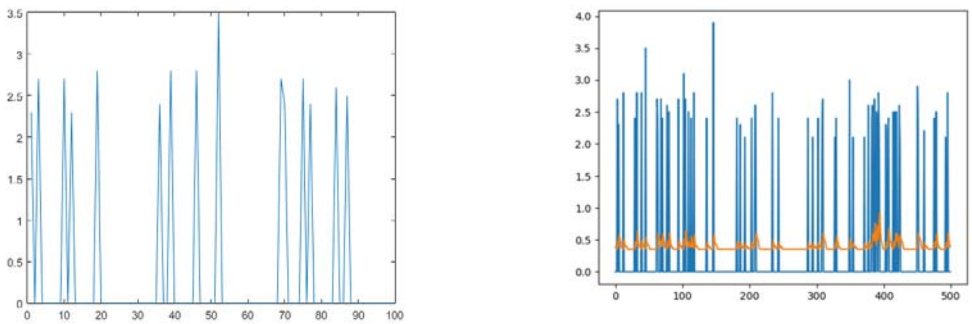
Figure 2. LSTM model for training (a), and prediction(b).

3. Results and discussion

3.1. The magnitude of the next event

In this work we divided the whole observation period in non-overlapping windows of duration 1 hour and considered only the event with the largest magnitude occurred in each hourly window.

Figure 3 (a) shows, an as example, the first 100 hours of our magnitude time series: some windows have magnitude 0, which means that no earthquakes occurred; actually, these values are not part of the behavior of the system and they really make very difficult the training of the network, as it is shown in Figure 3 (b).



(**a**) (**b**)

Figure 3. (a) Hourly maximum earthquake magnitude time series; (b) training of hourly maximum earthquake magnitude time series.

Therefore, it is then possible to take Zero-order hold model of the time series generating a new function that nevertheless reproduces the original behavior keeping all the peaks that correspond to the magnitudes of the events.

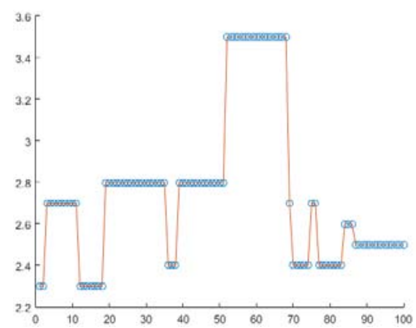
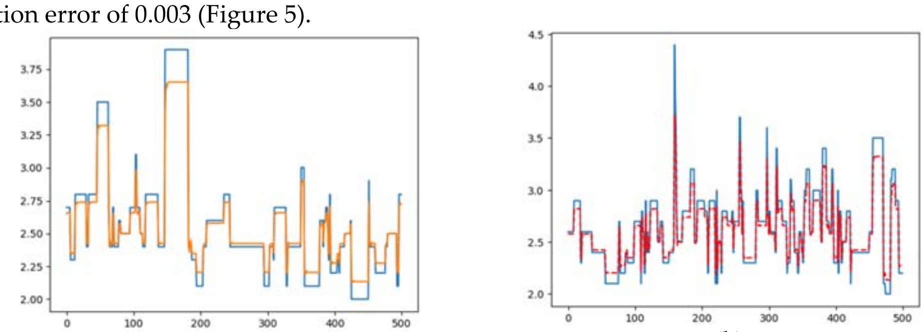


Figure 4. Hourly maximum earthquake magnitude time series, where all the zero values are filled by a value equal to last non-zero magnitude value.

Using the architecture proposed above with $n = 5$ that was the minimum delay to get a good result, with 10% of the data used for training and 2 epochs, we get a training error of 0.002 and a prediction error of 0.003 (Figure 5).

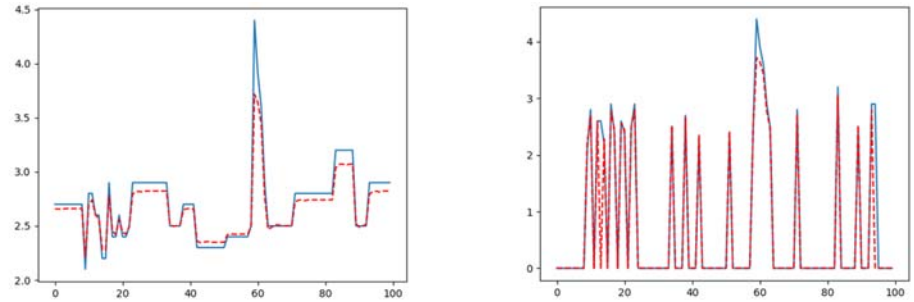


(**a**) (**b**)

Figure 5. Hourly maximum earthquake magnitude time series without zero values: (a) training, (b) prediction.

The accuracy of this prediction depends on the absence of contiguous events in the original time series with the same magnitude. However, in our seismic dataset no contiguous hourly windows were found with same maximum magnitude.

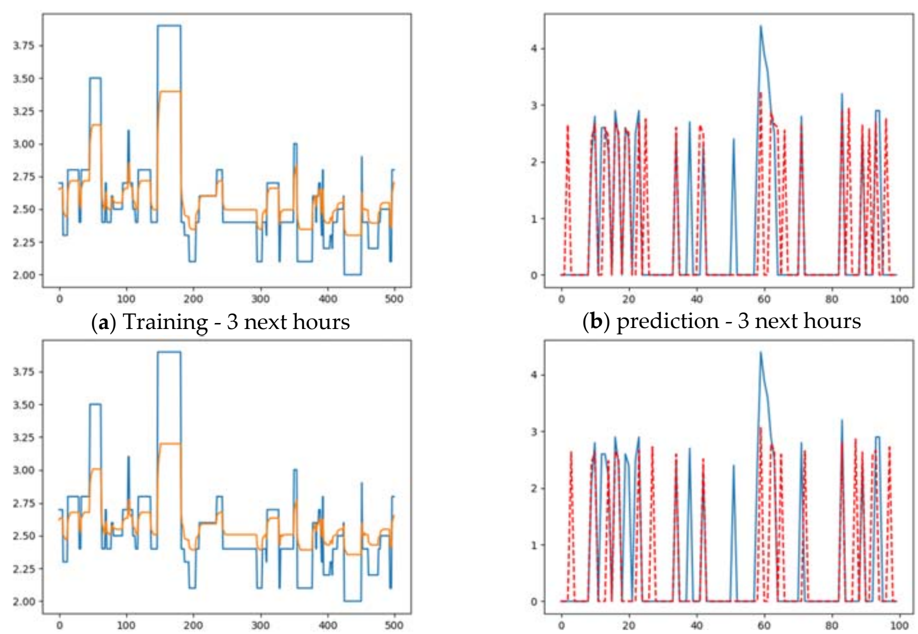
Then, after trained the model, in the prediction we returned to zero all those values that were used to fill the zeros in the original series, obtaining the original model as shown in Figure 6.



(**a**) (**b**)

Figure 6. Prediction (red) versus original (blue) time series: (a) with no-zero values, (b) after removing all the no-zero values that were used to fill the zeros in the original series.

By this procedure, the prediction of the maximum magnitude in the next hour can be performed with a minimum error; however, the prediction of the maximum magnitude in the next 3 hours, using as an input $y(k-3), \cdots, y(k-8)$, or in the next 5 hours, using as input $y(k-5), \cdots, y(k-10)$, does not furnish so good results, since the prediction error grows quite rapidly, as shown in Figure 7 (b, d).



(**a**) Training - 3 next hours
(**b**) prediction - 3 next hours
(c) Training - 5 next hours
(d) prediction - 5 next hours

Figure 7. Predictions of more than one hour

In the present work, we considered the maximum magnitude of events occurring in 1 hour; if the size of the window would be 1 day, the pattern of the zeros, as shown in Figure 3(a), changes, because the zero values become less numerous, and the prediction becomes more complex.

Raising the magnitude threshold of the catalog from 1.5 to 2, the pattern becomes similar and the same model previously analyzed can be applied. In case we need to predict the maximum magnitude of the events on a monthly basis, then the threshold magnitude should be even higher, to have a similar pattern, but the number of monthly windows would be lower than that of the daily or hourly windows, being not enough to have reliable predictions.

4. Conclusions

In this work, the prediction of the largest magnitude of the events occurring in the next hour was performed by using a recurrent neural network model and applied to the seismic catalogue of Italy spanning from 1995 to 2018. Our recurrent neural network model was found to be good enough to make a prediction of the magnitude of earthquakes, taking as the only available information the series of magnitudes. However, since earthquakes are characterized by several variables, these could be added to the network and possibly find more robust patterns that could further minimize the prediction error.

References

1. Kagan, Yan Y, Are earthquakes predictable?. Geophysical Journal International, December 1997, 131 (3): 505–525, Bibcode:1997GeoJI.131..505K, doi:10.1111/j.1365-246X.1997.tb06595.x
2. Rikitake, Tsuneji, Classification of earthquake precursors, Tectonophysics, 1 May 1979, 54 (3–4): 293–309, Bibcode:1979Tectp..54..293R, doi:10.1016/0040-1951(79)90372-X
3. Ghaedi, K., & Ibrahim, Z, Earthquake prediction. Earthquakes-Tectonics, Hazard and Risk Mitigation, 2017. 205-227.
4. Wang, Q., Guo, Y., Yu, L., & Li, P. Earthquake prediction based on spatio-temporal data mining: an LSTM network approach. IEEE Transactions on Emerging Topics in Computing. (2017).
5. Asim, K. M., Martínez-Álvarez, F., Basit, A., & Iqbal, T, Earthquake magnitude prediction in Hindukush region using machine learning techniques. Natural Hazards. 2017, 85(1), 471-486.
6. Narayanakumar, S., & Raja, K. A BP artificial neural network model for earthquake magnitude prediction in himalayas, india. Circuits Syst, 2016, 7(11), 3456-3468
7. Jordan, T. H., Chen, Y. T., Gasparini, P., Madariaga, R., Main, I., Marzocchi, W., Papadopoulos, G., Sobolev, G., Yamaoka, K., & Zschau, J. (2011). Operational earthquake forecasting. State of knowledge and guidelines for utilization. *Annals of Geophysics*, *54*(4).
8. Gonzalez, J., & Yu, W. Non-linear system modeling using LSTM neural networks. IFAC-PapersOnLine, (2018). 51(13), 485-489.