An End-to-End Earthquake Monitoring Method for Joint Earthquake Detection and Association using Deep Learning

Weiqiang Zhu∗
Department of Geophysics
Stanford University
Stanford, CA, 94305
zhuwq@stanford.edu

Kai Sheng Tai∗
Computer Science Department
Stanford University
Stanford, CA, 94305
kst@cs.stanford.edu

S. Mostafa Mousavi
Department of Geophysics
Stanford University
Stanford, CA, 94305
mmousavi@stanford.edu

Peter Bailis
Computer Science Department
Stanford University
Stanford, CA, 94305
pbailis@cs.stanford.edu

Gregory C. Beroza
Department of Geophysics
Stanford University
Stanford, CA, 94305
beroza@stanford.edu

Abstract

Earthquake monitoring through seismometer networks typically involves a pipeline consisting of detection, phase picking, association, and localization stages. We introduce an earthquake detection and localization method based on a novel end-to-end deep neural network architecture that maps collections of raw seismic waveforms to proposed event times and epicenter locations. Unlike traditional approaches to this task, our method does not rely on hand-designed time series features or rules for combining predictions across multiple stations. We evaluate our proposed method on data from the 2019 Ridgecrest earthquake sequence, demonstrating its effectiveness when compared with four state-of-the-art earthquake catalogs.

1 Introduction

Earthquakes are routinely monitored by local and global seismic networks, which consist of tens to thousands of seismometers that continuously record ground shaking. Most earthquake monitoring systems detect earthquakes by two stages: seismic phase detection and phase association (Figure 1). First, a phase detection algorithm identifies candidate earthquake signals independently at each station; then, an association algorithm combines these candidates by checking if the times are consistent with the travel-times from a common earthquake epicenter. This two-stage approach is robust and effective in most cases. However, a disadvantage of this two-stage approach is that it relies on accurate detection at a single station – a difficult task for low signal-to-noise ratio arrivals from small earthquakes whose numbers dominate the catalog. The association stage typically does not exploit potentially informative waveform features across stations. An alternative approach is to develop array-based earthquake detection, which could improve the sensitivity for events that are too weak to be detected reliably by a single station.

We propose an end-to-end earthquake monitoring method that combines the detection and association stages within an end-to-end deep neural network architecture. Our architecture first extracts

∗The two authors contributed equally to this paper.

AI for Earth Sciences Workshop at NeurIPS 2020.
Figure 1: Schematic of earthquake monitoring through seismometer networks: (a) continuously recorded seismic data; (b) the monitoring tasks include: detecting earthquake signals, associating signals from a common earthquake, and (c) estimating earthquake times and locations.

features from the seismic waveforms recorded at each station in the network, and then aggregates the resulting feature vectors using a pre-specified velocity model for wave propagation in the region. A classification network then processes these aggregated features to identify candidate events over a spatiotemporal grid. A distinguishing characteristic of our approach in comparison with traditional methods is that we jointly optimize the parameters of the feature extraction and classification networks to maximize detection accuracy over a training dataset of seismic waveforms and ground-truth events, thus avoiding the need for hand-designed features and association rules for combining detections across multiple stations.

2 Related Work

Station-based earthquake detection and phase picking Current earthquake detection and seismic phase picking methods are mainly developed for a single station by designing features, such as changes of amplitude and frequency content, to detect the arrival of seismic waves [1, 2, 3, 4, 5, 6, 7]. Recently, deep learning has emerged as an effective method for earthquake detection and phase picking [8, 9, 10, 11, 12, 13, 14, 15]. However, these methods only use single-station information and must be subsequently aggregated with an association step for application to seismic networks.

Seismic phase association Association methods aim to integrate detections from all seismometers in a network to determine if these detections come from a true earthquake event by imposing a physical constraint on the travel-times of seismic phases, based on earthquake location, station locations, and a wavespeed model for the Earth [16, 17, 18, 19, 20]. Recently, deep-learning-based association methods have also been proposed based on learned phase arrival-time patterns [21] or waveform similarity [22, 23].

Array-based earthquake detection and location An alternative approach to the two-stage method is to detect and locate earthquakes directly while considering multiple waveforms recorded across a seismic array/network. Methods like multi-station template matching [24, 25, 26] and shifting-and-stacking [27, 28, 29] explore the coherent waveform signals between events and stations and combine multiple waveforms to enhance detection sensitivity.

Object detection Earthquake monitoring bears some similarity to object detection in computer vision, which aims to locate and classify objects in an image [30, 31, 32, 33, 34]. Modern CNN-based object detection methods such as YOLO [34] scan a set of coarse grids to classify object categories and predict bounding boxes. Similarly, earthquake monitoring aims to detect an earthquake by determining its time and location, so we scan a 4D spatial-temporal volume to detect and locate earthquakes in our end-to-end model.

3 Method

Model Architecture Our end-to-end model architecture (Figure 2) consists of three primary components: (1) a feature extraction network, (2) a time shifting module, and (3) a feature aggregation network. The input to the feature extraction network is a time series of seismometer measurements at a monitoring station. We first convert the raw waveform to a short-time Fourier transform representation. The feature extraction network, parameterized as a modified Wide Residual Network [35], then maps this 2D signal representation to a sequence of feature representations. We use the same
feature extraction network for each station in the network; this allows for our approach to adapt to changes in the number of stations in the network. The time shifting module takes as input a candidate hypocenter location and the features extracted in the previous stage. Its output is an aligned view of the features, where the time series from each station is shifted by the theoretical travel time of the seismic wave from the candidate hypocenter to the station. This travel time is computed using a pre-specified velocity model, thus allowing practitioners to incorporate prior knowledge on physical constraints into the pipeline. Intuitively, high-activation features from multiple stations should coincide at the true hypocenter of an earthquake, thus allowing the network to localize events and to avoid false detections due to local noise at each station. Finally, the feature aggregation network combines these shifted features and classifies whether or not an earthquake exists at each candidate location and each time point. We describe the network architecture in more detail in Table A1.

**Training and inference** The feature extraction and feature aggregation steps are the same for both training and inference, while the input data length and the sampling strategy of the time shifting module is different. During training, we sample one earthquake from the SCSN catalog and cut a time window containing the corresponding seismic signals as input data. The time shifting module selects the cataloged earthquake location and 5 other random locations (negative samples) to generate 6 sets of shifted features for training. This negative sampling process helps balance the sparse true earthquake locations relative to the candidate locations in the entire 3D space and speeds up the training. During inference, we take the continuous seismic waveforms \(N_t\) as input data. The time shifting module uniformly samples the whole space at \(N_x \times N_y\) grid points with a horizontal interval of \(\sim 2\) km. From the spatial-temporal predictions above a threshold, we first extract earthquake times from peaks along the time axis (Figure A1c) and then determine the earthquake location using the geometric median of the top 20 activated grids (Figure A1d).

### 4 Evaluation

**Data** Two large earthquakes of magnitude 6.4 and 7.1 shook the Ridgecrest area in July, 2019. These earthquakes triggered a large sequence of aftershocks, which provide a good dataset for training and evaluating our model. From the Southern California Seismic Network (SCSN) earthquake catalog, we collected 42,660 reported earthquakes (black dots in Figure A1a) from Apr 1st, 2019 to Dec 31st, 2019 and downloaded the continuous seismic waveforms from the seismic stations within \(\sim 100\) km (blue triangles in Figure A1a). We selected continuous waveforms of July 7th with 1,226 earthquakes as the validation dataset and the continuous waveforms for July 5th and 6th with 4,769 earthquakes as the test dataset. Earthquakes were most frequent during these three days, making the detection task challenging. The remaining data are used for training our end-to-end model.

**Benchmarks** The Ridgecrest earthquake sequence has attracted a great deal of research attention. In addition to the SCSN catalog, Liu et al. [36] built a catalog using the deep-learning-based PhaseNet [11] picker and REAL [19] association. Shelly [37] and Ross et al. [38] built two different catalogs using the template matching method, which takes earthquakes in the SCSN catalog as tem-
Figure 3: Performance on the test dataset: (a) precision-recall curves compared with four different catalogs; (b) estimated earthquake locations; (c) error distribution of earthquake time; (d) error distribution of earthquake location compared with the SCSN catalog.

Table 1: Performance of the end-to-end method compared with state-of-art catalogs.

<table>
<thead>
<tr>
<th>Comparing catalog</th>
<th>Precision</th>
<th>Recall</th>
<th>F1</th>
</tr>
</thead>
<tbody>
<tr>
<td>SCSN</td>
<td>0.620</td>
<td>0.807</td>
<td>0.701</td>
</tr>
<tr>
<td>Ross et al. (2019) [38]</td>
<td>0.849</td>
<td>0.443</td>
<td>0.582</td>
</tr>
<tr>
<td>Shelly (2020) [37]</td>
<td>0.775</td>
<td>0.726</td>
<td>0.750</td>
</tr>
<tr>
<td>Liu et al. (2020) [36]</td>
<td>0.793</td>
<td>0.824</td>
<td>0.808</td>
</tr>
</tbody>
</table>

Table 2: Performance comparison assuming Shelly [37]'s catalog as ground truth.

<table>
<thead>
<tr>
<th>Catalog</th>
<th>Precision</th>
<th>Recall</th>
<th>F1</th>
</tr>
</thead>
<tbody>
<tr>
<td>SCSN</td>
<td>0.834</td>
<td>0.598</td>
<td>0.696</td>
</tr>
<tr>
<td>Ross et al. (2019) [38]</td>
<td>0.489</td>
<td>0.877</td>
<td>0.628</td>
</tr>
<tr>
<td>Liu et al. (2020) [36]</td>
<td>0.836</td>
<td>0.756</td>
<td>0.794</td>
</tr>
<tr>
<td>End-to-end method</td>
<td>0.775</td>
<td>0.726</td>
<td>0.750</td>
</tr>
</tbody>
</table>

Plates and scans the continuous seismic data to detect more small earthquakes. The SCSN catalog reports 4,769 earthquakes, Liu et al. [36] detect 5,536 earthquakes, Shelly [37] detects 6,702 earthquakes, and Ross et al. [38] detect 11,224 earthquakes. These state-of-art catalogs provide useful benchmarks for evaluating the performance of our end-to-end method.

Results. Figure 3a shows the precision and recall curves compared with the four catalogs on the test data between July 5th – 6th. Our end-to-end model detects 5,807 earthquakes with a threshold of 0.2. We consider the earthquakes within 5s from the catalog time as true positives and report the corresponding precision and recall values in Table 1. Most earthquakes of these catalogs (except Ross et al. [38]) can be detected by our method. Assuming Shelly [37]'s catalog as ground truth, our end-to-end method achieves similar performance as the other state-of-art catalogs (Table 2). The error distributions of earthquake time and location compared with the SCSN catalog are shown in Figure 3c and d. Most detected earthquakes have a time difference within 1 s and an epicenter difference within 5 km. We also observe that the earthquake spatial distribution is similar to the SCSN catalog in Figure 3b.

5 Conclusions

Deep learning is an effective approach for earthquake detection, however, current methods that process a single seismic waveform as an isolated 1D time series may not realize its full potential. We developed an end-to-end array-based method that combines detection and association. Our method simultaneously trains two neural networks – for feature extraction and aggregation – so that the model can process multiple waveforms from a seismic network to improve detection sensitivity and reduce false positives. The evaluation results on a temporally dense earthquake sequence demonstrates that our end-to-end model can effectively detect and locate earthquakes.
Acknowledgements

Waveform data, metadata, or data products for this study were accessed through the Southern California Earthquake Data Center (SCEDC). This work was supported by the Department of Energy (Basic Energy Sciences; Award DE-SC0020445).

Appendix

The parameters of the feature extraction and feature aggregation networks are shown in Table A1. We use the AdamW [39] optimizer with a learning rate of $3 \times 10^{-4}$ and a weight decay rate of $5 \times 10^{-4}$ for training the two networks. We also add a cosine learning rate decay strategy [40] for training with a total 12.5 million repeatedly sampled earthquakes. The training dataset and an earthquake example is shown in Figure A1.

Table A1: The network parameters of the end-to-end model in Figure 2

<table>
<thead>
<tr>
<th>Module</th>
<th>Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>Feature Extraction</td>
<td>Layer 1: Short-time Fourier transform</td>
</tr>
<tr>
<td></td>
<td>Layer 2: 2D convolution $[k=3 \times 3, c=16]$ with stride $[1 \times 1]$</td>
</tr>
<tr>
<td></td>
<td>+ batch normalization + leaky relu</td>
</tr>
<tr>
<td></td>
<td>Layer 3: Residual block: 2D convolution $[k=3 \times 3, c=32]$ with stride $[2 \times 1]$</td>
</tr>
<tr>
<td></td>
<td>+ batch normalization + leaky relu</td>
</tr>
<tr>
<td></td>
<td>Layer 4: Residual block: 2D convolution $[k=3 \times 3, c=64]$ with stride $[2 \times 1]$</td>
</tr>
<tr>
<td></td>
<td>+ batch normalization + leaky relu</td>
</tr>
<tr>
<td></td>
<td>Layer 5: Residual block: 2D convolution $[k=3 \times 3, c=128]$ with stride $[2 \times 1]$</td>
</tr>
<tr>
<td></td>
<td>+ batch normalization + leaky relu</td>
</tr>
<tr>
<td></td>
<td>Layer 6: Fully connected layer + leaky relu</td>
</tr>
<tr>
<td>Feature Aggregation</td>
<td>Layer 1: 1D convolution $[k=3, c=128]$ + batch normalization + leaky relu</td>
</tr>
<tr>
<td></td>
<td>Layer 2: 1D convolution $[k=3, c=64]$ + batch normalization + leaky relu</td>
</tr>
<tr>
<td></td>
<td>Layer 3: 1D convolution $[k=3, c=1]$ + batch normalization + sigmoid</td>
</tr>
</tbody>
</table>

Figure A1: Dataset and examples: (a) The 2019 Ridgecrest earthquake sequence (black dots) between Apr 1, 2019 and Dec 31, 2019 and nearby seismic stations (blue triangles) from the Southern California Seismic Network (SCSN). (b) One example of earthquake waveforms recorded by these stations. (c) The aggregated prediction along the time axis from which we extract the earthquake time. (d) The aggregated prediction along spatial axes from which we determine the earthquake epicenter. (e) Waveforms re-ordered based on the earthquake epicenter distance, which shows the physical relationship between the epicenter distance and the arrival time and amplitude recorded at each seismic station.
References


