Toward Knowledge Extraction in Classification of Volcano-Seismic Events: Visualizing Hidden States in Recurrent Neural Networks

Manuel Titos, Luz García, Milad Kowsari, and Carmen Benítez

***Abstract*—Understanding how deep hierarchical models build their knowledge is a key issue in the usage of artificial intelligence to interpret the reality behind data. Depending on the discipline and models used, such knowledge may be represented in ways that are more or less intelligible for humans, limiting further improvements on the performance of the existing models. In order to delve into the characterization and modeling of volcano-seismic signals, this article emphasizes the idea of deciphering *what* and *how* recurrent neural networks (RNNs) model, and how this knowledge can be used to improve data interpretation. The key to accomplishing these objectives is both analyzing the hidden state dynamics associated with their hidden units as well as pruning/trimming based on the specialization of neurons. In this article, we process, analyze, and visualize the hidden states activation maps of two RNN architectures when managing different types of volcano-seismic events. As a result, the class-dependent discriminative behavior of most active neurons is analyzed, thereby increasing the comprehension of the detection and classification tasks. A representative dataset from the *deception island volcano* (Antarctica), containing volcano-tectonic earthquakes, long period events, volcanic tremors, and hybrid events, is used to train the models. Experimental analysis shows how neural activity and its associated specialization skills change depending on the architecture chosen and the type of event analyzed.**

***Index Terms*—Knowledge based systems, learning (artificial intelligence), supervised learning, machine learning, deep learning, representation learning, pattern analysis, seismology, volcanoes, volcanic activity.**

I. Introduction and Related Works

INTERPRETING information about the knowledge acquired by artificial intelligence models is an intriguing task. For decades, machine learning researchers have tried to decipher the knowledge-based upon which models construct their decision-making rules. Depending on the tasks to be solved and the models used, this information may be represented in a more or less intelligible way for humans. This notion of general model understanding is called interpretability or explainability [1].

A decade ago, this representation process was focused on the interpretation of results from the theoretical foundations of the existing models, such as support vector machines, discriminant analysis, decision trees, or hidden Markov models among others [2], [3]. However, with the overwhelming growth of both massive data acquisition and more powerful hardware processing systems, new machine learning data-based processing methods have emerged [4]. These newly generated methods, known as deep learning, mainly learn hierarchical representations of data. Representations with a higher level of abstraction are obtained from multiple iterative nonlinear transformations using raw data [5].

These methods have become the state-of-the-art for problems, which were previously regarded as difficult, such as natural language processing (NLP) (including language modeling, machine translation, speech recognition, and sentiment analysis), reading comprehension, or video analysis [6]–[9]. Nevertheless these methods are often considered as "black boxes" due to the lack of understanding of the mechanisms behind their effectiveness. While they are able to approximate any function, studying their structure will not help us to decipher the true nature of the function approximated, as their high internal complexity and nonlinear structure make it difficult to understand the underlying processes through which they acquire knowledge. Understanding how deep hierarchical models build their knowledge is an incipient line of research, which emphasizes the idea of abstraction based on the specialization of substructures, neurons, or units as a key to improve the usability and effectiveness of the models.

The benefits of comprehending these underlying learning mechanisms have already become apparent in the classification of images. Given the symbolic content of images, the knowledge acquired can be represented as a set of hierarchical visual abstractions, and the activation of neurons becomes more than the simple index or numerical value associated to them. As a whole, it produces symbolic representations such as parts of an object within a certain framework. Visualizing and interpreting activation maps representing the state dynamics associated with hidden layer neurons [10]–[13] has become a widespread method to analyze features learned by the models. Works presented so far have been applied to image segmentation problems based on

Manuscript received October 15, 2021; revised January 20, 2022; accepted February 25, 2022. Date of publication March 3, 2022; date of current version March 23, 2022. This work was supported in part by the MINECO under Grant PID2019-106260GB-I00 FEMALE and in part by the FEDER/Junta de Andalucia-Consejería de Economía y Conocimiento/ Proyecto A-TIC-215-UGR18. *(Corresponding author: Manuel Titos.)*

Manuel Titos was with the Department of Signal Theory, Telematics and Communications, University of Granada, 18071 Granada, Spain. He is now with the Processing and Research Division, Icelandic Meteorological Office, 105 Reykjavík, Iceland (e-mail: manuel@vedur.is).

Luz García and Carmen Benítez are with the Department of Signal Theory, Telematics and Communications, University of Granada, 18071 Granada, Spain (e-mail: luzgm@ugr.es; carmen@ugr.es).

Milad Kowsari is with the Faculty of Civil and Environmental Engineering, School of Engineering and Natural Sciences, University of Iceland, 102 Reykjavík, Iceland (e-mail: milad@hi.is).

Digital Object Identifier 10.1109/JSTARS.2022.3155967

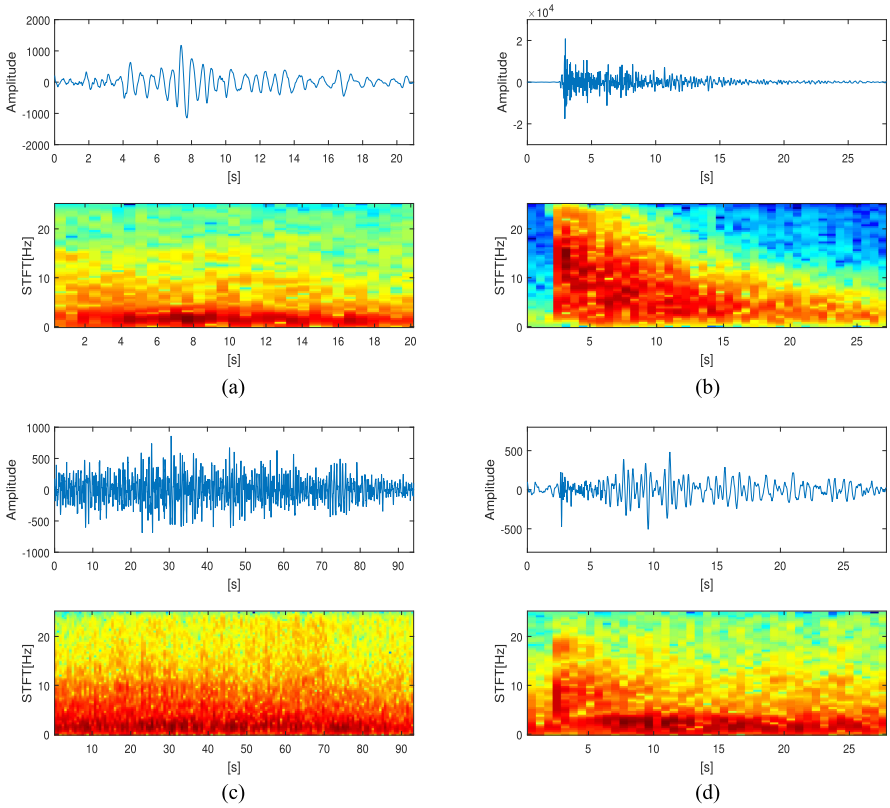

Fig. 1. Amplitude and spectrograms of the four types of VSRs recorded at Deception Island volcano during three seismic surveys: 1994–1995, 1995–1996, and 2001–2002. (a) LP. (b) Volcano-tectonic (VT) earthquake. (c) Tremor (TRE). (d) HYB.

the analysis of saliency maps or abstract correlations between neurons. In so doing, images are represented as a set of contours where each of them is computed depending on properties, such as color, texture, or intensity [14]–[16]. More recently, Arif *et al.* [17] proposed an action recognition framework by utilizing motion maps based on frame-level deep features of input video.

A. Volcano-Seismic Data: Modeling 1-D Temporal Structures

There are, however, many disciplines in which inputs to the neural network are not images but rather 1-D temporal series that require the modeling of continuous sequential data with a temporal structure [18]–[25]. Examples of such disciplines are, among many others, NLP, network traffic anomalies detection, analysis of biometric signatures, stock market analysis, or automatic recognition of volcano-seismic events (VSRs), the latter of which is the target application of this proposal. In those 1-D modeling networks, the interpretation of neurons excited in certain spatial areas of the architecture is not an easy task. Several works have started to tackle the challenge of visualizing hidden states in the field of NLP. The interactive visualization of hidden states proposed by Strobelt *et al.* [13], the comparison of RNN models and coclustering of hidden states proposed by Ming *et al.* [26], or the visual tool to explore sequence-to-sequence models of work [27] are examples. In the field of biometric patterns processing, Kwon *et al.* [28] provided a visual tool to increase interpretability and interactivity of RNN predictions on electronic medical records.

To the best of authors' knowledge, there have been no attempts to visualize hidden states in the neural network approaches used up to now to process volcano-seismic data. Through a wide range of state-of-the-art technologies [29], automatic recognition of VSRs uses the enormous amounts of seismic data registered in vulcanologic observatories to detect and classify the different types of volcanic events by identifying their source mechanisms. In so doing, it provides valuable help in implementing early warning systems and reducing the risk of potential eruptions. Certain types of events detected on-the-fly over the continuous registers are clear precursors of volcanic eruptions [30]. Therefore, the temporal structure of seismic registers is a key issue. The sequence of events detected together with their frequency, energy, and combination, provides fundamental information about the internal dynamics of the volcano. In addition, the temporal structure of each event detected renders important details about the evolution of the source mechanism producing them [30]–[32]. As an example, Fig. 1 describes the events of the Deception Island volcano (Antarctica) analyzed in this article and shows their temporal structure both in time and frequency domains. Variations in amplitude and frequency during the event are direct consequences of the source mechanism defining said event and can, therefore, serve to identify it. Given this clear temporal structure and the interest of the specific sequence of events occurring, several continuous VSR systems have been proposed in the literature using memory-based architectures, such as hidden Markov models [33], or the current state-of-the-art recurrent neural networks (RNNs) [34], and temporal

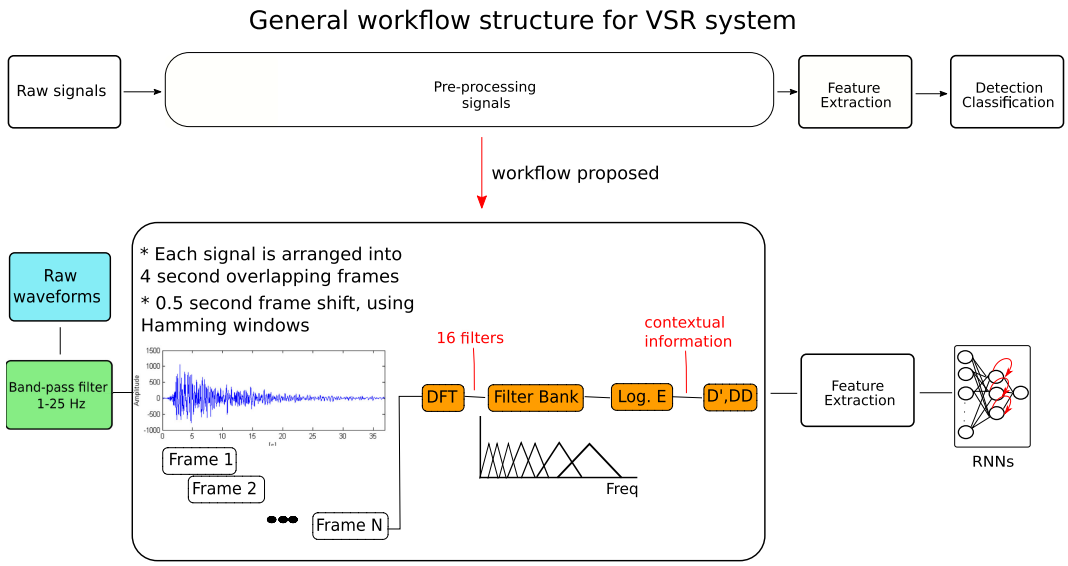

Fig. 2. Workflow structure of a common VSR system (top) and the specific preprocessing used in this article (bottom). Specific preprocessing illustrates the feature engineering process based on frequency analysis (in the logarithmic filter bank domain) used as input to classify the seismic records using an RNN.

convolutional neural networks (TCNs) [35]. Fig. 2 shows the common experimental methodology followed by continuous VSR systems [36]. According to the requirements of each architecture, the input information (raw signals) is preprocessed and parsed into a set of descriptive features, e.g., spectrograms, filters bank in logarithmic scale, or linear prediction coefficients. Resulting feature vectors are further used in geological modeling frameworks through different machine learning approaches [37].

B. RNNs and TCNs as Memory-Based Approaches for Continuous VSR Systems

This article presents a tool to visualize and understand hidden states of RNNs used for automatic detection and classification of VSRs. Experiments have been performed to justify the use of RNNs for VSR systems instead of TCNs, the other state-of-the art memory-based approach optimal for capturing intra and interevent temporal dependencies in temporal data series.

Classical convolutional neural networks (CNNs) are the state-of-the-art in many applications because they present lower training times and yield similar results to those of RNNs [38] for several time-sequences problems. However, VSRs do not have a fixed duration for events of the same class nor for events of different classes. The only possible way to use CNNs would be to do a mere classification of frames, i.e., time stamps, without any temporal dependence. In doing so, worthwhile information would then be lost.

To take into account such temporal information in convolutional models, TCNs could be applied [39]. Bai *et al.* [40] make a careful comparison of the potential of RNNs and TCNs for temporal series analysis highlighting factors, such as the memory retention needs or data storage capabilities during evaluation. RNNs are usually associated with longer training times and lower parallelization, given their training method of gradient propagation through time. Nevertheless, they are still an optimal tool to manage temporal series, in particular when the model needs to capture long-term time dependencies and handle varying sizes of input. While TCNs present shorter training times and a greater degree of parallelization, since their memory capacity is obtained from the stacking and dilation of convolutions between layers, they do, however, need a greater number of parameters to be tuned. This can be a disadvantage when data are scarce or the available databases are not very large.

C. Contributions of This Proposal

In this article, we propose and explore several visualization approaches based on neural activation maps of RNN trained with continuous volcano-seismic data. The idea behind this exploration is to analyze the hidden state dynamics of two recurrent architectures, Vanilla and long short-term memory (LSTM), in order to understand how architectures influence recognition tasks and knowledge extraction.

In addition, we highlight the idea of the pruning/trimming based on the specialization of neurons as a key to improve the interpretation of the seismic data. The activity changes of highly specialized neurons can be interpreted as a useful tool to segment, analyze, and label new seismic records from which to address active learning approaches and improve the performance of the existing models. Such information can be highly useful for volcanological observatories whether they are interested in either simple systems focused only on event recognition yielding a certain event label or interested in more specialized tools to improve the information about the events analyzed, e.g., the time evolution of their frequency activity or their similarity to other types of events in terms of neurons activation. This approach can leverage users' domain expertise and use hidden knowledge as inputs to design new models and applications.

The contributions of this work are fourfold. First, according to the best of authors' knowledge, this is the first work linking the

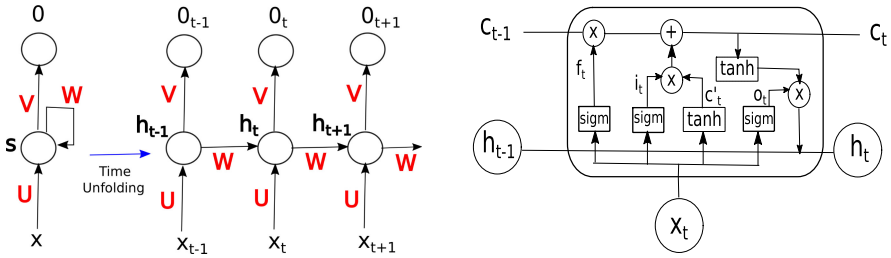

Fig. 3. (a) General overview of an RNN architecture. An input sequence X $= x_0, x_1, \ldots, x_{n-1}$ is mapped into an output sequence O $= o_0, o_1, \ldots, o_{n-1}$. (b) LSTM cell description showing how gate mechanisms are connected. i_t, f_t, and o_t correspond to input, forget, and output gates, respectively, and are used to regulate the amount of information that will flow through the time steps. c_t is an internal state or memory cell, which combines the previous memory and the new input information. Finally, h_t corresponds to the hidden state at time step t and is computed using the information contained in both, c_t and o_t at this time step.

neuron activation maps corresponding to RNNs and the classification of volcano-seismic signals in order to better understand the process. Second, this article shows how the use of different architectures leads the models to obtain different specialization skills. Third, we illustrate how the absence of sophisticated memory cells and the use of very short-term information in classic RNNs leads to a high neuronal activity and decreases the performance of the systems. Fourth, considering that the degree of specialization of the models is closely related to the architecture and the topology chosen, we analyze the relevance of less specialized units in the final performance of the models and thereafter conclude that they play an important role in the discrimination of the events.

The rest of this article is organized as follows. Section II provides the theoretical framework of RNNs. Section III introduces the dataset, parameterization approach, and experimental setup. Section IV defines the concepts of average neuron polarization, susceptibility, and specialization of architectures and presents the visualization approaches proposed. Visualization results are analyzed in Section V. Section VI-A focuses on the analysis of neuron specialization. Finally, Section VII concludes this article.

II. Recurrent Neural Networks

RNNs [41] are bioinspired computing algorithms that are able to capture the temporal dependencies between data. Basically, an RNN (see Fig. 3) is a classical neural network, which maps an input sequence X $= x_0, x_1, \ldots, x_{n-1}$ into an output sequence O$= o_0, o_1, \ldots, o_{n-1}$ using nonlinear transformations of the input information at time step t_n and the latent information at time step t_{n-1}.

Each time step can be considered as an additional layer in a classical neural network where the parameters are shared over time. This property drastically reduces the number of parameters to be tuned. Similar to the classical neural networks, nonlinear transformations project information into a new linearly separable representation space, which facilitates the subsequent prediction, detection, or classification tasks.

A. Vanilla Architecture

In its simplest version, called Vanilla architecture, the hidden state or memory of an RNN that collects the latent information at each time step h_t, is obtained from the latent information at the previous time step h_{t-1} and the incoming information x_t as follows:

$$h_t = \sigma(x_{(t)} * U + h_{(t-1)} * W + b) \tag{1}$$

where σ corresponds to a nonlinear activation function as tanh, sigmoid, or ReLU, U describes the parameters relating the latent information with the incoming information, W relates the latent information at time step t with the latent information at the time step $t-1$, and b corresponds to the bias values of the hidden units.

The output of the model o_t at time-step t can be obtained as

$$o_{(t)} = \text{softmax}(V * h_{(t)}) \tag{2}$$

where V is the weight matrix from hidden to the output layer, and softmax() is the softmax function applied over the outputs for computing the normalized per-class output probabilities.

The training stage of an RNN is performed through the error function *gradient propagation* using a procedure known as back-propagation through time (BPTT) [42]. Unlike CNNs, in RNN architectures, the setting of the gradient depends not only on the current time-step but also the previous ones. This dependence between gradients often leads the models to exploding/vanishing scenarios preventing them from learning long-term dependencies [43], [42].

B. Long Short-Term Memory Architecture

LSTMs are a complex variant of RNN in which the hidden state h_t is computed using a gate mechanism and a memory cell. These models have the ability to model long-term dependencies, removing or adding information inside the memory cell state and allowing the information to flow through time.

As shown in Fig. 3(b), the LSTM architecture introduces the following [42].

1) A three-gate mechanism composed by input i_t, forget f_t, and output o_t gates, respectively, regulating the amount of information that will flow through time steps. On the one hand, *input* and *forget gates* regulate the quantity of information related to the input and the previous state used to compute the hidden state of the current time step. On the other hand, *output gates* regulate the quantity

of information related to the internal state exposed to compute the hidden state at the next time step.
2) An internal state or memory cell c_t, which combines the previous memory and the new input information.

The hidden state h_t at time step t is computed using the information contained in both the memory cell (c_t) and the output gate (o_t) at this time step. Output states are computed using a softmax layer following the same procedure used for the vanilla-RNN [see (2)]. The training stage is carried out via BPTT with a defined loss-function.

III. RNNs as VSR Systems

This section provides a brief description of how RNNs are used for automatic recognition of VSRs. Titos *et al.* [34] rendered a full description of the approach. In further sections implementation details are given.

A. Parameterization Scheme

Drawing on the parameterization schemes used in [34], we implement a different approach, adding temporal context information noted as log filter bank (LFB) + $\delta, \delta\delta$. Feature vectors are extracted as follows [see (2)].

1) Raw data are windowed with 4 s Hamming windows and 3.5 s overlapping. The sampling frequency of the waveform is 100 Hz.
2) For each window or frame, a 512-point FFT is computed. The magnitude of the spectrum obtained is used as input into a bank of 16 triangular filters, which are uniformly distributed on a logarithmic frequency scale with 50% overlapping between adjacent filters. Thereafter, the logarithm of the output filter-bank energies is computed. As a result, each window provides a feature vector of 16 components.
3) In addition to the information provided by the filter bank, we include information about the temporal context, adding the first and second order temporal derivatives ($\delta, \delta\delta$) for each frame [44]. The size of the feature vector will triple the initial one.

B. Dataset

This section provides a brief summary of the geology background of the survey, a small introduction of the data and the per-class distribution of events. A full description of the dataset, sensor, and acquisition systems can be obtained in [34] and [45], respectively.

Deception Island (62° 59 'S, 60° 41' W) is one of the three main active volcanic islands in the South Shetland archipelago and Antarctic Peninsula. *Deception Island* is located astride a Quaternary marginal basin-spreading center in the Bransfield Strait separating the South Shetland Islands from the Antarctic Peninsula [46], [47]. Its geodynamics are typical of a rift framework and it is considered to be the main active volcano of the back arc basin of the Bransfield Strait with at least six eruptive periods recorded during the last 200 years. The volcano, from whose eruption the island emerged during the Quaternary period, is a horseshoe-shaped volcano with a submerged basal diameter of 25–30 km and an emerged structure of 15 km.

The dataset was collected deploying a dense short-period seismic antenna during three austral seismic Antarctic surveys in 1994–1995, 1995–1996, and 2001–2002 at the *Deception Island* volcano. During the first survey, 10 vertical and 3 three-dimensional (3-D) seismometers were deployed, meanwhile in the other two experiments, the array was composed of 15 vertical and 3 3-D ones. Mark L25 (with a natural frequency at 4.5 Hz and electronically extended to 1 Hz) and Mark L4 C (with a natural frequency of 1 Hz and electronically extended to 0.1) were used as vertical and 3-D seismometers, respectively. The internal clock was synchronized by GPS time every second, and the sampling rate used was 200 Hz (samples per second). Data labeling has been performed by a group of geophysicists with deep knowledge on and experience with the dynamics of this volcano. Only the most representative volcano-seismic records belonging to each seismic family (see Fig. 1) have been selected. As a result, a total of 512 continuous data streams were obtained (2193 events). Per-class distribution is the following: 1222 silences (SIL), 77 tremors (TRE), 765 long period events (LP), 75 volcano-tectonic earthquakes (VTE), and 54 hybrid events (HYB).

TABLE I
Classification Accuracy (acc %), Number of Parameters Tuned and Training Times for the Best Configurations Obtained for RNN and TCN Architectures

	RNN-Vanilla	RNN-LSTM	TCN
Test 1 acc. (%)	84.73	88.88	82.82
Test 2 acc. (%)	82.73	85.05	80.10
Test 3 acc. (%)	83.60	84.10	84.99
Test 4 acc. (%)	78.25	81.48	78.74
Avg. acc. (%)	82.39	**84.88**	81.66
No. of parameters	7220	11130	**63105**
Training times (s)	1239	16834	2073

Bold entities correspond with the architecture obtaining the best accuracy and the architecture with a greater number of parameters to tune.

C. Motivating the Use of RNNs as Continuous VSR

Table I and Fig. 11 summarizes the best recognition results obtained classifying the events on the dataset of the Deception Island volcano using both TCN and RNN for the sake of comparison. The parametrization used is as described in Section III-A, windowing each seismic record into several frames. Architectures have been configured optimally, providing the number of parameters used. In this regard, different models for each architecture have been evaluated. On the one hand, the best Vanilla and LSTM models were obtained using 140 and 210 hidden units, respectively [34]. On the other hand, TCNs were evaluated with models having different residual blocks (stacks or sets of layers considered as single layer). The numbers of filters used in the convolutional layers were ranged from 8 to 200 to be similar to hidden units for RNNs. Different dilation values (where dilation refers to the expansion of the kernel by inserting holes between its consecutive elements) were tested in order to modify the receptive field and

consequently the memory capability of the models. Dilation of the convolution was varied among four values, i.e., 8, 16, 32, and, 64 considering the length of the input signals. This parameter also defines the depth of the residual block, varying between 1 and 4 layers. In order to control the spatial area considered in the convolutional operations, the kernel size was ranged between 2 and 4 time stamps [96 and 192 features, respectively, considering that each feature vector (time stamp) has 48 features]. Larger kernel sizes will make the network much larger as well. Given the size of our dataset and the length of the signals, a kernel size between 2 and 4 was justified. Finally, considering the nature of our seismic record, we did not stack more than one residual block, which is highly useful when the sequences are quite long, e.g., in the case of waveforms with hundreds of thousands of time samples. The best model was obtained using 50 filters, a kernel size of 2, and dilation of [8, 16, 32]. Accuracies obtained for both architectures are similar. For this database and classification task, LSTM outperforms TCN by 3% but the number of parameters to be trained in TCN is 5 times lower. Comparing LSTM and Vanilla-RNN, the former outperforms the latter, with the cost of a higher number of parameters. Confusion matrices are placed in the Appendix. Training times are based on NVIDIA K40c and NVIDIA GEFORCE GTX 1080 graphic cards and correspond with the results of training and evaluating one partition.

IV. Visualizing RNNs in VSR Systems: Proposal

The analysis of the RNN internal activation patterns can give insight on how the architectures work and how certain configurations are able to specialize and decrease their neural activity. Such information will help us to overcome the actual perception of neural networks as "black boxes," providing more knowledge to select optimal architectures and configurations for specific tasks in the field of seismic signal processing. For this purpose, we will analyze the architectures through the following three concepts.

1) Degree of polarization of the neurons (extrapolated from the field of Neuroscience [48]). This is the ability to set activation values close to -1 or 1 in the range $[-1,1]$ defined for activation function.
2) Susceptibility of the architecture, defined in this article as its capability to detect changes in the presence of incoming information. Architectures with higher susceptibility become more sensitive and aware of seismic activity. The perception of changes in seismic activity is useful additional information beyond the classification provided. Seismic activity detection, phase detection, or phase picking tasks can make use of the neuron's susceptibility to changes.
3) Degree of specialization of an architecture (also named neuron class-selectivity [49], [50]). This is the ability to improve the detection and classification of events by decreasing the global neural activity concentrated in a few highly sensitive neurons with high activation values. Those specialized neurons are dedicated to specific events, frequency bands, or other specific attributes.

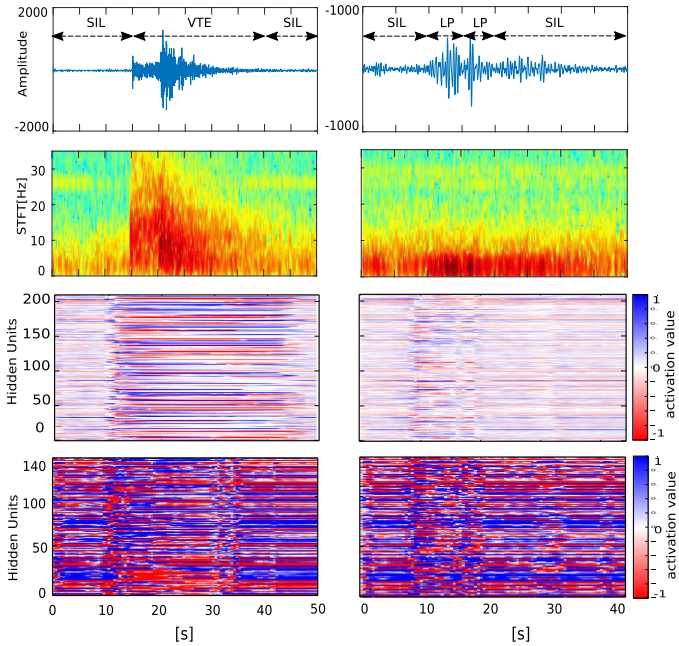

Fig. 4. Activation maps associated with the two RNN architectures. Each column depicts, respectively, the waveform and spectrogram corresponding to the seismic record (first and second rows) with the respective activation maps obtained by LSTM and Vanilla architectures (third and fourth rows). The left column shows the analysis for a volcano-tectonic event. The right column shows the performance for two concatenated LP. While waveforms are plotted on a time sample basis, activation maps and spectrograms use overlapped time windows containing several time samples to depict hidden unit activity and frequency bands related to energy. This difference explains why event arrival and ending take place sooner in time in the activation maps compared to the waveform plot.

We propose the following four visualization schemes to study the hidden state dynamics of the RNN in different scenarios with several architectures and types of events within the framework of continuous detection and classification of seismic events.

1) Neural activation maps (see Fig. 4), graphical representations (patterns) capable of describing the evolution of the activation values of the hidden units across time for each seismic event. To create these maps, activation values of the hidden units at each time step are mapped into a color diagram, where rows correspond to the hidden units and columns correspond to different time steps. Each activation value (in our case, in the range $[-1,1]$ as we used a hyperbolic tangent as activation function) will be mapped in a color scale associating the lowest range values, -1, with the red color, and highest values, +1, with dark blue. This is a perfect tool to analyze the degree of polarization of the neurons and the susceptibility of the architectures over time. To the best of authors' knowledge this has not been used before.
2) Histograms of neural activation values (see Fig. 5), empirical probability distribution of the neural activation values, used in the literature to provide knowledge about the functioning of the neural architecture (e.g., [51]). They provide a global view of the different average activity levels for the hidden units compared for different events, architectures, etc. This visualization scheme is useful to

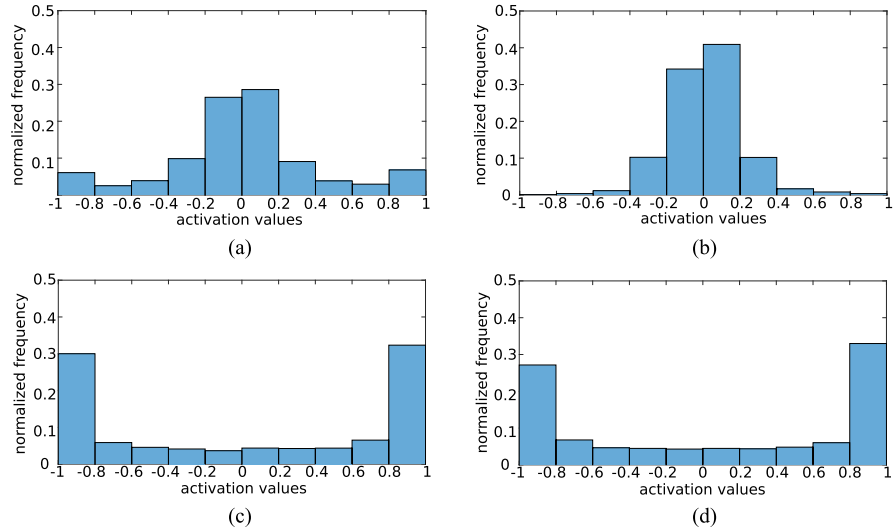

Fig. 5. Histogram associated with the activation values for a VTE and an LP event for each RNN architecture. (a) VTE (LSTM). (b) LP (LSTM). (c) VTE (Vanilla). (d) LP (Vanilla).

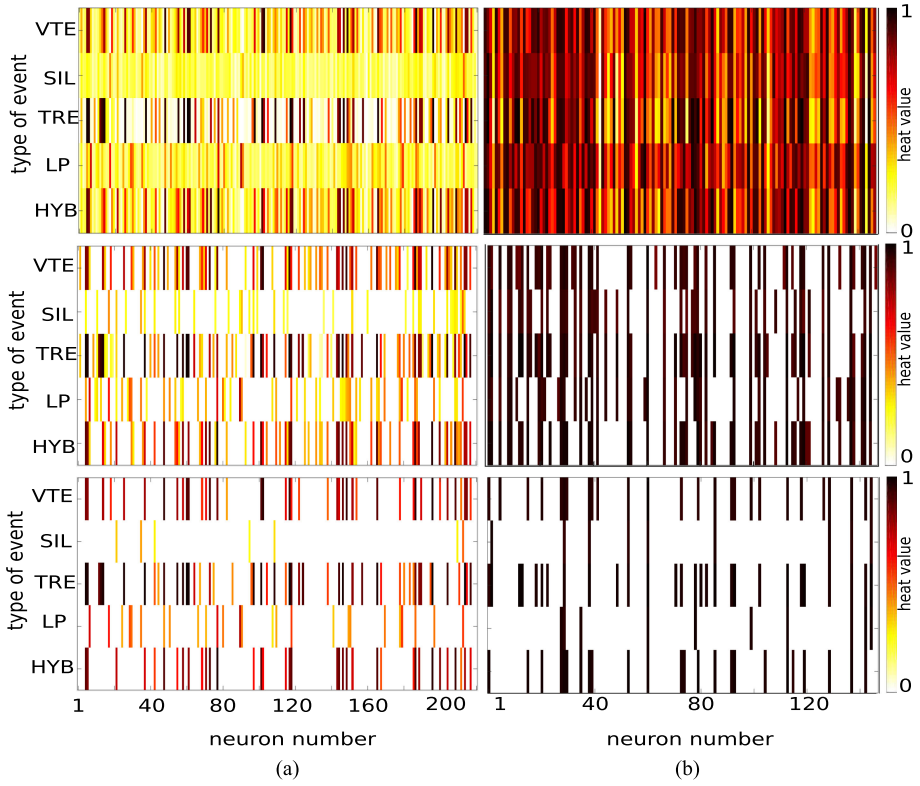

Fig. 6. Heat map associated with the hidden units of the LSTM (left) and the Vanilla (right) architectures for each type of events using 100% (top), 50% (middle) and 30% (bottom) of the most activated common neurons. The number of the neuron, from 1 to 210/140 in the LSTM/Vanilla architecture, is used to identify each specific neuron.

analyze the sparsity of the neuronal architecture. Sparse neural network structures, with average activation values distant from extreme values -1 and 1, are beneficial as they improve the performance by easing the training convergence and improving the generalization capability [52], [53]. The distribution of activation values also provides information on the capacity of the architecture to specialize neurons in certain tasks or types of events. Section VI-A will demonstrate that neuron specialization is related to sparsity and will further develop this analysis.

3) Heat maps of the hidden units for each type of event (see Fig. 6), considering the unfolding process that takes place in the RNNs, we compute the heat values of each hidden unit, defined as the average of its absolute activation values over time. Once the absolute activation values have been obtained, events are clustered by type and the heat value

is computed by averaging the absolute activation values for each hidden unit recognizing all the events of the same type. As a result, a heat map showing the activity level of each unit recognizing a specific type of seismic event is obtained. The resulting graph shows structural similarities/alignments in the network architecture for different events that might be useful for unclear challenging detection or classification tasks. Heat maps have been used in CNN image processing, mostly applied to the image under classification to identify its most discriminative parts (e.g., [54]). The novelty of this proposal is in creating the heat map of the RNN average activation values per neuron in the architecture for the different types of events.

4) Time evolution of the five most specialized hidden units for each type of event (see Figs. 8 and 9), we analyze the degree of the specialization and evolution of the hidden units in both LSTM and Vanilla architectures. For each type of event, the most active, and therefore, the most *specialized*, hidden units are first identified. Then, the temporal evolution of their activity is computed and plotted. This novel visualization approach provides relevant information on the behavior of neurons when recognizing different inputs, opening an interesting field of work. As the figures show, neurons specialize in a different manner for the different types of events. In addition, neurons are sensitive to different frequencies present in the input. The analysis of their behavior provides interesting applications for seismic analysis underlying the basic event classification expected from the neural network. Some neurons behave as activity detectors for certain frequency bands or as P or S phase detectors. In other cases, comparing specialized neurons of different types of events can also provide useful information in challenging classification tasks.

V. Visualization Results and Discussion

This section provides a visual analysis of the behavior of Vanilla and LSTM architectures when processing several seismic records for their classification. We will illustrate the most relevant differences obtained in terms of degree of polarization, specialization, and susceptibility from the perspectives of architectures and type of event.

A. Analysis Regarding Architecture

This section details how Vanilla and LSTM present different levels of neuronal activity. The neurons' susceptibility, possible event-related structural similarities and specialization patterns are analyzed.

1) Comparing Susceptibilities: As defined in Section IV, susceptibility is related to how much neurons react in the presence of incoming information. Figs. 4 and 5 compare, respectively, the neural activation maps and histograms of activation values obtained by LSTM and Vanilla architectures using LFB $\delta + \delta\delta$ as parameterization schemes for two typical events. Several conclusions can be drawn as follows.

1) The absence of memory cells and the use of short-term temporal information in the Vanilla architecture results in high neuronal activity (see Fig. 4); for each time step, Vanilla architecture tries to infer the class of the incoming event without taking into account any long-term past information. Therefore, most of the neurons are highly excited around 1 and -1 values. This conclusion is reinforced in Fig. 5, where the histograms associated with the activation values for a VTE and for a sequence of LP events for each RNN architecture are depicted. The comparison of histograms indicates that for the LSTM a greater number of neurons remain inactive or less active (around zero value). The use of more complex gating mechanisms leads to a lower polarization of the activity of the neurons. The LSTM infers what kind of event is entering the network once their gating mechanisms have been updated. Changes in the spectral content are detected earlier by complex architectures, since these are "dragging" a state of minimal activity. This property results in the accurate detection of the arrival of emerging events.
2) Once detected, the delimitation of events is also strongly influenced by the architecture implemented. This property can also be observed in the neural activation maps of Fig. 4. The ability of the LSTM to model very long-term temporary dependencies implies detecting only information changes that are maintained over time. As a consequence, the LSTM will generally place the end of the events several time instant later (therefore introducing a certain delay) than the Vanilla architecture. In turn, this delay will depend on the incoming information and is more pronounced in events whose spectral content changes more smoothly.

2) Comparing structural similarities: Using the heat map described in Section IV, we study the behavior of the architectures, using several examples of incoming information from different types of events. Heat maps represent, for a given architecture and for each neuron, the average activation value for all the events of the same nature in the database. Through their study, we investigate possible structural similarities and specialization patterns of each of the architectures and different types of events during the recognition phase. Fig. 6 corresponds to the heat map of the architectures considering 100% (top), 50% (middle), and 30% (bottom) of the most active neurons common to all events of the same class. To make the similarities among architectures and events more evident, Fig. 7 shows event-based pairwise correlation matrices of the heat map depicted in Fig. 6. From these images, we can draw several conclusions, as follows.

1) Vanilla specializes roughly the same neurons for events with similar characteristics. It is worth noting that these specializations are achieved over the polarized hidden units. On the contrary, LSTM achieves specialization through less polarized hidden units.
2) Regarding the structural similarities between the architectures and types of events, specific neuron usage can be observed. For both architectures, the specialized neurons for very characteristic events, such as VTE and HYB, coincide almost entirely (see Fig. 6). Vanilla finds several specializations over those events with similar spectral content and relative long duration (TRE, HYB, VTE). The presence of more sophisticated memory mechanisms

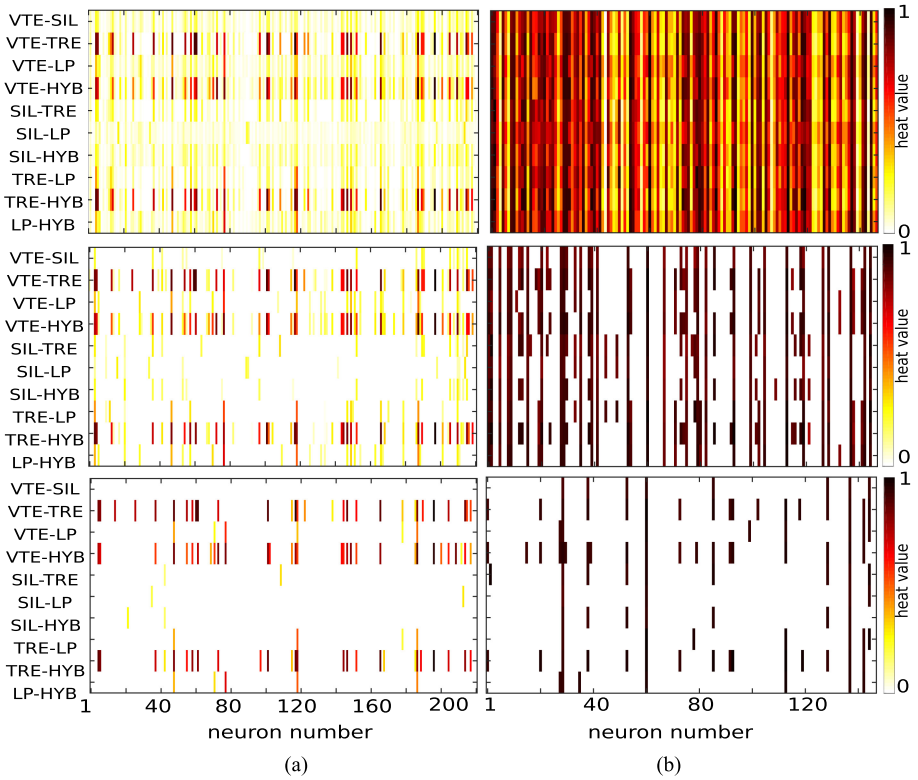

Fig. 7. Pairwise *events-based* correlation matrices of the heat maps obtained in Fig. 6. The left and right columns depict, respectively, correlation matrices for LSTM and Vanilla using 100% (top row), 50% (middle row), and 30% (bottom row) of the most active common neurons for each type of event. The number of the neuron, from 1 to 210/140 in the LSTM/Vanilla architecture, is used to identify the specific neuron.

on the LSTM allows the detection of minimal, severe and very long changes in its input, resulting in a higher specialization. Therefore, gating mechanisms allow architecture to define subsets of specialized hidden units to recognize each type of event. However, as shown in Fig. 7, LSTM also finds highly marked structural similarities over TRE, HYB, and VTE. After a posterior supervision by a geophysical expert, we can consider these similarities as correct. Often, at the beginning of the TRE events, a short and overlapped VTE can be recorded by the seismometer, but the signal has been labeled as a TRE (depending on the geophysical subjectivity of the human operator), since the source of these types of TRE is preceded by a small earthquake.

3) Finally, both architectures offer low specialization for SIL.

B. Analysis Regarding the Type of Event

This section analyzes the visualization approaches of the LSTM architecture using LFB + $\delta + \delta\delta$ as a parameterization scheme regarding the type of the event. For this purpose, we have selected some highly representative volcano-seismic records within the dataset. Then, we depict the waveform and spectrogram along with its activation map for each record (see Fig. 4). Several conclusions can be drawn from these results.

1) Considering that the LFB parameterization represents the spectral characteristics of the events, hidden units are activated or deactivated depending on the energy distribution in the different frequency bands. For low energy signals, like the sequence of LP events, as shown in Fig. 4 (third row, right column), only a few neurons present a strong positive or negative activation. Indeed, higher energies produce stronger activations in a greater number of neurons. This conclusion is reinforced in Fig. 5(b) where most of the neurons are activated between -0.5 and 0.5 values. As the signals become more energetic, an increase in the activation level of neurons is observed. It is important to note that the distribution of the histograms is not only sensitive to the energy distributed in the different frequency bands but also to the type of architecture. As the architectures become more specialized, the empirical distribution function of the activations (histograms) converge to a Gaussian with a smaller variance (see Fig. 5). The models decrease their neuronal activity to properly detect and classify each of the events. In Section VI-A, we will carry out a deeper study on the specialization of architectures.
2) The time interval during which the hidden neurons remain active also depends on the spectral content of the event (third row of Fig. 4). According to Fig. 6, the number of more excited neurons increases when the events have a wider spectral content.
3) Regarding the structural similarities between architectures and type of events, the usage of specific common neurons between the VTE and the HYB can be observed (see Fig. 6). This is due to the similarity of spectral content of the events. Contrary to what might be expected, there are no significant structural similarities between TRE and

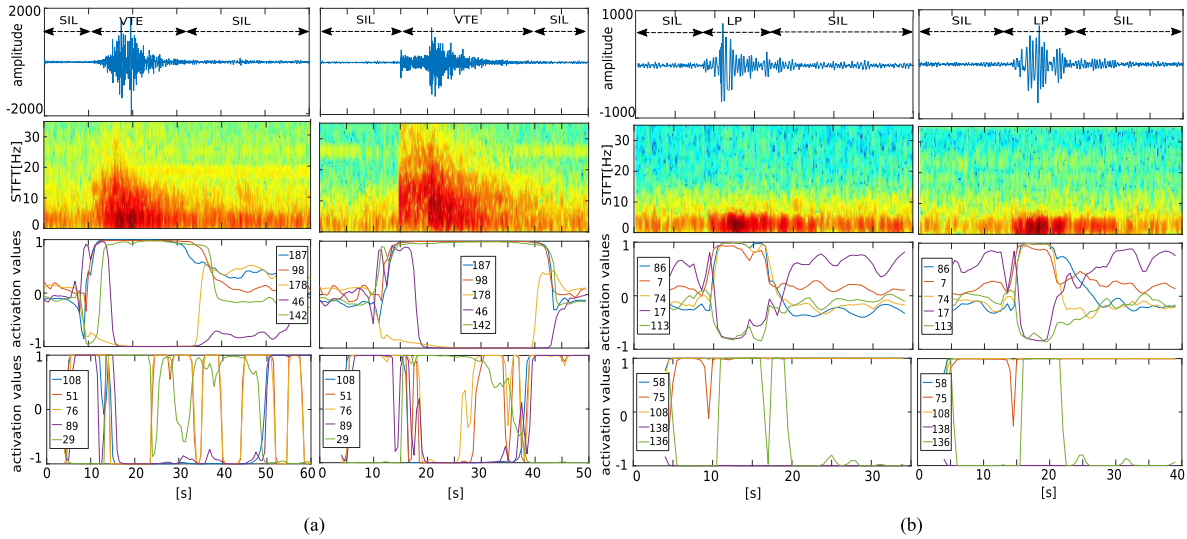

Fig. 8. Examples of time evolution of the neural activity of the five most specialized hidden units (see Table II) for the LSTM (third row) and Vanilla (fourth row) architectures, while recognizing two different VTE and LP events. Waveforms, spectrograms, and time evolution of the most active neurons are plotted for each event. While waveforms are plotted on a time sample basis, activation maps, and spectrograms use overlapped time windows containing several time samples to depict hidden units activity and frequency bands related to energy. This difference explains why event arrival and ending take place sooner in time in the activation maps and spectrograms, compared to the waveform plot.

TABLE II
MOST SPECIALIZED HIDDEN UNITS FOR VANILLA AND LSTM (OPTIMAL CONFIGURATION) USED TO RECOGNIZE THE FIVE TYPES OF EVENTS IN THE DATASET

	RNN-Vanilla	RNN-LSTM
VTE	108 51 76 89 29	187 98 178 46 142
LP	58 75 108 138 136	86 7 74 17 113
TRE	108 2 138 58 89	187 112 204 58 59
HYB	58 51 108 82 123	74 187 199 178 70
SIL	2 82 123 58 138	203 17 34 87 21

LP. This fact underlines the difficulty to describe the nature of these LP events that suffer a high intraclass variability in their spectral properties, due to the existence of several geophysical phenomena clustered into the LP class (e.g., lava-flow originated LPs, distant attenuated VTEs, and others) [55]–[58].

4) Similar to LP, SIL class has a high variability. Given that in manual labeling noises and even certain weak energy events are included as SIL, this type of event presents significant dispersion in its characteristics. Therefore, both architectures offer minimal specialization for it.

VI. SPECIALIZATION OF ARCHITECTURES

As described in Section V-A, the use of more sophisticated architectures leads to a higher specialization of neurons, improving the detection and classification results while decreasing the global neural activity concentrated in a few neurons with high activation values. In order to test the specialization capability of the architectures, this section will analyze how they work for each type of event and how gating mechanisms and memory cells help the hidden units to specialize in the recognition of different events.

Once the specialization of neurons is analyzed, we then analyze how less specialized neurons can affect the final performance of the system, concluding that they play an important role in the discrimination of the events against noise.

A. Specialization Analysis

This section will delve deeper into the study depicted in Figs. 6 and 7 where the average activation value of each hidden unit is represented. It will analyze how Vanilla and LSTM shape the degree of specialization of the hidden units. We will focus on the time evolution of neural activity of the most specialized hidden units when recognizing some of the most characteristic events in the dataset. This visualization approach (see Section IV) analyzes over time the activation values of the average five most active neurons when recognizing each type of event. Table II depicts these five most specialized neurons per event and per architecture used in the following graphs.

Fig. 8 shows the time evolution of neural activity of the five most specialized hidden units for LSTM and Vanilla recognizing two VTE and LP events. Each event is described by three subfigures: The first one showing the waveform, the second one showing the spectrogram and the third one showing the time evolution of neural activity of the most active neurons, where tanh activation values are in the range $[-1,1]$. As we can see, there are no neurons in common for these two types of events. Each architecture configures different class-oriented specializations for the different hidden units (see Table II).

Continuing the analysis of Fig. 8, LSTM exhibits a stable behavior. The neural activity evolves according to the spectral content. Generally, the hidden units remain in a state of semilethargy or low activation, becoming highly excited when the energy increases and returning to its nonactive state when the energy decreases.

Moreover, some hidden units have a very specific role that could be associated with specific characteristics of the signals.

1) In the third row of Fig. 8(a) where we analyze two different VTE events processed in the LSTM architecture, all neurons evolve according to the spectral contents of the event. However, neuron 46 in particular is devoted to changing its activation around the S-phase arrival. After that, the activity remains constant, changing again when the energy drops out.
2) In the third row of Fig. 8(b) where we analyze different LP events processed in the LSTM, neurons 7 and 17 show a slightly different behavior. While the level of energy is low, the unit exhibits moderate excitation. When the energy changes, the activity changes drastically, exhibiting high activity with an opposite sign. After that, the activity gradually returns to its initial state as the signal loses energy. If the energy changes during this period of time, the neuron interrupts the activity and changes the activation values again, suggesting that the neuron is specialized in short-term temporal changes.

In the case of Vanilla, in the fourth row of Fig. 8, hidden units exhibit a variable and polarized behavior. However, we can find some alignments between neural activity and spectral content.

1) In Fig. 8(a), which analyzes different VTE events, hidden units exhibit a variable behavior during a certain period of time, which corresponds to the duration of the seismic event.
2) Similarly, in Fig. 8(b) where different LP events are analyzed, hidden units regulate their activity when they detect the arrival of the seismic wave. From this moment on, the activity remains constant, unable to determine when the energy drops out. This would suggest that the Vanilla architecture is less susceptible to smooth changes in its input due to the lack of sophisticated memory mechanisms.

Based on the results obtained, we can conclude that memory mechanisms increase the specialization capacity of the architectures, improving the susceptibility and therefore decreasing the activation level of the hidden units. As a result, the models increase their generalization capabilities. They are also able to provide information on processes related to the event, such as changes in frequency, activity onsets and offsets, or phase arrivals.

B. Effect of Number of Neurons in Specialization

An important question related to artificial neural networks is how the number of neurons per hidden layer affects the specialization of the model. While relevant progress has been made in optimization and generalization tasks to avoid, respectively, overfitting and underfitting problems [59]–[61], to the best of authors' knowledge, there is no evidence of works exploring the relationship of the topology of the network and its ability to specialize.

Considering that the models tested in this article only have one hidden layer, in this section, we analyze how the use of a different number of hidden units in the LSTM architecture results in different degrees of specialization. Based on the conclusions of Section VI-A, we omit the analysis for the Vanilla architecture given its lower capability of specialization. We analyze how efficiently RNNs work using only the most specialized neurons for testing proposals.

TABLE III
CLASSIFICATION ACCURACY PERFORMANCE (%) OBTAINED BY BOTH ARCHITECTURES, LSTM AND VANILLA, VARYING THE NUMBER OF HIDDEN UNITS

	30	110	210	250	290
RNN-LSTM	82.18	82.90	84.88	84.43	83.19
RNN-Vanilla	81.03	80.96	79.45	79.14	81.15

Fig. 9 shows the time evolution of the five most active neurons of three different LSTM models recognizing two VTE and LP events, using 30 (top), 110 (middle), and 210 (bottom) hidden units, respectively. Following the same procedure as shown in Fig. 6, the number and activity level of the most active neurons for each model were obtained by computing their heat maps.

We observe that as the number of neurons in the hidden layer increases, a greater class-oriented specialization in the detection of events is observed. As mentioned above, neuron 46 in particular is devoted to changing its activation around the S-phase arrival, while the neurons 7 and 17 act as detectors of LP events. This observation is closely related to the conclusions drawn in [62], which confirms empirically that, if enough data are available as we increase the size of the network, the probability of optimal convergence of some of the connections also increases.

The use of configurations with a higher number of units increases the ability to specialize: like a human brain [63], the model tries to segment the information. When the number of hidden units available is high, the information is clustered by contents dealt with by small subsets of neurons. By training the model with a large enough database, each subset specializes in the recognition of an input according to its content, reducing the intersection between subsets of neurons [62]. In other words, each subset will be excited to a greater or lesser extent depending on the incoming information. This excitement will serve the model to weigh the membership of the incoming information to the different classes. Thus, classification or detection tasks are carried out based on the specialization criteria of each subgroup of neurons, increasing the final performance of the models (see Table III).

On the contrary, when the number of hidden units available is low, although the information is still clustered, the number of specialized neurons per subset is reduced.

In addition, it is important to note that the use of a very high number of units when large datasets are not available results in overfitting scenarios. Instead of obtaining useful discriminative information, units memorize specific information about training instances and, thus, decrease the generalization capability.

Finally, we analyze the role of the not-so-active neurons in the architectures, addressing their importance in the classification process. For this purpose, we compute the performance of the best model dropping out during the classification (not during training) of less specialized units for each type of event.

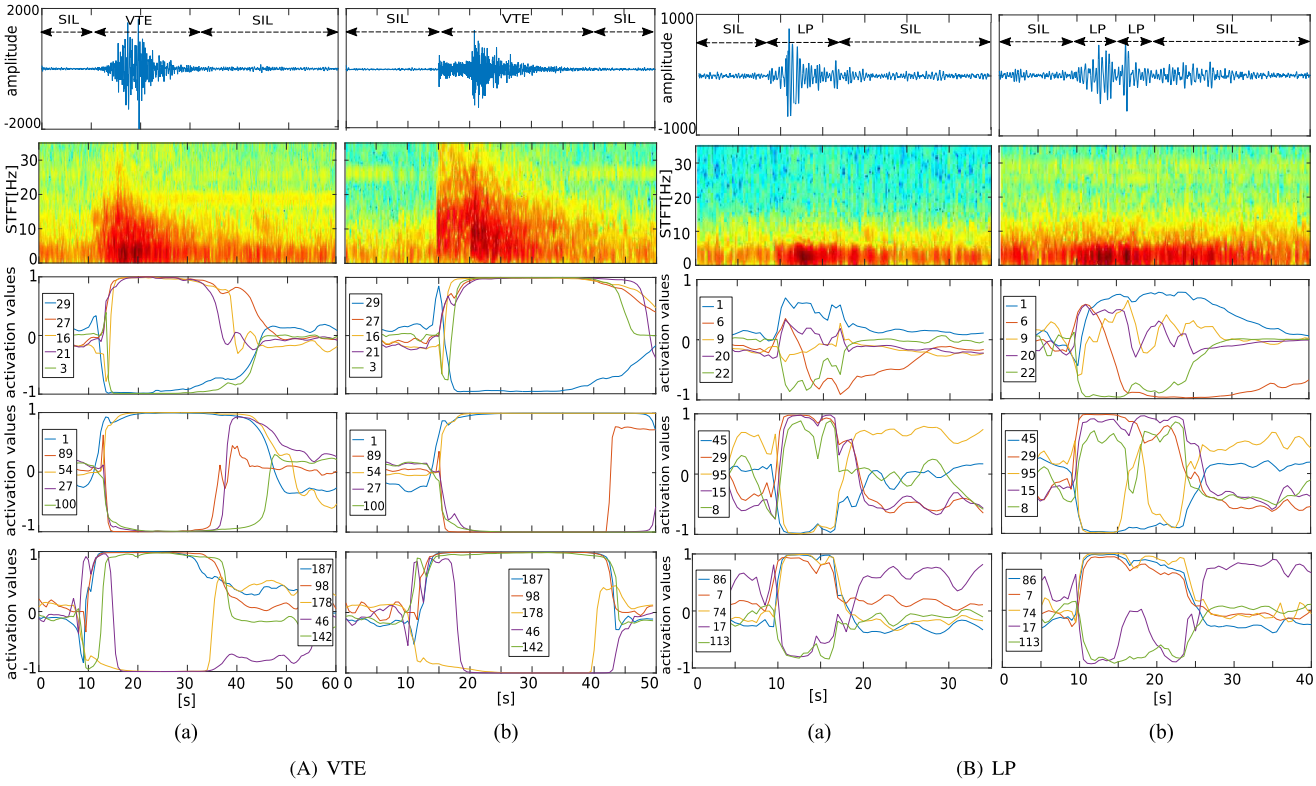

Fig. 9. Time evolution of neural activity of the five most specialized hidden units recognizing two examples of VTE events (left) and LP events (right). Three different configurations of the LSTM architecture are depicted, varying the number of hidden units: 30 (third row), 110 (fourth row), and 210 (fifth row) neurons, respectively. Waveforms, spectrograms and time evolution of the most active neurons are plotted for each event. While waveforms are plotted on a time sample basis, activation maps, and spectrograms use overlapped time windows containing several time samples to depict hidden unit activity and frequency bands related to energy. This difference explains why event arrival and ending take place sooner in time in the activation maps and spectrograms compared to the waveform plot.

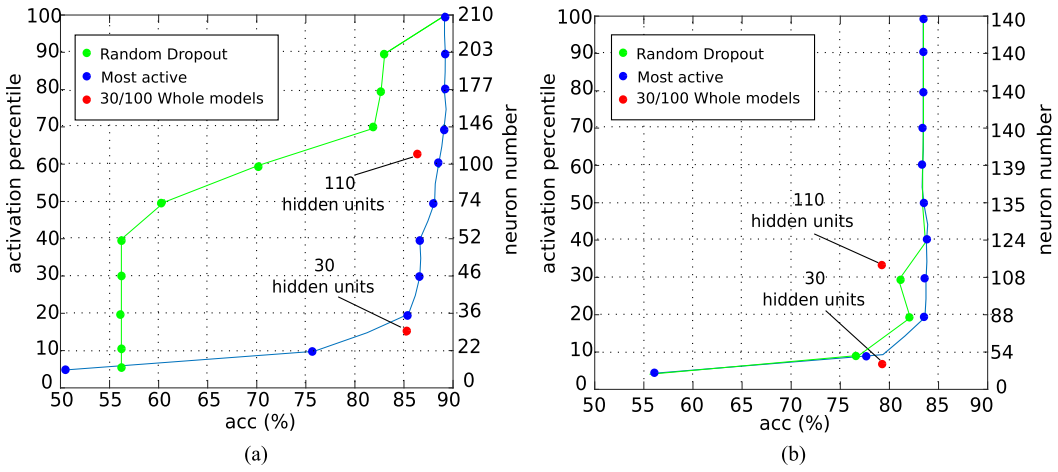

Fig. 10. Classification accuracy (%) of the LSTM architecture trained using up to 210 hidden units and the Vanilla architecture trained with up to 140 hidden units, tested without considering the less specialized units (blue line). Red points correspond to the performance obtained by two alternative LSTM and Vanilla architectures trained and tested using only 30 and 110 hidden units, respectively. The green lines correspond with the LSTM and Vanilla architectures tested applying a random dropout but keeping the number of active units in line with the percentiles of activation.

To test the role of specialized and nonspecialized neurons, Fig. 10 depicts the classification accuracy for both architectures using the most accurate topologies (210 and 140 hidden units for LSTM and Vanilla, respectively), but using a smaller number of neurons to classify events. The neurons used to perform classification are selected using their level of activation. The left-side Y-axis represents the percentile of activity of the neurons used to test the architectures. The right-side Y-axis represents the corresponding number of active neurons reaching every activation percentile measured on the left-side Y-axis. For example, the 100th percentile on the left-side Y-axis means all neurons (210 neurons as per the right-side Y-axis) are used to test events; the tenth percentile on the left-side Y-axis means that only 10% of the most active neurons, in other words, those neurons reaching 90% of the maximum activation value (22 as per the right-side Y-axis) are used to perform tests. The X-axis

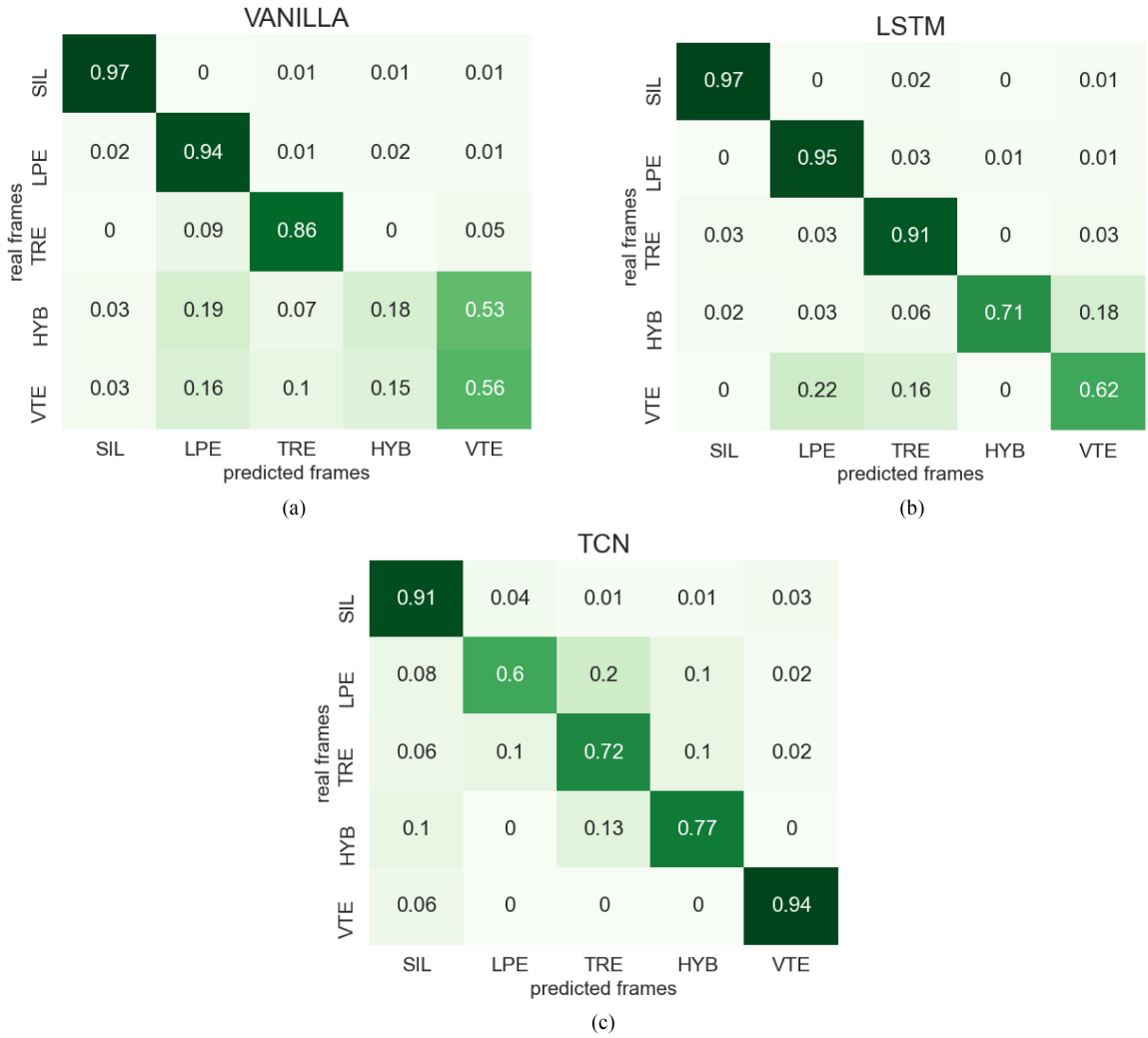

Fig. 11. Normalized confusion matrices related to the implemented architectures. The results are over the whole set of tests. (a) Vanilla. (b) LSTM. (c) TCN.

shows the classification results in terms of accuracy percentage. For comparison purposes, the red points correspond with the performance obtained by each architecture trained and tested with 30 and 110 hidden units, respectively, while the green lines correspond with classification accuracy achieved by both architectures applying a random dropout, keeping the number of active units in line with activation thresholds. Classification results are obtained with partition 1 of the cross-validation process described.

Fig. 10 shows that LSTM, a highly specialized model, obtains better results than Vanilla, a less specialized one, even when removing several units (blue lines versus red points): approximately 86% versus 84%, respectively. Vanilla, which is a lesser memory-based architecture, has most of its neurons highly polarized, as shown in Section V-A. This fact has a highly significant impact: approximately 80% of the neurons are always active, and therefore, even when lowering the activation threshold, the number of neurons remains constant (see number of neurons on the right side of the plot). This clearly shows almost null specialization, which is also previously shown in Section VI-A. Again, this observation confirms another conclusion empirically obtained in [62]: after training, only a subset of the connections, and subsequently a subset of the units, remain strong, while the others end up comparatively weak, allowing them to be dropped out without affecting the performance of the model. In other words, for testing proposals and considering specialized architectures, we can consistently reduce the starting network to between 10% and 20% of its original size while still achieving a high performance (approximately 88% accuracy). Another important conclusion is related to the random dropout. As shown in Fig. 10 (as per the green line), applying dropout to specialized-architectures trained without dropout has a greater impact. If specialized neurons (LSTM) are excluded from the classification process, the performance of the system is greatly affected. In contrast, when applying dropout to a less specialized architecture (Vanilla), the classification accuracy behaves almost similar to that of most active neurons. This conclusion invites us to consider the correlation between the usage of training dropout and the specialization of the models. That is to say, it might happen that using dropout in training would imply greater neuron-level generalization skills but lower specialization capability of those neurons. Exploring this idea is

beyond the scope of this article but will be addressed in future works.

VII. Conclusion

This article presents several visual approaches to understanding the learning mechanisms used by RNNs when processing VSRs. It investigates the hidden states activation maps of Vanilla and LSTM architectures in order to comprehend their internal functioning and decipher how they acquire knowledge. The experimental analysis has shown how neural activity changes depending on the architecture and events. Second, this article has analyzed the degree of specialization of the hidden units for both RNN architectures and has found a disparity in their neural activity as well as in the amount of neurons that each architecture specializes to detect and classify the different types of events.

Once the most specialized neurons have been identified, we have analyzed the evolution of their activation values over time for several seismic events. The results obtained show that sophisticated architectures like the LSTM have a more stable behavior (according to the spectral evolution of the seismic events) compared to the more elementary Vanilla. We have found that neuron specialization permits interesting applications for volcanic seismicity analysis, based on: their capacity to activate in the presence of different seismic events, on their sensitivity to activity in different frequency bands, or on their reactivity to certain phase arrivals. Through these visualization approaches RNNs can provide extra knowledge in addition to the expected classification output.

Finally, we have analyzed the effect of the size of the network on the degree of specialization of the neurons, concluding that the use of a higher number of hidden units allows models to create class-oriented specializations based on specific neurons. Depending on the degree of specialization achieved by the models, we found that 20 or 25% of the less specialized neurons can be omitted at test time without affecting the performance obtained. This fact relates the explainability of RNNs with the field of deep network sparsity and active learning, opening a wide range of possible future lines of research.

Acknowledgment

The authors would like to thank the Instituto Andaluz de Geofísica for providing us with the Decepction Island dataset and invaluable geophysical insight.

Appendix

Confusion Matrices

See Fig. 11 in previous page.

References

[1] A. B. Arrienta *et al.*, "Explainable artificial intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI," *Inf. Fusion*, vol. 58, pp. 82–115, 2020, doi: 10.1016/j.inffus.2019.12.012.
[2] M. Bishop, *Pattern Recognition and Machine Learning*. New York, NY: Springer, vol. 4, no. 4, 2006, p. 738.
[3] K. P. Murphy, *Machine Learning: A Probabilistic Perspective*. Cambridge, MA, USA: MIT Press.
[4] M. Jordan and T. Mitchell, "Machine learning: Trends, perspectives and prospects," *Science*, vol. 249, no. 6245, pp. 255–260, 2016.
[5] I. G. Y. Bengio and A. Courville, *Deep Learning*, vol. 1. Cambridge, MA, USA: MIT Press, 2016.
[6] E. D. Liddy, "Natural language processing," in *Encycl. Libr. Inf. Sci.*, 2nd Ed. New York, NY, USA: Marcel Decker, Inc. 2001.
[7] W. Medhat, A. Hassan, and H. Korashy, "Sentiment analysis algorithms and applications: A survey," *Ain Shams Eng. J.*, vol. 5, no. 4, pp. 1093–1113, 2014.
[8] B. Liu *et al.*, "Sentiment analysis and subjectivity," *Handbook Natural Lang. Process.*, vol. 2, pp. 627–666, 2010.
[9] F. Camastra and A. Vinciarelli, *Machine Learning for Audio, Image and Video Analysis: Theory and Applications*. Berlin, Germany: Springer, 2015.
[10] C. Olah, "Visualizing representations: Deep learning and human beings," vol. 22, p. 2018. Accessed: May 2015. [Online]. Available: colah.github.io/posts/2015-01-Visualizing-Representations/
[11] J. Yosinski *et al.*, "Understanding neural networks through deep visualization," 2015, *arXiv:1506.06579*.
[12] S. Carter *et al.*, "Using artificial intelligence to augment human intelligence," *Distill*, vol. 2, no. 12, p. e9, 2017, doi: 10.23915/distill.00009.
[13] H. Strobelt, S. Gehrmann, H. Pfister, and A. M. Rush, "LSTMVis: A tool for visual analysis of hidden state dynamics in recurrent neural networks," *IEEE Trans. Visual. Comput. Graph.*, vol. 24, no. 1, pp. 667–676, Jan. 2018.
[14] K. Simonyan *et al.*, "Deep inside convolutional networks: Visualising image classification models and saliency maps," in *Workshop Int. Conf. Learn. Represent.*, 2014.
[15] R. Zhao *et al.*, "Saliency detection by multi-context deep learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 1265–1274.
[16] G. Li and Y. Yu, "Visual saliency based on multiscale deep features," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 5455–5463.
[17] S. Arif *et al.*, "3D-CNN-based fused feature maps with LSTM applied to action recognition," *Future Internet*, vol. 11, no. 2, pp. 42–59, 2019.
[18] L. Zhaoxin and M. Zhu, "Recurrent neural networks with mixed hierarchical structures for natural language processing," in *Proc. Int. Joint Conf. Neural Netw.*, 2021, pp. 1–8.
[19] M. Thomas and C. Latha, "Sentimental analysis of transliterated text in malayalam using recurrent neural networks," *J. Ambient Intell. Humanized Comput.*, vol. 12, no. 6, pp. 6773–6780, 2021.
[20] J. Lin, Y. Shao, Y. Djenouri, and U. Yun, "ASRNN: A recurrent neural network with an attention model for sequence labeling," *Knowl.-Based Syst.*, vol. 212, 2021, Art. no. 106548.
[21] M. Pielka *et al.*, "Tackling contradiction detection in german using machine translation and end-to-end recurrent neural networks," in *Proc. 25th Int. Conf. Pattern Recognit.*, 2021, pp. 6696–6701.
[22] J. Wang and L. Raymond, "Chaotic recurrent neural networks for financial forecast," *Amer. J. Neural Netw. Appl.*, vol. 7, no. 1, pp. 7–14, 2021.
[23] G. D'Angelo and F. Palmieri, "Network traffic classification using deep convolutional recurrent autoencoder neural networks for spatial-temporal features extraction," *J. Netw. Comput. Appl.*, vol. 173, 2021, Art. no. 102890.
[24] R. Tolosana, R. Vera-Rodriguez, J. Fierrez, and J. Ortega-Garcia, "Deepsign: Deep on-line signature verification," *IEEE Trans. Biometrics, Behav., Identity Sci.*, vol. 3, no. 2, pp. 229–239, Apr. 2021.
[25] A. Moghar and M. Hamiche, "Stock market prediction using LSTM recurrent neural network," *Procedia Comput. Sci.*, vol. 170, pp. 1168–1173, 2020.
[26] Y. Ming *et al.*, "Understanding hidden memories of recurrent neural networks," in *Proc. IEEE Conf. Vis. Analytics Sci. Technol.*, 2017, pp. 13–24.
[27] H. Strobelt, S. Gehrmann, M. Behrisch, A. Perer, H. Pfister, and A. M. Rush, "Seq2seq-vis: A visual debugging tool for sequence-to-sequence models," *IEEE Trans. Visual. Comput. Graph.*, vol. 25, no. 1, pp. 353–363, Jan. 2019.
[28] B. Kwon *et al.*, "Retainvis: Visual analytics with interpretable and interactive recurrent neural networks on electronic medical records," *IEEE Trans. Visual. Comput. Graph.*, vol. 25, no. 1, pp. 299–309, Jan. 2019.
[29] J. Palmer, "The new science of volcanoes harnesses ai, satellites and gas sensors to forecast eruptions," *Nature*, vol. 581, no. 7808, pp. 256–259, 2020.
[30] B. Chouet, "Dynamics of a fluid-driven crack in three dimensions by the finite difference method," *J. Geophys. Res. Sol. Earth*, vol. 91, no. B14, pp. 13967–13992, 1986.
[31] G. Alguacil *et al.*, "Observations of volcanic earthquakes and tremor at deception Island-Antarctica," *Ann. Geophys.*, vol. 42, no. 3, pp. 417–436, 1999.
[32] J. Ibañez. Abril *et al.*, "The recent seismo-volcanic activity at deception island volcano," *Deep Sea Res. II, Topical Stud. Oceanogr.*, vol. 50, no. 10, pp. 1611–1629, 2003.

[33] C. Benítez, J. Ramírez, J. Ibáñez, J. Almendros, A. García-Yeguas, and G. Cortés, "Continuous HMM-based seismic event classification at deception Island, Antartica," *IEEE Trans. Geosci. Remote Sens.*, vol. 45, no. 1, pp. 138–147, Jan. 2007.
[34] M. Titos, A. Bueno, L. García, M. C. Benítez, and J. Ibañez, "Detection and classification of continuous volcano-seismic signals with recurrent neural networks," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 4, pp. 1936–1948, Apr. 2019.
[35] A. B. Rodriguez, C. Benítez, L. Zuccarello, S. D. Angelis, and J. Ibañez, "Bayesian monitoring of seismo-volcanic dynamics," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–14, 2022, Art. no. 5903414, doi: 10.1109/TGRS.2021.3076012.
[36] P. Jiao and A. Alavi, "Artificial intelligence in seismology: Advent, performance and future trend," *Geosci. Front.*, vol. 11, pp. 739–744, 2020.
[37] Q. Kong, D. T. Trugman, Z. E. Ross, M. J. Bianco, B. J. Meade, and P. Gerstoft, "Machine learning: Trends, perspectives and prospects," *Seismological Res. Lett.*, vol. 90, no. 1, pp. 3–14, 2019.
[38] H. Hewamalage, C. Bergmeier, and K. Bandara, "Recurrent neural networks for time series forecasting: Current status and future directions," *Int. J. Forecasting*, vol. 37, no. 1, pp. 388–427, 2021.
[39] L. W. R. R. A. Z. J. Yan and L. Mu, "Temporal convolutional networks for the advance prediction of ENSO," *Sci. Rep.*, vol. 10, no. 1, pp. 1–15, 2020.
[40] S. Bai, K. Kolter, and V. Koltun, "An empirical evaluation of generic convolutional and recurrent networks for sequence modeling," 2018, *arXiv:1803.01271*.
[41] A. Graves, A. Mohamed, and G. Hinton, "Speech recognition with deep recurrent neural networks" in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2013, pp. 6645–6649.
[42] J. Schmidhuber, "Deep learning in neural networks: An overview," *Neural Netw.*, vol. 61, pp. 85–117, 2015.
[43] R. Pascanu *et al.*, "On the difficulty of training recurrent neural networks," in *Int. Conf. Mach. Learn.*, 2013, pp. 1310–1318.
[44] K. Kumar, C. Kim, and R. M. Stern, "Delta-spectral cepstral coefficients for robust speech recognition," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2011, pp. 4784–4787.
[45] J. M. Ibáñez *et al.*, "Seismovolcanic signals at deception island volcano, antarctica: Wave field analysis and source modeling," *J. Geophys. Res., Solid Earth*, vol. 105, no. B6, pp. 13905–13931, 2000.
[46] J. L. Smellie, "Recent observations on the volcanic history of deception Island, South Shetland Islands," *Brit. Antarctic Surv. Bull.*, no. 81, pp. 83–85, 1988.
[47] E. Carmona, J. Almendros, I. Serrano, D. Stich, and J. M. Ibáñez, "Results of seismic monitoring surveys of deception Island Volcano, Antarctica, from 1999–2011," *Antarctic Sci.*, vol. 24, no. 5, pp. 485–499, 2012.
[48] A. Sakakibara and Y. Hatanaka, "Neuronal polarization in the developing cerebral cortex," *Front. Neurosci.*, vol. 9, 2015, Art. no. 116.
[49] B. Zhou, Y. Sun, D. Bau, and A. Torralba, "Revisiting the Importance of individual units in CNNs via ablation," 2018, *arXiv:1806.02891*.
[50] M. Leavitt, "Selectivity considered harmful: Evaluating the causal impact of class sensitivity in DNNs," 2020, *arXiv:2003.01262*.
[51] S. Park, J. Park, S.-J. Shin, and I.-C. Moon, "Adversarial dropput for supervised and semi-supervised learning," in *Proc. 32nd AAAAI Conf. Artif. Intell.*, 2018, pp. 3917–3924.
[52] T. Hoefler, D. Alistarh, T. Ben-Nun, N. Dryden, and A. Peste, "Sparsity in deep learning: Pruning and growth for efficient inference and training in neural networks," *J. Mach. Learn. Res.*, vol. 22, no. 241, pp. 1–124. 2021.
[53] T. Zhuang, Z. Zhang, Y. Huang, X. Zeng, K. Shuang, and X. Li, "Neuron-level structured pruning using polarization regularizer," in *Proc. 34th Conf. Neural Inf. Process.*, 2020, vol. 33, pp. 9865–9877.
[54] W. Pei, H. Dibeklioglu, T. Baltrusaitis, and D. M. Tax, "Attended end-to-end architecture for age estimation from facial expression videos," *IEEE Trans. Image Process.*, vol. 29, pp. 1972–1984, 2020, doi: 10.1109/TIP.2019.2948288.
[55] E. Del Pezzo, M. Simini, and J. Ibanez, "Separation of intrinsic and scattering Q for volcanic areas: A comparison between Etna and Campi Flegrei," *J. Volcanol. Geothermal Res.*, vol. 70, no. 3/4, pp. 213–219, 1996.
[56] E. Del Pezzo, M. La Rocca, and J. Ibanez, "Observations of high-frequency scattered waves using dense arrays at teide volcano," *Bull. Seismological Soc. Amer.*, vol. 87, no. 6, pp. 1637–1647, 1997.
[57] C. Martinez-Arevalo, F. Bianco, J. Ibañez, and E. D. Pezzo, "Shallow seismic attenuation and shear-wave splitting in the short period range of deception Island Volcano (Antarctica)," *J. Volcanol. Geothermal Res.*, vol. 128, no. 1, pp. 89–113, 2003.
[58] A. Moreno-Vacas and J. Almendros, "On the origin of recent seismic unrest episodes at deception Island Volcano, Antarctica," *J. Volcanol. Geothermal Res.*, vol. 419, 2021, Art. no. 107376.
[59] L. Deng and D. Yu, "Deep learning: Methods and applications," *Foundations Trends Signal Process.*, vol. 7, no. 3/4, pp. 197–387, 2014.
[60] S. Salman and X. Liu, "Overfitting mechanism and avoidance in deep neural networks," 2019, *arXiv:1901.06566*.
[61] R.-Y. Sun, "Optimization for deep learning: An overview," *J. Operations Res. Soc. China*, vol. 8, no. 2, pp. 249–294, 2020.
[62] J. Frankle *et al.*, "The lottery ticket hypothesis: Finding sparse, trainable neural networks," 2018, *arXiv:1803.03635*.
[63] C. A. Kurby and J. M. Zacks, "Segmentation in the perception and memory of events," *Trends Cogn. Sci.*, vol. 12, no. 2, pp. 72–79, 2008.

Manuel Titos received the M.Sc. degree in computer engineering, the master's degree in computer and network engineering, and the Ph.D. degree in information technology and communications from the University of Granada, Granada, Spain, in 2012, 2013, and 2018, respectively.

He is currently working in the field of computational volcanology, where high-performance computing (HPC) is being used in the framework of Probabilistic Volcanic Hazard Assessment (PVHA) to model ash dispersal and airborne tephra concentration at different flight levels. He has also worked on the development of advanced signal processing algorithms for description and characterization of seismo-volcanic signals. His main research interests include computational volcanology, volcanic hazard assessment, deep learning techniques (machine learning), and computational intelligence in remote sensing signals, particularly in the area of volcanic eruption early warning systems. He is also interested in data analysis and high-resolution image processing.

Luz García received the M.Sc. degree in telecommunication engineering from the Polytechnic University of Madrid, Madrid, Spain, in 2000 and the Ph.D. degree in telecommunication engineering from the University of Granada, Granada, Spain, in 2008.

After working as a Support Engineer for Communication Networks with Ericsson-Spain, Madrid, Spain, for five years, she joined a European Research Project with the University of Granada. She was an Assistant Professor with the Department of Signal Theory, Telematics and Communications, University of Granada, where she has been an Associate Professor, since 2005. Her research interests are signal processing, pattern recognition, and machine learning in the fields of speech and geophysics.

Milad Kowsari received the M.S. degree in earthquake engineering from the University of Kurdistan, Sanandaj, Iran, in 2012 and the Ph.D. degree in civil engineering from the University of Iceland, Reykjavik, Iceland, in 2019.

He is currently a Postdoctoral Researcher Associate with the Faculty of Civil and Environmental Engineering, University of Iceland. The principal focus of his research is probabilistic seismic hazard assessment, earthquake ground motion modeling, engineering seismology, and data analysis, in which he has authored or coauthored around 40 papers in ISI accredited scientific journals, international, and national conferences proceedings.

Dr. Kowsari was the recipient of the Elite Prize in research at the University of Kurdistan.

Carmen Benítez received the M.Sc. and Ph.D. degrees in physics from the University of Granada, Granada, Spain, in 1991 and 1998, respectively.

She was a Visiting Researcher with the International Computer Science Institute, Berkeley, CA, USA, and United States Geological Survey (USGS), Menlo Park, CA, USA. From 1990 to 2004, she was with the Department of Electronics and Computer Sciences, Faculty of Science, University of Granada, where she has been with the Department of Signal Theory, Telematics and Communications, Escuela Técnica Superior (ETS) of Computer and Telecommunication Engineering, since 2004. She was an Associate Professor with the University of Granada from 2003 to 2018, and from 2015 to 2019, the Head of the Department of Signal Theory, Telematics and Communications, where she is currently a Full Professor. Her research interests include signal processing, computational geophysics, speech recognition, machine learning, and pattern recognition.