Mus, S. [et al.]. Long short-term memory networks for earthquake detection in Venezuelan regions. A: International Conference on Machine Learning, Optimization, and Data Science. "Machine Learning, Optimization, and Data Science, 5th International Conference, LOD 2019: Siena, Italy, September 10-13, 2019: proceedings". Berlín: Springer, 2019, p. 751-754. The final authenticated version is available online at https://doi.org/10.1007/978-3-030-37599-7_62

Long Short-Term Memory Networks for Earthquake Detection in Venezuelan Regions

Sergi Mus[1], Norma Gutiérrez[1], Ruben Tous[1], Beatriz Otero[1], Leonel Cruz[1], David Llácer[1], Leonardo Alvarado[2], and Otilio Rojas[3,4]

[1] Universitat Politècnica de Catalunya (UPC). Barcelona, Spain
[2] Fundaciòn Venezolana de Investigaciones Sismológicas, Caracas 1070, Venezuela
[3] Universidad Central de Venezuela, Facultad de Ciencias, Caracas, Venezuela
[4] Barcelona Supercomputing Center (BSC)

Abstract. Reliable earthquake detection and location algorithms are necessary to properly catalog and analyze the continuously growing seismic records. This paper reports the results of applying Long Short-Term Memory (LSTM) networks to single-station three-channel waveforms for P-wave earthquake detection in western and north central regions of Venezuela. Precisely, we apply our technique to study the seismicity along the dextral strike-slip Boconó and La Victoria - San Sebastián faults, with complex tectonics driven by the interactions between the South American and Caribbean plates.

Keywords: Earthquake detection· neural networks· deep learning· LSTM

1 Introduction and Related Works

Most earthquake detection methods are designed for moderate and large earthquakes, and fail to detect low-magnitude events, buried in seismic noise. However, correctly detecting these earthquakes through the existing seismic records is key to understanding their causes and to mitigate the seismic risk. This paper reports the results of an approach to apply Long Short-Term Memory (LSTM) networks over seismic data collected by broadband stations at western, central and northern Venezuela, during the time period of 2015 to 2018. The seismicity in the region results from the right-lateral strike-slip faulting experienced along the interface between the Caribbean and South American plates, as the former moves to the east with respect to the latter. A review of the seismic history and tectonics of our study area and related regions can be found in [3]. Artificial Neural Networks have been actively applied to earthquake detection since mid-late 90s. In particular recurrent networks, well suited for recognition and inference of temporal patterns, have been used for small-event detection in noisy data in [7] and [6], and for early warning systems and earthquake forecasts in [4]. The network architectures in [6] and [4], present few preprocessing convolutional layers, preceeding the core LSTM structure, in a similar way to our current approach.

2 Methodology

2.1 Dataset and Preprocessing

The Venezuelan Foundation of Seismological Research (FUNVISIS) network counts with 40 broadband stations recording three-channel continuous data at 100 Hz. The input dataset includes waveforms from seismic events with magnitudes in the range [1.7, 5.2] Mw that occurred between 2015 and 2018. Few of these events took place on the western state of Tachira, while hypocenters of the remaining bigger set are located across the Northcentral states of Carabobo, Aragua and Miranda.

Input waveforms are first normalized and divided into single-station streams. In P signal detection, the Z component is the most relevant and that is why the rest of the components are deleted [2]. After that, we cut the signals in 50 seconds windows, which gives us windows of 5001 samples. Then, we compare this windows starting and final times to the times specialized analysts label as containing an event, to determine if a window contains an event or not. After that, we put the windows containing events in one folder and the ones not containing on another one. From this folders we load the same number of events and noise windows so we have a balanced dataset. Then, the dataset is splitted into 20% for validation and the rest for training.

2.2 Network

In our first experiment, we tried to use only LSTM layers with some fully connected layers. This first experiments did not give positive results, since the accuracies did not exceed 65%. We hypothesized that this could be due to the erratic nature of seismographic signals. In order to solve this problem, we added some convolutional layers before the LSTM layers. The reasoning behind these layers that we added is to first extract the features using these layers and afterwards to feed this features to the LSTM.

Our final network model (see Figure 1) starts with three convolutional layers, each one of them followed by a max-pooling layer. Convolutional layers had 128 filters each, and a window size of 3 samples. As said previously, the purpose of these layers is to extract all the important features from the wave signal. After these layers, three LSTM layers were added with 128 units each one. Furthermore, we have used the CuDNN implementation [1]. This CUDA library enables to use GPU acceleration to train our neural network, which makes the training process remarkably faster. The main goal of this step is to extract time related features from the input signal and to extract the important data from it. The last phase consists of two fully connected layers with 128 units and a rectified linear activation function (ReLU). The purpose of this step is to make the final classification after having extracted all the features. And after the two dense layers a final output layer with a single perceptron, that has a sigmoid activation function which classifies the signal, as containing an event or containing just noise. The neural network was trained with the ADAM optimizer [5] with a learning rate of 0.001. The loss function used is binary cross entropy, which was chosen because the problem can be summed up to be a binary classification problem.

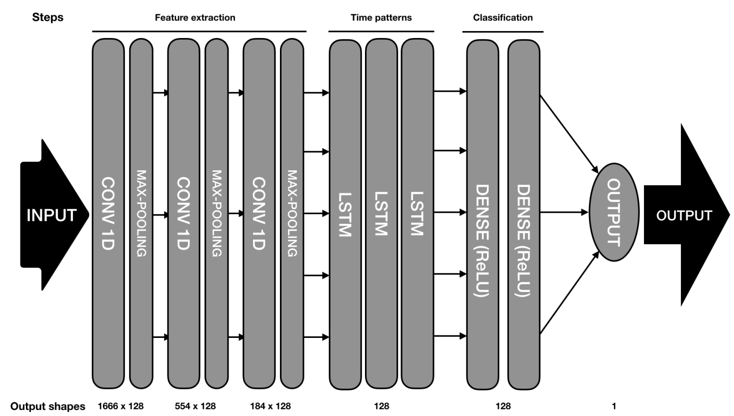

Fig. 1. Network architecture with 3 convolutional+max_pooling layers, 3 LSTM layers and 2 fully connected layers.

3 Experiments and Results

The network has been trained using a system with an Intel Core i7-4770 CPU @ 3.40GHz, 16GB of RAM memory and an NVIDIA GeForce GTX 1060 6GB. The version of Tensorflow used is 1.12.0 with CUDA 9.0.176. During the training process a Tensorboard callback was used, which enabled to monitor the accuracy and loss over each epoch (see Figure 2). This way the networks could be compared to each other. After some networks were trained, we determine which architectures and parameters gave the biggest benefits and those were kept while the underperforming ones were rejected. Finally, the accuracy has been topped out at 97.61%, with a difference in loss between the training and validation datasets smaller than 1%. This indicates that no overfitting takes place.

4 Discussion and Conclusions

In this paper we have reported the results of an approach to apply LSTM networks over single-station waveforms for P-wave based earthquake detection in western and North Central Venezuelan regions. Our data contains waveforms from diverse magnitude and cause earthquakes recorded by more than 40 heterogeneous and distant seismic stations, within a wide geographic area encompassing four different geological faults. We obtain a 97.61% accuracy when we

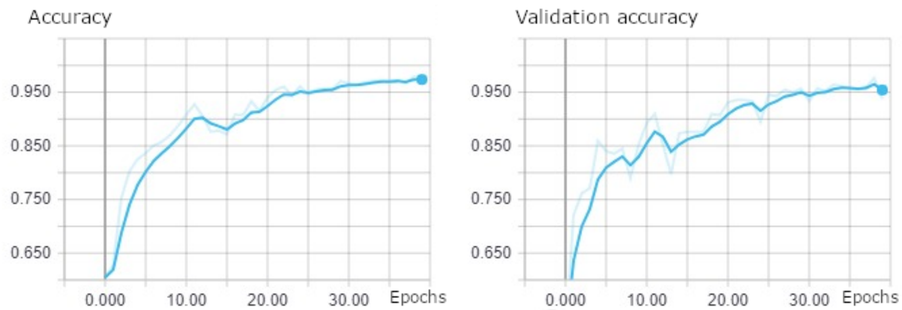

Fig. 2. Evolution of training (left) and validation (right) accuracy during training.

train the network for 40 epochs. As future work, we will address the problem of estimating the pick time for the P-wave with an Encoder-Decoder LSTM, approaching it as a sequence-to-sequence problem. We also plan to apply our approach to S-wave detection.

Acknowledgements

This work is partially supported by the Spanish Ministry of Economy and Competitivity under contract TIN2015-65316-P and by the SGR programmes (2014-SGR-1051 and 2017-SGR-962) of the Catalan Government. We thank FUNVISIS for providing the seismic data subject of our current studies.

References

1. Nvidia cudnn. https://developer.nvidia.com/cudnn, accessed: 2018-06-15
2. Aki, K., Richards, P.G.: Quantitative Seismology. University Science Books, 2 edn. (2002), http://www.worldcat.org/isbn/0935702962
3. Audemard, F.A.: Revised seismic history of the el pilar fault, northeastern venezuela, from the cariaco 1997 earthquake and recent preliminary paleoseismic results. Journal of Seismology **11**(3), 311–326 (Jul 2007). https://doi.org/10.1007/s10950-007-9054-2
4. Ibranhim, M.A., Park, J., Athens, N.: Earthquake warning system: Detecting earthquake precursor signals using deep learning networks. In: Proceedings of the University of Stanford. pp. 1–7. CS230, Stanford, California, USA (2018)
5. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. In: 3rd International Conference on Learning Representations, 2015, San Diego, CA, USA (2015)
6. Mousavi, S.M., Zhu, W., Sheng, Y., Beroza, G.C.: CRED: A deep residual network of convolutional and recurrent units for earthquake signal detection. CoRR **abs/1810.01965**, 1–23 (2018)
7. Wiszniowski, J., Plesiewicz, B., Trojanowski, J.: Application of real time recurrent neural network for detection of small natural earthquakes in poland. Acta Geophysica **62**(3), 469–485 (2014)