

Detecting Earthquakes in Noisy Real-Time GNSS Data With Deep Learning for Improved PGD Magnitude Estimation

Sydney N. Dybing^{*1,2}, Diego Melgar¹, Amanda M. Thomas³, Dara E. Goldberg⁴, David Mencin⁵, and Brendan W. Crowell⁶

ABSTRACT

To disseminate accurate and useful warnings, earthquake early warning (EEW) systems must quickly determine the size and location of an earthquake to estimate expected shaking. Traditional seismic-based algorithms tend to underestimate the true magnitudes of large earthquakes, a phenomenon known as magnitude saturation. This limitation motivated the recent inclusion of Global Navigation Satellite Systems (GNSS) data into the U.S. Geological Survey's ShakeAlert EEW system with the Geodetic First Approximation of Size and Time (GFAST) algorithm because GNSS data do not saturate with large ground motions. However, the noise levels of GNSS data are very high compared with traditional seismic data, which obscures *P*-wave arrivals and can result in less accurate magnitude estimations if displacement amplitudes are low, such as for lower magnitude earthquakes or large source–station distances. In this study, we develop a deep-learning model that detects earthquakes in GNSS data and use the Ridgecrest, California, earthquake sequence as a case study to demonstrate how the model could act as a filter to reduce the amount of low-quality data that enters an algorithm like GFAST. To preserve our limited real earthquake data for model inference, we generated a training dataset composed of > 700,000 synthetic displacement waveforms. We combined the synthetic waveforms with real-time GNSS noise to produce realistically noisy training waveforms and then tested our model on additional synthetic data and performed inference using the real data that were held back. We discuss the performance of our trained model on both the unseen synthetic data and real inference data. Our model can be used to selectively filter only high-quality data where an earthquake signal is observed for input into an algorithm like GFAST (outperforming a simple signal-to-noise ratio–based filter) to reduce the error in GFAST's real-time earthquake magnitude estimations.

KEY POINTS

- High-noise levels in Global Navigation Satellite Systems (GNSS) data result in slower, less-accurate real-time estimates of earthquake magnitude.
- We developed a deep-learning method that can identify waveforms wherein earthquake signals are present.
- This may allow GNSS-based early warning algorithms to more accurately estimate earthquake magnitudes.

[Supplemental Material](#)

INTRODUCTION AND BACKGROUND

The recent incorporation of Global Navigation Satellite Systems (GNSS) data into the U.S. Geological Survey's ShakeAlert earthquake early warning (EEW) system through the Geodetic First Approximation of Size and Time

(GFAST) algorithm (Crowell *et al.*, 2013, 2016) represents a first-of-its-kind advancement in preparing EEW systems to handle very large earthquakes (Murray *et al.*, 2023). GFAST uses peak ground displacement (PGD) measurements from

1. Department of Earth Sciences, University of Oregon, Eugene, Oregon, U.S.A., <https://orcid.org/0000-0002-9274-6568> (SND); <https://orcid.org/0000-0001-6259-1852> (DM); 2. Now at The Department of Earth and Space Sciences, University of Washington, Seattle, Washington, U.S.A.; 3. Department of Earth and Planetary Sciences, University of California, Davis, California, U.S.A.; 4. U.S. Geological Survey, Geologic Hazards Science Center, Golden, Colorado, U.S.A., <https://orcid.org/0000-0002-0923-3180> (DEG); 5. EarthScope Consortium, Washington, D.C., U.S.A., <https://orcid.org/0000-0001-9984-6724> (DM); 6. School of Earth Sciences, The Ohio State University, Columbus, Ohio, U.S.A., <https://orcid.org/0000-0001-7096-601X> (BWC)

*Corresponding author: sdybing@uw.edu

Cite this article as Dybing, S. N., D. Melgar, A. M. Thomas, D. E. Goldberg, D. Mencin, and B. W. Crowell (2026). Detecting Earthquakes in Noisy Real-Time GNSS Data With Deep Learning for Improved PGD Magnitude Estimation, *Bull. Seismol. Soc. Am.* **XX**, 1–22, doi: [10.1785/0120250220](https://doi.org/10.1785/0120250220)

© Seismological Society of America

GNSS stations to invert for earthquake magnitude (M_{PGD}). Most EEW systems rely only on seismic data to detect and analyze earthquakes, but for great earthquakes, seismic-only systems generally fail to register the true size of these events because of a phenomenon called magnitude saturation (e.g., Kanamori, 2005; Trugman *et al.*, 2019), where large earthquakes are indistinguishable from even larger ones ($M \gtrsim 7.5$). Many studies have demonstrated that GNSS techniques do not saturate in the near field of large earthquakes because, for a fixed geometry of stations and faults, near-field deformation scales linearly with moment release. For a comprehensive overview of existing GNSS studies for early warning, refer to Crowell (2024).

Magnitudes calculated using seismic *P*-waves saturate at the high end of the magnitude scale for several reasons. Traditionally, to calculate magnitudes, seismic data are twice integrated from acceleration to displacement, but this requires the implementation of high-pass filters to remove baseline drift effects that appear during integration. This removes low-frequency content from the data, changing the relationship between observed displacement and the calculated magnitude (Colombelli *et al.*, 2012). Although the relationship between the log of PGD in seismic data and magnitude is linear for moderate-magnitude earthquakes, for larger magnitudes (e.g., $M \gtrsim 7$ when peak displacement is measured over a 4 s time window, or $M \gtrsim 7.5$ when measured over a 20 s time window), this relationship does not hold (Trugman *et al.*, 2019).

In addition, the source spectra of earthquakes of different magnitudes can affect the accuracy of a magnitude computed from those seismic waves because, as earthquake magnitude increases, the source duration and rise time of the earthquake also increase (Melgar and Hayes, 2017). This results in the corner frequency of the source spectrum of the earthquake decreasing as the earthquake magnitude increases (Geller, 1976), and for some seismically derived magnitude types, such as short-period body wave (m_b) or surface wave (m_s), the magnitude calculation is made from data of a specific period (e.g., 1 and 20 s for those two examples, respectively). If the corner frequency for an earthquake is smaller than the measuring frequency, the source spectrum is not flat at that frequency, resulting in a calculated magnitude that is too small and not proportional to the earthquake moment.

This limitation of seismic data for large-magnitude earthquakes has major implications for the dissemination of accurate earthquake and tsunami warnings for systems that only use seismic data to solve for source parameters. A notable example of this occurred during the 2011 Tōhoku earthquake in Japan, where even the fifteenth EEW “forecast” 116.8 s after the system was triggered interpreted the M 9.1 earthquake as an M 8.1 due to magnitude saturation (Hoshiba and Ozaki, 2014). In contrast, GNSS data do not suffer from magnitude saturation because they measure displacement of the ground relative to an absolute reference frame (Melgar *et al.*, 2015),

making it possible to determine the true size of the largest earthquakes. In anticipation of large earthquakes within ShakeAlert’s alerting region, covering the states of Washington, Oregon, and California as of the time of this writing, the ShakeAlert system has been upgraded to include GNSS data analysis with the GFAST algorithm. GNSS position solutions are derived from the time delay and phase accrual of microwave signals emitted by orbiting satellites and collected by receivers on the ground (Melgar *et al.*, 2020). These calculations require knowledge of how much delay time the troposphere and ionosphere introduce, as well as the orbits of the satellites and timing of their clocks, which can be difficult to ascertain with precision in real time. In addition, the physical setting of a receiver station can also affect the position solutions because of multipathing of signals off surrounding structures and terrain. Combined, these factors result in real-time GNSS data that have a very high noise floor on the order of a few to 10 cm (Geng *et al.*, 2018), which obscures small signals such as *P*-wave arrivals. Postprocessed GNSS position solutions, calculated after ~ 2 weeks using “final” clocks and orbits from the International GNSS Service, are less noisy than real-time GNSS data, but real-time solutions are still good enough to use for real-time earthquake characterization in a setting like EEW (Melgar *et al.*, 2019; Goldberg and Haynie, 2022).

Still, the real-time GNSS noise can be highly irregular, containing large spikes with unidentified origins that can be mistaken for large ground motions caused by earthquakes (Kawamoto *et al.*, 2017). This means that to be reliably operational in a real-time EEW environment, GNSS algorithms like GFAST must be triggered by earthquake detections made using seismic instruments to begin analysis for earthquakes in progress. Moreover, as of the time of this writing, for a GFAST magnitude solution (M_{PGD}) to be used in a ShakeAlert warning, three internal conditions within GFAST must be met: (1) the M_{PGD} estimate must be greater than or equal to 7.0; (2) the M_{PGD} uncertainty must be less than or equal to 0.5; and (3) three or more GNSS stations must have recorded PGDs above a time-dependent PGD threshold (Murray *et al.*, 2023). The latter two conditions are empirically derived from real-time GNSS displacement records and are time dependent due to the colored noise structure of GNSS data. Having a separate mechanism to determine if the signal observed on a GNSS time series is the result of seismic shaking or simply noise could directly affect the existing threshold conditions in GFAST and allow for more reliable warnings of intermediate magnitude earthquakes.

Although it could be useful to have a standalone GNSS-based EEW system that could both accurately identify when earthquakes are occurring, as well as estimate their magnitudes without saturation, the former goal is much more difficult to achieve because of the aforementioned noise levels that would obscure the earliest seismic wave arrivals until long after they had been detected by seismometers. However, it has been demonstrated for methods such as the Machine-Learning Assessed Rapid Geodetic Earthquake (M-LARGE) algorithm, which

uses three-component GNSS data to ascertain earthquake source properties such as magnitude, centroid location, and rupture length and width, that reductions in the noise levels of data fed into the algorithm improve the accuracy of the model's predictions of the source properties (Lin *et al.*, 2023). In this study, we developed a model that can quickly detect earthquake signals in noisy real-time GNSS data so that it may act as a filter for high-noise data, ensuring that only waveforms wherein earthquakes are reliably observed and the earthquake signal is strong are fed into algorithms to determine earthquake source properties. A similar dynamic station selection approach was taken using GNSS velocities computed through the time-differenced carrier phase method (e.g., Crowell, 2021) and a random-forest classifier (Dittman *et al.*, 2022; Dittmann *et al.*, 2023); however, we aimed to look directly at the displacements similar to those that are currently available within ShakeAlert.

To accomplish this GNSS-based earthquake identification task, we took a deep-learning (DL) approach. DL is a subset of machine learning that utilizes neural networks to learn patterns from data. Applications of machine learning in seismology and earth sciences have expanded in recent years, with a review of the history, current use, and future directions available from Mousavi and Beroza (2023). These applications include earthquake detection and location (e.g., Perol *et al.*, 2018; Mousavi and Beroza, 2020a; Zhang *et al.*, 2020; Dittman *et al.*, 2022), low-frequency earthquake detection (Thomas *et al.*, 2021), picking of *P*- and *S*-phase arrivals for improved locations and velocity models and earthquake catalog expansion (e.g., Ross *et al.*, 2018; Zhu and Beroza, 2019; Mousavi *et al.*, 2020), and also determination of source parameters such as magnitude (e.g., Lomax *et al.*, 2019; Mousavi and Beroza, 2020b; van den Ende and Ampuero, 2020; Lin *et al.*, 2021; Münchmeyer *et al.*, 2021; Ristea and Radoi, 2021; Licciardi *et al.*, 2022; Lin *et al.*, 2023).

DL methods typically require large datasets with which to train the models. Unlike seismic instrument data, which are maintained in large public datasets (e.g., the EarthScope waveform data archives), there is a relatively small availability of GNSS earthquake data. GNSS instruments and data analysis techniques were introduced into the study of earthquakes much more recently than seismometers, so GNSS records of earthquakes are less plentiful. Furthermore, GNSS data-based earthquake and tsunami early warning algorithms are primarily used during large earthquakes because smaller earthquakes can be handled by seismic algorithms without magnitude saturation. However, there are many more small-magnitude earthquakes than large ones, contributing to a limited GNSS dataset (Gutenberg and Richter, 1949). Therefore, to train a model on a more balanced dataset with increased availability of waveforms with large ground displacements, we generated a dataset of synthetic displacement waveforms using FakeQuakes, a part of a set of codes called MudPy described in Melgar *et al.* (2016)

that have subsequently been used to generate synthetic GNSS data for several DL studies (e.g., Lin *et al.*, 2021, 2023).

In this study, we will present a convolutional neural network (CNN) method for the detection of earthquakes in realistically noisy GNSS data. This CNN architecture was based on U-net (Ronneberger *et al.*, 2015), a model developed for biomedical image segmentation. This structure was also used by Zhu and Beroza (2019) to develop PhaseNet, a model for identifying *P*- and *S*-phase arrivals in seismic data. U-net features both a down-sampling branch to localize and distill the most important features of the training data, as well as an up-sampling branch to increase the resolution of the output and reconnect features from the down-sampling branch to this output. In the case of detection of an earthquake, the model is being trained to determine whether a waveform contains an earthquake, and if so, to identify the time at which the earthquake signal begins (i.e., the *P*-wave arrival).

The variation of the U-net that we used was built for identifying low-frequency earthquakes in seismic data (Thomas *et al.*, 2021), which are more emergent than impulsive. This method also seemed appropriate for the detection of *P*-waves in noisy GNSS data because for some time after arrival, these small-amplitude waves are likely to be obscured by noise. Our model was trained with three-component displacement data derived from synthetic earthquakes and contaminated with real noise (i.e., real-time GNSS recordings during seismic quiescence), as well as signals consisting only of noise. Each waveform sample was labeled with the timing of the *P*-wave arrival for earthquake samples or as noise for the noise-only samples. We then tested the performance of the model on additional, unseen synthetic data and performed inference using a real real-time GNSS dataset provided by EarthScope. Finally, we applied the trained model to a simplified version of the magnitude determination method used in GFAST to determine whether using only GNSS data where an earthquake was detected would improve the accuracy of the real-time magnitude calculation.

DATA AND METHODS

Real data processing

We focused on the 2019 Ridgecrest earthquake sequence (e.g., Ross *et al.*, 2019; Shelly, 2020) for this study because there were many nearby GNSS stations from which to collect real earthquake waveforms, as well as the availability of fault and local velocity models to facilitate simulation of additional earthquakes for training data. We received a dataset provided by EarthScope composed of day-long, three-component, 1 Hz real-time GNSS displacement recordings spanning from 1 June 2019 to 31 July 2020. The data came from all available GNSS stations within 300 km of the Ridgecrest, California, area (Fig. 1). We parsed these data from the provided real-time extended format into day-long mini Standard for the Exchange of Earthquake Data (miniSEED) files.

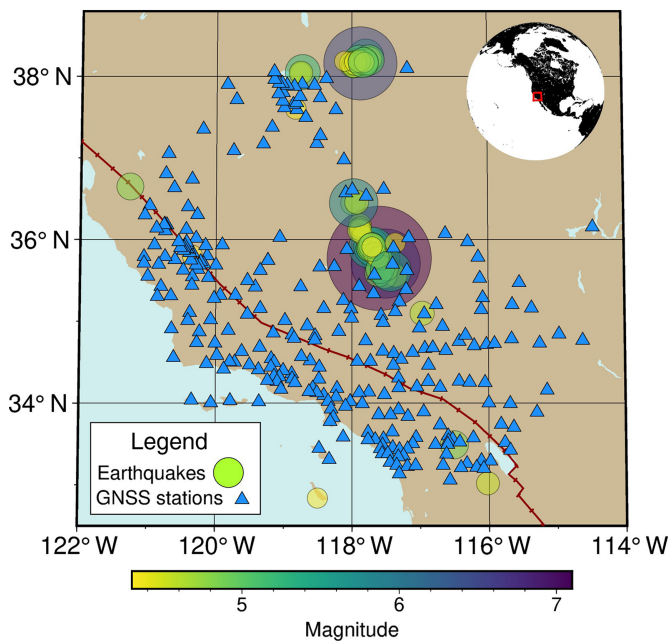


Figure 1. Map of the study region. Global Navigation Satellite Systems (GNSS) stations from which we were given access to archived real-time position solutions in a 300 km radius around Ridgecrest, California, are shown with blue triangles. Earthquakes $M \geq 4.3$ from the Southern California Earthquake Data Center catalog (Southern California Earthquake Data Center [SCEDC], 2013) from which P -wave picks were available are shown by the circles, which vary in size and color by magnitude. The red hatched line indicates the boundary between the Pacific and North American plates. The color version of this figure is available only in the electronic edition.

These real-time data served two purposes for this study. First, they provided a source of observed real-time GNSS noise to add to the noise-free synthetic displacement waveforms we generated for the training of the DL model, which will be described in the next section. Second, they provided real examples of how earthquakes of a range of magnitudes (due to the extensive aftershock sequence) appear in noisy GNSS data across our study region at a range of hypocentral distances. These samples were used as an inference dataset upon which to evaluate the performance of our DL algorithm after it was trained, validated, and tested on synthetic data.

We built a large array of noise-only data samples by selecting random stations and random days from the daily real-time miniSEED dataset, skipping any days in which there were M 4.3 or greater earthquakes in the catalog, and selecting random starting times from the valid daily samples. We trimmed the noise samples to 256 s, demeaned them, and combined the three components of displacement into single samples so that the model was trained on information from all three directions of motion. An example of this format can be found in Figure S1a, available in the supplemental material to this article. This culminated in a noise dataset consisting of over 1.8 million noise samples. These noise samples were then randomly combined with

the synthetic displacement waveforms we generated in a process that trimmed them down to 128 s, which will be described in more detail in the [Data Generation for the DL Model](#) section.

To create the real earthquake-containing samples for the inference dataset (data the model would not see in training that we would use to test its effectiveness), we split the daily miniSEED files into separate 128 s samples. Traces were merged with ObsPy's (Beyreuther *et al.*, 2010) linear interpolation method if there were gaps and demeaned, and the three components of displacement were also combined into single samples. This process resulted in an inference dataset with nearly 1.1 million samples. We then cleaned the dataset by removing any samples where there was a data gap of ≥ 60 s, reducing the dataset to $\sim 994,000$ samples.

To evaluate the performance of the DL model on the inference dataset, we needed to know which of these 1.1 million samples actually contained earthquakes. There is no traditional process for picking P -waves in GNSS data because they are often below the noise floor, so we downloaded a seismic pick catalog for earthquakes $M \geq 4.3$ (circles shown in Fig. 1) during the 1 June 2019 to 31 July 2020 study window using the Southern California Earthquake Data Center catalog (SCEDC, 2013). For a total of 362 events (including two distant earthquakes not shown in Fig. 1), we collected P -wave picks from all available seismic stations in the study region from network codes CI (California Institute of Technology and United States Geological Survey Pasadena, 1926) and PB (Plate Boundary Observatory Borehole Seismic Network; operated by EarthScope) for all the event origin times in the earthquake catalog. It is somewhat rare for seismic stations and GNSS stations to be collocated, so we approximated P -wave arrival times at the locations of our GNSS stations using the seismic pick catalog. To do this, we first calculated the travel times from each event origin time to each of the seismic stations for which we had P -wave picks using ObsPy's (Beyreuther *et al.*, 2010) TauP function and the 1D Mojave area velocity model from the Southern California Earthquake Center's (SCEC) Broadband Platform v. 15.3.0 (Maechling *et al.*, 2015). We then interpolated those seismic travel times to the locations of the GNSS stations using a piecewise linear interpolator and converted these back to P -wave arrival times using the event origin times.

Because our $\sim 994,000$ data samples still had the original timing information associated with them, we were able to use these interpolated P -wave arrival times to determine which samples contained a P -wave arrival and where it was in the waveform, and which did not. This left us with 2123 samples from the full inference dataset that contained earthquakes, or 0.214% of the dataset. A histogram of the magnitude distribution of the waveform samples is shown in Figure S2a.

Producing the synthetic displacement data

DL methods require a substantial amount of training data to generalize in testing. In our real dataset, less than 1% of the

waveforms contained the *P*-wave arrival from an earthquake. To produce a more balanced training dataset, we generated a set of realistic earthquake ruptures and displacement waveforms using FakeQuakes (Melgar *et al.*, 2016). These waveforms were used in the training, validation, and initial testing process of our DL algorithm, and after testing on this simulated data, we also evaluated the algorithm's proficiency with the inference dataset containing real earthquakes described previously.

Synthetic earthquakes are generated with FakeQuakes by defining slip distributions and kinematic rupture parameters for a rupture, then synthesizing waveforms from that rupture. This process is explained in greater detail in Melgar *et al.* (2016) and specifically for DL use in Lin *et al.* (2021) but will be briefly summarized here. Slip distributions are generated by calculating the effective length and width of a fault with the target magnitude defined by the user, which are based on previously published scaling laws using a stochastic approach that allows for some variability in rupture size (Blaser *et al.*, 2010). The mean slip across the entire fault is prescribed to meet the target magnitude, and the actual slip on each individual subfault varies and is controlled by a Von Karman correlation function, which describes real observed earthquake slip patterns well (Graves and Pitarka, 2010, 2015). This allows for the construction of a covariance matrix scaled to the total number of subfaults, wherein the vector of slip at each subfault is expressed by the Karhunen–Loève expansion and is a function of the mean slip and the eigenvalues and eigenvectors of the covariance matrix. This allows for the generation of many slip distributions, which are sampled from a probability density function with the covariance and, therefore, the spatial statistics of slip required to produce slip distributions with the characteristics that the modeler desires (LeVeque *et al.*, 2016).

Next, kinematic rupture parameters are defined, including the velocity of rupture, which depends on the local shear wave-speed and the depth of each subfault. The rise time at each subfault, or the amount of time that slip is occurring on that particular subfault, is scaled by the square root of the total slip on that subfault. This rise time is then used to compute the slip-rate function for each subfault, based on the Dreger slip-rate function (Mena *et al.*, 2010). To generate waveforms, a matrix of Green's functions is constructed by computing impulse responses for every subfault–station pair following the frequency–wavenumber algorithm from Zhu and Rivera (2002). The subfault slip-rate functions are then convolved with the Green's function matrix and multiplied by the slip vector, producing three-component displacement waveforms.

For this study, we elected to generate strike-slip FakeQuakes ruptures along the largest of the four fault planes involved in the 2019 *M* 7.1 Ridgecrest mainshock from the inversion of Goldberg *et al.* (2020), which was composed of 426 triangular subfaults. This fault plane was set in SCEC's 1D Mojave area velocity model (Maechling *et al.*, 2015). We placed receiver

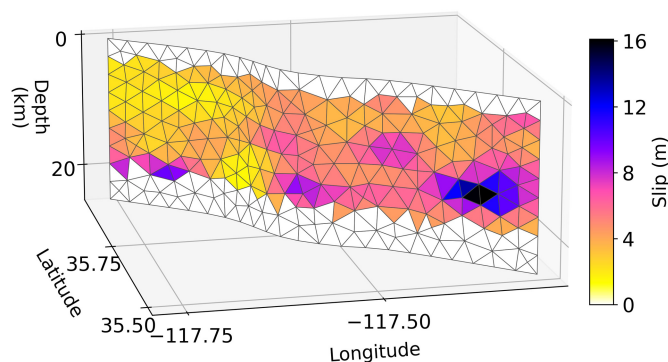


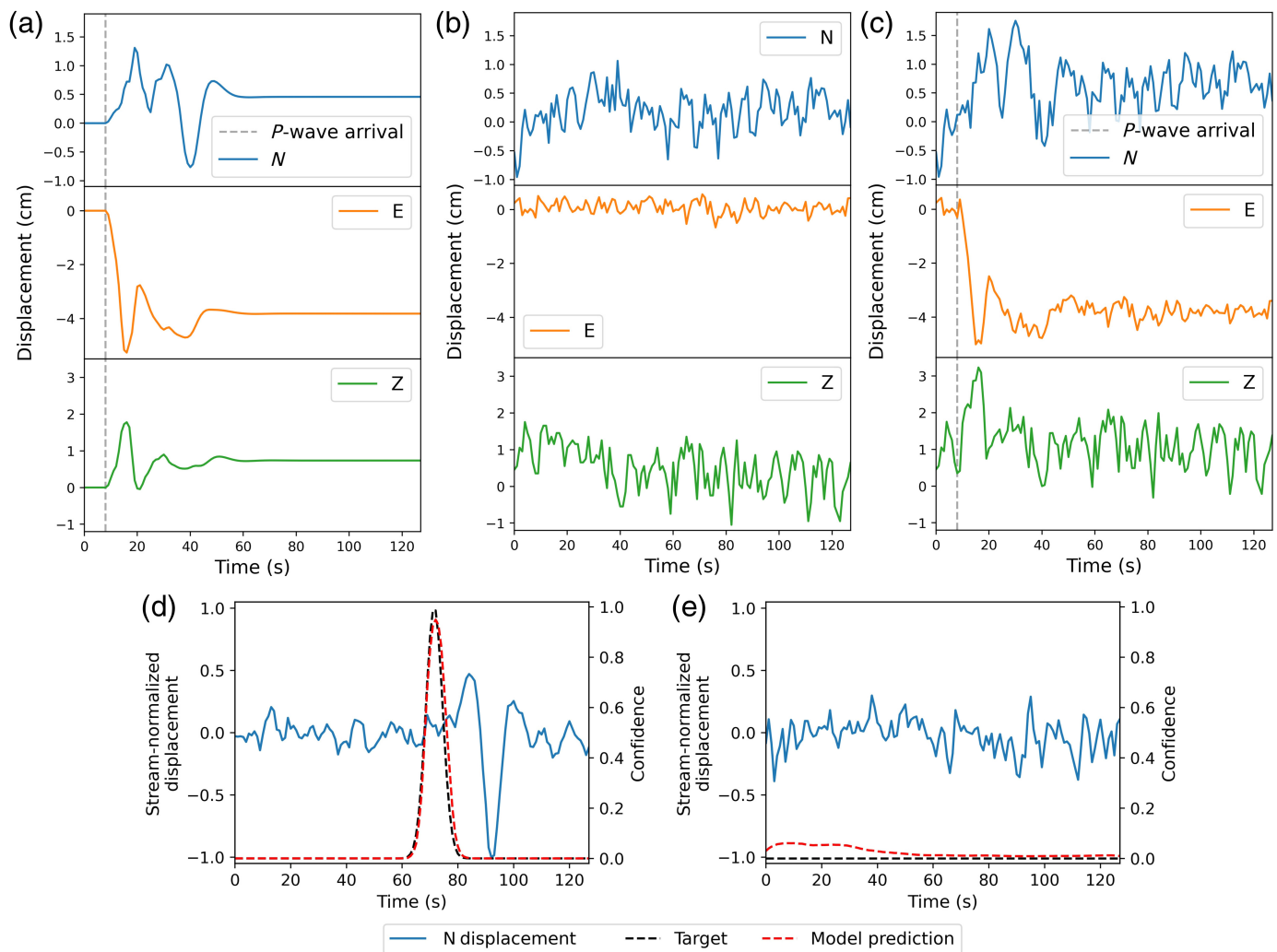
Figure 2. Example of the rupture pattern for one of the FakeQuakes ruptures used to generate synthetic displacement waveforms for the training dataset. The triangular shapes represent individual subfaults. This earthquake had a magnitude of *M* 7.45. The color version of this figure is available only in the electronic edition.

stations for the simulations at the locations of the real GNSS stations within 300 km of Ridgecrest because we also received real data for those stations.

We generated ruptures with requested magnitudes between 4.3 and 7.5 in intervals of 0.1 magnitude units with 100 ruptures per magnitude bin, for a total of 3300 ruptures. The earthquake hypocenters were assigned randomly with the generation of each rupture, and we allowed the earthquake magnitudes to vary slightly with each rupture as well. A visualization of one example rupture plotted by slip amplitude on the fault geometry can be found in Figure 2. A histogram of the final magnitude distribution of all ruptures (*M* 3.6–7.8) can be found in Figure S2b. The displacement waveforms generated were 250 s long at a 1 Hz sampling rate to match the real Ridgecrest sequence GNSS data we received from EarthScope.

After the ruptures and waveforms were generated, we again calculated *P*-wave arrival times using ObsPy's (Beyreuther *et al.*, 2010) TauP function and the 1D Mojave velocity model. The data were then transformed into the format in which they would be introduced to the DL model. The 250-s-long waveforms were zero padded and then trimmed to 256 s so that the *P*-wave arrival was placed in the center of the waveform for each component. The three components were then combined to match the format of the real data as described previously (Fig. S1b). This provided a total of ~916,000 displacement waveforms.

Because FakeQuakes produces ruptures systematically in order across the range of magnitudes requested, the waveforms were shuffled to ensure that the DL model could not learn by the order of the data. Finally, the waveforms were divided into three groups: 80% of the data (~733,000 waveforms) were set aside for training, 10% (~92,000) were used for the validation dataset, and the remaining 10% were saved for the first stage of the testing process.



Data generation for the DL model

The FakeQuakes displacement waveforms we produced were noise free, which is not a sufficient analog to genuine real-time GNSS data that are utilized in EEW. Although MudPy has the capability to generate realistic synthetic noise (Melgar *et al.*, 2020), we chose to use real noise from the real-time GNSS data around Ridgecrest.

We constructed a data generator similar to that of Thomas *et al.* (2021) to feed the data samples into the DL model for training. This data generator first took a random subset of the real noise data to be used as nonearthquake samples in training. It then took an identically sized random subset of the FakeQuakes displacement waveforms and a different random subset of the real noise and combined them. An example of this process and the resulting noisy waveforms are shown in Figure 3a–c, where Figure 3a shows a noise-free displacement waveform produced by FakeQuakes, Figure 3b shows a random time series of pure noise, and Figure 3c shows the resulting noisy waveform from the combination of the two. This resulted in a training dataset where 50% of the waveforms contained synthetic earthquakes, and 50% contained only noise. Recall that the FakeQuakes waveforms were 256 s long and

Figure 3. Example of the process of combining the (a) synthetic displacement waveforms generated with FakeQuakes with (b) real noise from the GNSS stations near Ridgecrest to build (c) realistically noisy synthetic displacement waveforms for the deep-learning (DL) model training process. N, E, and Z indicate the north-south, east-west, and vertical components of displacement, respectively. (d) An example of a sample from the FakeQuakes synthetic testing dataset that contained an earthquake, where the N component of displacement is shown in blue, the target Gaussian function indicating the *P*-wave arrival is shown in black, and the prediction made by the trained DL model is shown in red. (e) An example of a sample from the synthetic testing dataset that contained only noise, where the N component of displacement is shown in blue, the target Gaussian function indicating the *P*-wave arrival is shown in black, and the prediction made by the trained DL model is shown in red. The color version of this figure is available only in the electronic edition.

padded so that the *P*-wave arrival was directly in the center of the waveform. The data generator randomly offset each of the waveforms and trimmed them down to 128 s so that the *P*-wave arrival was no longer in the center of the waveform and instead varied throughout the dataset, with the *P*-wave eligible to be positioned at any sample within the 128 s waveform (functionally, an offset of up to ± 64 s). Finally, to preserve relative amplitudes, all earthquake and nonearthquake (noise)

waveform samples were stream normalized by dividing each component's data by the single maximum absolute amplitude value among all three components.

The data generator also created “target” arrays during this process to facilitate learning for the DL model. The target arrays for nonearthquake samples of pure noise consisted of an array of zeroes. The target arrays for samples with earthquakes consisted of a Gaussian distribution with a user-defined standard deviation (3 s for this study) centered on the same sample as the *P*-wave arrival in the waveform so that the model could learn features of the waveform that indicated where the *P*-wave arrival was. Examples of these targets can be found in Figure 3d,e.

Data generation for the inference dataset was performed similarly but only involved creating the Gaussian distribution target array for the samples that did contain *P*-wave arrivals, and stream normalizing all samples as described previously.

DL model training process

We utilized Thomas *et al.* (2021) variation on the U-net (Ronneberger *et al.*, 2015) CNN architecture for our DL model. Our model was constructed and run using TensorFlow (Abadi *et al.*, 2015). The contracting side of its structure consisted of an input layer followed by three pairs of 1D convolutional layers with 1D MaxPooling (down-sampling) layers. The resulting output was flattened and fed into a dense, fully connected layer that formed the base of the network. The dense layer output was reshaped before entering the up-sampling side of the architecture with three sets of 1D convolutional, 1D up-sampling, and concatenation layers (which connected the convolutional layers from the contracting side of the network to the up-sampling side). A 20% dropout layer was added to help prevent overfitting, and then one final 1D convolutional layer completed the network before the output was flattened again. This architecture can be visualized in Figure S3. The model had a total of 132,161 trainable parameters.

During training, we used the Adam optimizer (Kingma and Ba, 2015) and a binary cross-entropy loss function. An updated trained model was saved if the validation loss had improved at the end of each epoch. Training was initiated with a learning rate of 10^{-4} , and we implemented a learning rate plateau function where the learning rate would decrease when the model proceeded through four training epochs without an improvement in validation loss. We also implemented an early stopping function where the model would quit training after 10 epochs with no validation loss improvement once the learning rate had already reached its lowest allowed value to avoid overfitting. Training stopped automatically because of these functions after 65 epochs, and the training history is shown in Figure S4.

RESULTS

Model performance on the synthetic testing dataset

After completion of training, we first tested the DL model with the remaining 10% of the FakeQuakes waveforms (combined

with noise), which the model had not seen in training or validation. We began with a set of binary classification tests. As with the distribution of the training dataset, the testing dataset consisted of 50% earthquake samples containing *P*-wave arrivals and 50% noise samples, so we wanted to determine how often the model was correct or incorrect about whether a sample contained an earthquake or not (regardless of what time it predicted for the *P*-wave arrival). However, our model did not simply output this binary classification. Instead, the model predicted the probability between 0 and 1 that each timestep in the GNSS time series was a *P*-wave arrival (Fig. 3d,e). To classify whether the model predicted any particular waveform sample to contain a *P*-wave arrival, we first needed to choose a confidence level threshold. If the prediction array's maximum value exceeded that of the confidence threshold, the model predicted an earthquake; if the maximum value of the prediction array was lower than the threshold, the model predicted only noise. This binary classification system resulted in four possible outcomes: (1) true-positive (TP) cases, where the sample contained a *P*-wave arrival and therefore an earthquake, and the model predicted that the sample contained an earthquake/*P*-wave arrival (regardless of the position of the arrival); (2) true-negative (TN) cases, where the sample contained only noise, and the model predicted that the sample contained only noise; (3) false-positive (FP) cases, where the sample contained only noise, but the model predicted that the sample contained an earthquake; and (4) false-negative (FN) cases, where the sample contained an earthquake, but the model predicted that the sample contained only noise.

We calculated classification statistics for all the predictions made on the samples in the FakeQuakes testing dataset using a range of confidence threshold values from 0 to 1. Accuracies, *A*, were computed with

$$A = \frac{TP + TN}{TP + TN + FP + FN}, \quad (1)$$

and the results are shown in Figure 4a. At a threshold of 0, any value of a prediction array higher than 0 resulted in the prediction that a sample did contain an earthquake so the model would predict that every sample contained an earthquake. Because the FakeQuakes testing dataset was 50% earthquakes and 50% noise, the accuracy at this threshold was 50%. Similarly, at a threshold of 1, the model would only predict that a sample contained an earthquake if the confidence value was 1, so it predicted that every sample contained only noise, resulting in a 50% accuracy as well. We achieved the highest overall accuracy of 59.73% at an intermediate threshold of 0.125.

We also computed the precision and recall of the model for the same range of thresholds. Precision, *P*, was computed with

$$P = \frac{TP}{TP + FP}, \quad (2)$$

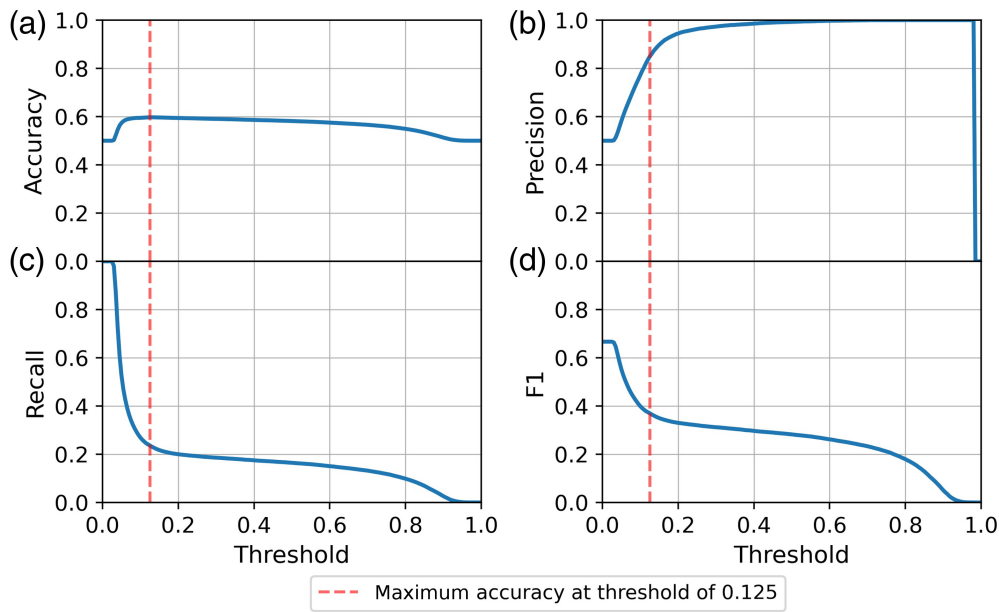


Figure 4. Results of the testing of the DL model on the remaining 10% of the FakeQuakes synthetic displacement waveforms that it never saw in training. (a) Accuracy, (b) precision, (c) recall, and (d) F1 scores were calculated across a range of confidence thresholds from 0 to 1, wherein the threshold determined the cutoff of the maximum confidence of the predicted *P*-wave arrival Gaussian function (Fig. 3d,e). A maximum confidence above the threshold meant a prediction that the sample contained a *P*-wave arrival from an earthquake, and a maximum confidence below the threshold meant a prediction that the sample contained only noise. The threshold at which maximum accuracy was achieved was 0.125, which is indicated on all four panels with the vertical red dashed line. The color version of this figure is available only in the electronic edition.

representing how often the model was correct about the samples that did actually contain earthquakes (Fig. 4b). The highest precisions were found from thresholds of 0.8–0.98. Using a threshold this high in practice would result in limited FPs, but it would also increase the number of FNs (missed earthquakes). Recall, *R*, was computed with

$$R = \frac{TP}{TP + FN}, \quad (3)$$

and represented how successful the model was at finding all the earthquakes in the dataset (Fig. 4c). Recall was highest at very low thresholds from 0 to 0.025. However, using a threshold this low would result in very high numbers of FPs because the model would predict nearly every sample to contain an earthquake. We also computed F1 scores for the threshold range using

$$F1 = \frac{2 * (P * R)}{P + R}, \quad (4)$$

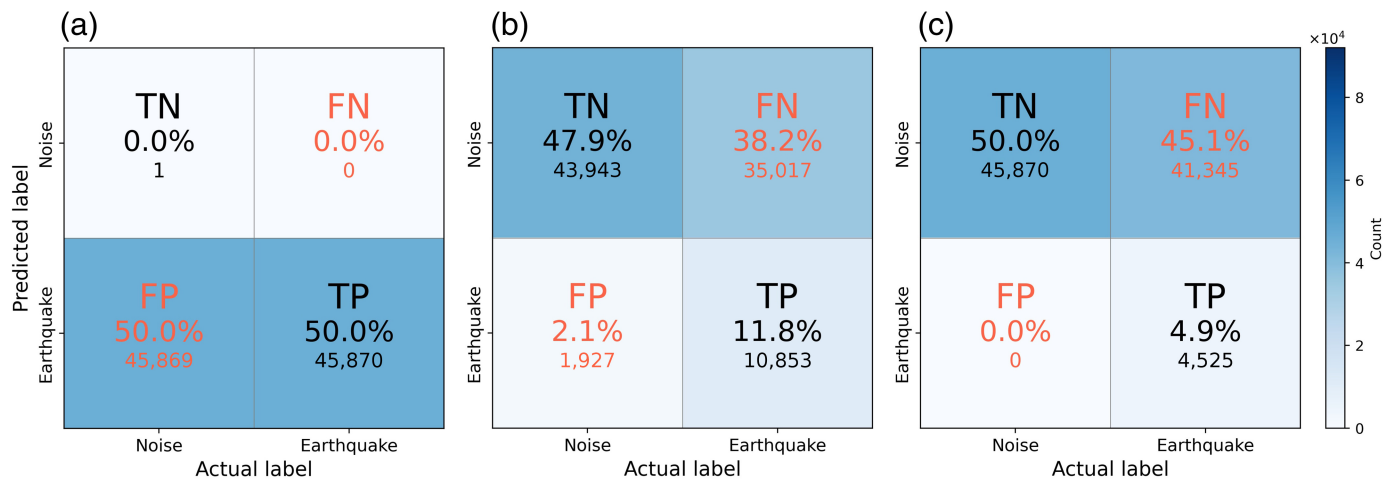
to explore a metric that balances the contrasting precision and recall (Fig. 4d), but the highest F1 score was also reached at a low threshold of 0.025, which would lead to a preponderance of FPs.

Figure 5 shows confusion matrices plotted for the three threshold values that optimized recall and F1 score, accuracy, and precision. This visualization demonstrates how using a low threshold

to optimize recall or F1 score resulted in the model correctly identifying all of the earthquake samples but incorrectly identifying nearly all of the noise samples, leading to a very high FP rate (Fig. 5a). Increasing the threshold to 0.125 at which point accuracy was optimized reduced the number of FPs and led to the model correctly identifying the majority of the noise samples (TN), but the FN rate of missed earthquakes increased as the TP rate decreased (Fig. 5b). Finally, choosing a high threshold that optimized precision eliminated the FPs but further reduced the number of TP earthquakes that the model correctly identified (Fig. 5c). For the following analyses, we focused on the accuracy-optimized threshold of 0.125 for a more balanced view of the model's performance.

For the samples that did contain earthquakes, we wanted to explore how well

the model performed based on the characteristics of those waveforms or the earthquake sources associated with them. We observed that although the model tended to be slightly more accurate for stations closer to the hypocenters of the earthquakes than farther ones, the overall accuracy across the distance range was low, indicating that hypocentral distance was not a primary control on the accuracy of the model's predictions (Fig. 6a). In contrast, we observed a much stronger correlation for earthquake magnitude, with high accuracies exceeding 75% only for earthquakes $M \geq 7$ (Fig. 6b). This aligned with our observation that PGD was similarly positively correlated with accuracy because PGD increases with earthquake magnitude (Crowell *et al.*, 2013; Ruhl *et al.*, 2018; Fig. 6c). Waveforms containing earthquakes with higher PGDs would be expected to be more easily distinguishable from the noise, likely increasing the model's ability to identify them. We checked this by computing the mean signal-to-noise ratio (SNR) from the average of the SNR from each of the three components of data. We calculated SNR using the standard deviations of 20 s of pre-*P* arrival noise and 20 s of post-*P* arrival signal (when possible, based on the position of the *P*-wave arrival in the waveform). We found a positive correlation between mean SNR and accuracy, as expected (Fig. 6d). By the time the SNR exceeded ~ 2 , the model was $\sim 80\%$ accurate, and by an SNR of ~ 3 , it was nearly 100% accurate.



We next explored how well the model was able to identify the correct position of the *P*-wave arrival in the TP cases with earthquakes, again using the intermediate accuracy-optimizing threshold of 0.125. Of the model-predicted arrival times, 94% fell within 30 s of the actual *P*-wave arrival times, 81% fell within 5 s, and 48% fell within 1 s (Fig. 7). These differences are larger than those typically observed for seismic arrival time data from machine learning studies (e.g., Zhu and Beroza, 2019), but this was not unexpected given the noise levels of the GNSS data because in many cases the arrivals were not obvious.

Model performance on the inference dataset

After testing the trained model on the remaining FakeQuakes data, we then performed inference with the dataset containing real earthquakes from the stations and events around Ridgecrest, California (Fig. 1), beginning by calculating accuracy, precision, recall, and F1 score across the 0–1 confidence threshold range (Fig. 8).

The results for the inference dataset looked very different from the results for the FakeQuakes testing data. Whereas the FakeQuakes data were split evenly to be half earthquakes and half noise, in the inference dataset, only 2123 of the ~994,000 samples actually contained earthquake *P*-wave arrivals (Fig. S2a). Therefore, the model predicted that almost every sample did not contain an earthquake at all thresholds above very low ones, resulting in a very high accuracy because so few of the samples did actually contain earthquakes (Fig. 8a). The highest accuracy for the inference dataset was 99.79% at several thresholds, the lowest of these being 0.89. Precision was also highest at thresholds of 0.89–0.895 (Fig. 8b), and recall was highest at very low thresholds of 0–0.025 (Fig. 8c). F1 scores for this dataset remained very low across the entire threshold range, but they were highest at a threshold of 0.67 (Fig. 8d).

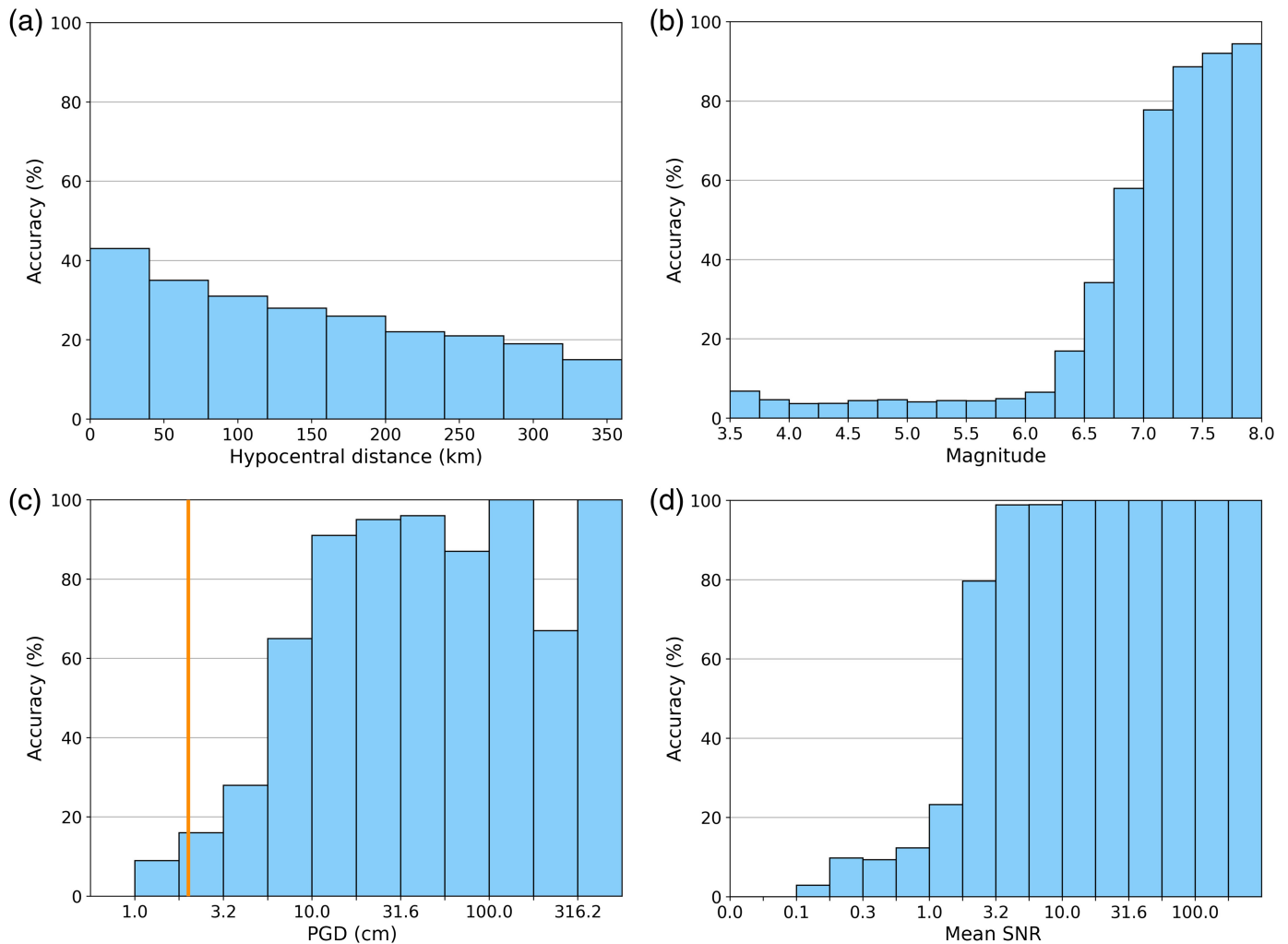
Because of the low number of true earthquake samples in the dataset from which TP or TN results could be produced, we also plotted the number of each of the TP, TN, FP, and FN outcome cases across the full threshold range (Fig. 9). The most

Figure 5. Confusion matrices of the results of the DL model's predictions on the FakeQuakes testing data compared with the true sample labels for (a) a best recall and F1 score confidence threshold of 0.025, (b) the best accuracy threshold of 0.125, and (c) a best precision threshold of 0.8. True-negative (TN) cases indicate when the sample contained only noise and the model predicted only noise; false-negative (FN) cases indicate when the sample contained a *P*-wave arrival from an earthquake but the model predicted only noise; false-positive (FP) cases indicate when the sample contained only noise but the model predicted there was an earthquake; and true-positive (TP) cases indicate when the sample contained an earthquake and the model predicted that there was an earthquake. The colorbar indicates how many samples of each case were in a particular box in the matrix. The percentages indicate the percentage of the total number of predictions that each of the four categories contained, and the numbers under the percentages indicate the number of predictions in each category. The FP and FN boxes that represent incorrect predictions by the model have red-colored text. The color version of this figure is available only in the electronic edition.

dramatic change in outcomes occurred at very low thresholds around 0.04. Lower than this threshold, the TP and FP cases were maximized because at low thresholds, the model predicted that every sample contained an earthquake. The number of TPs and FPs quickly decreased asymptotically to zero higher than this threshold. In contrast, below this threshold, there were no FN or TN cases, but above the threshold, these cases quickly rose in opposition to the decrease in TPs and FPs. This made it difficult to choose an appropriate threshold for analysis because ideally the threshold would maximize TP and TN cases and minimize FP and FN cases, and there was no threshold at which both goals could be simultaneously achieved.

For this reason, we performed the next analyses at several different thresholds: (1) 0.025, where recall was maximized and all 2123 earthquakes were found; (2) 0.04, where the crossover occurred in Figure 9; (3) 0.125, which was the highest accuracy threshold found with the FakeQuakes testing dataset; (4) 0.67, where the F1 score was highest for the inference dataset; and (5) 0.89, which maximized the accuracy and precision for the inference dataset.

We plotted confusion matrices for each of these thresholds in Figure 10. Using the recall-optimized threshold of 0.025, there



were 2123 TP cases, capturing all the earthquakes in the inference dataset, but most of the results for this threshold were FPs (Fig. 10a). With the threshold of 0.04 from the crossover in Figure 9, approximately 57% of the real earthquakes were captured as TP cases, with the remainder moving to the FN category (Fig. 10b). The number of FP cases also decreased, instead becoming TNs. Using the threshold of 0.125 from the best accuracy results of the FakeQuakes testing data reduced the number of actual earthquakes captured (TPs) to only 4.1% of the total number of earthquakes, but the number of FP cases improved substantially (Fig. 10c). Further raising the threshold to the F1 score-optimized 0.67 reduced the number of TPs again to 1.4% and again improved the number of FP cases (Fig. 10d). Finally, using the inference dataset accuracy and precision-optimized threshold of 0.89, the number of TPs and FPs were both very low because the model predicted that almost no samples contained *P*-wave arrivals from earthquakes, missing all but five waveforms with *P*-wave arrivals (Fig. 10e). However, because the number of TN cases was so high and most of the samples in the inference dataset were just noise, the overall accuracy was very high.

We next took a closer look at the model's performance on the 2123 samples that did contain earthquake *P*-wave arrivals by

Figure 6. Plots showing the accuracy of the DL model's predictions on the FakeQuakes testing data compared with binned source and waveform characteristics such as (a) hypocentral distance in kilometers, (b) earthquake magnitude, (c) waveform peak ground displacement (PGD) in centimeters, and (d) the mean signal-to-noise ratio (SNR; computed as the mean of the SNR for the north, east, and vertical components). All plots show results for the confidence threshold of 0.125. Note in (c) and (d) that the x axis scales logarithmically, not linearly. In (c), there is also a vertical orange line indicating the position of 2 cm as a reference to a reasonable approximation of the noise levels of GNSS data (e.g., ~1 to 5 cm for periods between ~10 and 100 s; Melgar *et al.*, 2020). The color version of this figure is available only in the electronic edition.

examining how they were classified compared with their associated magnitude, PGD, and mean SNR (Fig. 11). For this analysis, we focused on the three more conservative thresholds, skipping 0.025 and 0.04, because using those thresholds resulted in most waveform samples being identified as earthquakes and did not tell us much about how the model performed relative to how clear the earthquake signal was. The average magnitude associated with the waveforms in the inference dataset was 4.82, the average PGD was 2.93 cm, and the average mean SNR was 1.18.

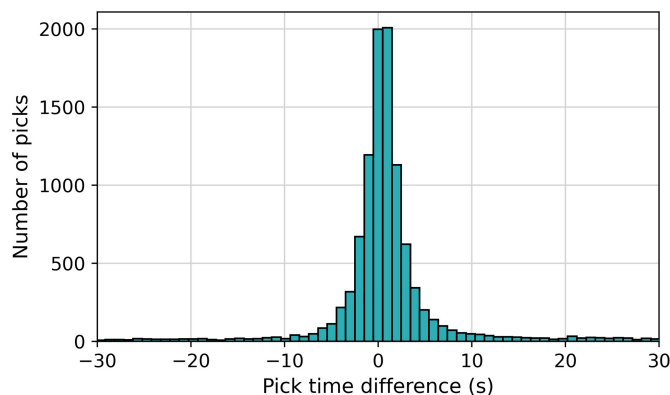


Figure 7. Histogram of the distribution of differences between the actual P -wave arrival times and the DL model's predicted arrival times for the FakeQuakes testing dataset using the confidence threshold of 0.125. The color version of this figure is available only in the electronic edition.

When using a lower confidence threshold of 0.125, the best accuracy threshold from the FakeQuakes testing dataset, the model correctly identified the inference dataset samples consisting of waveforms from the largest earthquake in the dataset (the M 7.1 Ridgecrest mainshock) and many of the samples from the M 6.4 Ridgecrest foreshock, but the results were more scattered for magnitudes smaller than this (Fig. 11a). There was not a particularly strong correlation between PGD and

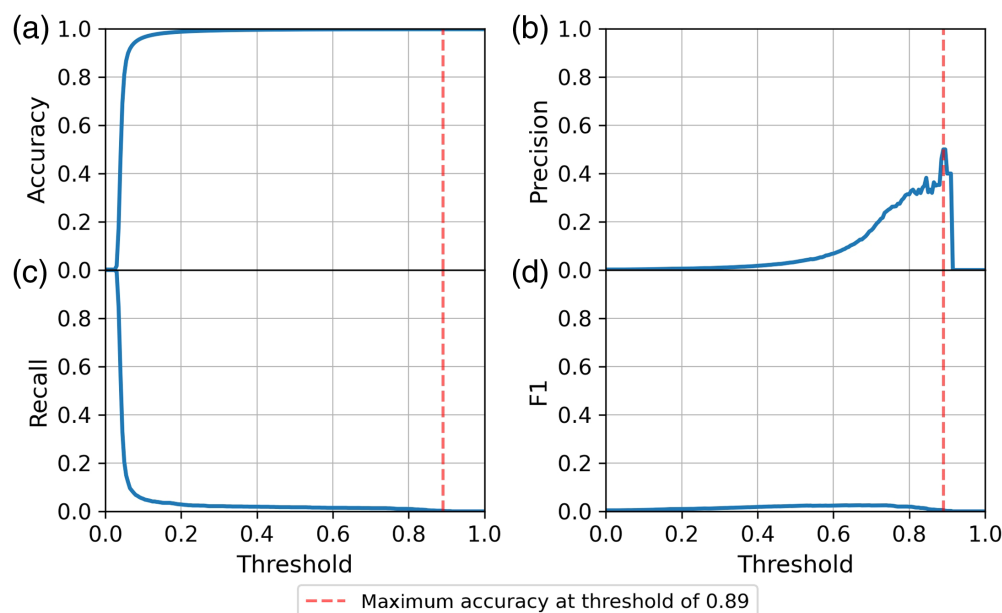
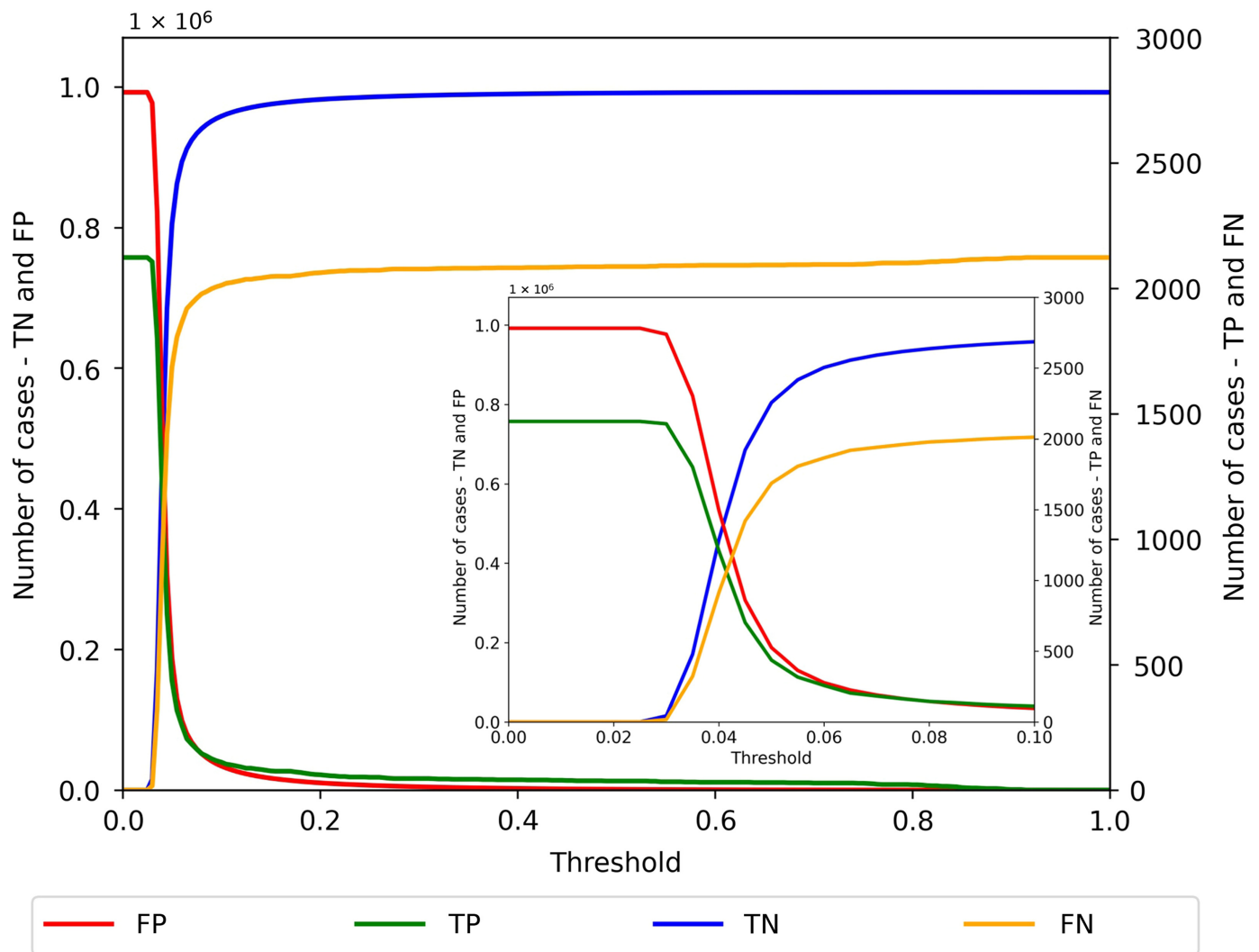


Figure 8. Results of the testing of the DL model on the inference dataset. (a) Accuracy, (b) precision, (c) recall, and (d) F1 scores were calculated across a range of confidence thresholds from 0 to 1, wherein the threshold determined the cutoff of the maximum confidence of the predicted P -wave arrival Gaussian function (Fig. 3d,e). A maximum confidence above the threshold meant a prediction that the sample contained a P -wave arrival from an earthquake, and a maximum confidence below the threshold meant a prediction that the sample contained only noise. The minimum threshold at which maximum accuracy was achieved was 0.89, which is indicated on all four panels with the vertical red dashed line. The color version of this figure is available only in the electronic edition.

whether the model identified the samples correctly; there were several very high-PGD waveforms that the model missed (Fig. 11b). The majority of the high-SNR samples were correctly identified by the model (Fig. 11c). For this low confidence threshold, the average magnitude associated with the earthquake waveforms the model correctly identified was 5.78, and the average magnitude associated with the earthquake waveforms the model failed to identify was 4.78. The average PGD for the correctly identified samples was 5.66 cm, and the average PGD for the missed samples was 2.82 cm. The average mean SNR for the correctly identified samples was 3.84, and the average mean SNR for the missed samples was 1.05. These results demonstrate that the model performed better with waveforms from larger magnitude earthquakes with higher PGDs and SNRs, as expected.

Increasing the threshold to a more conservative 0.67 (at which the F1 score was optimized for the inference dataset) resulted in the model failing to identify earthquakes in samples from any earthquakes other than the M 6.4 and 7.1 Ridgecrest sequence events, and very few of the M 6.4 earthquake samples were identified (Fig. 11d). The average magnitude of the waveforms the model correctly identified was 7.03, and for those missed, the average was 4.79. With this threshold, no samples with PGDs under the mean for the dataset were identified as containing earthquakes (Fig. 11e), with the average PGD for correctly identified samples being 8.4 cm and, for missed samples, 2.85 cm. There was a similar reduction in correct earthquake identifications for samples with mean SNRs under the mean SNR of the dataset but not a complete elimination (Fig. 11f). The average mean SNR for the correctly identified samples was 6.37, and for the missed samples, it was 1.09.

Finally, when we increased the confidence threshold to 0.89, at which point the inference dataset accuracy and precision were optimized, and the model was only able to correctly pick samples from the M 7.1 Ridgecrest earthquake (Fig. 11g). The few samples the model did correctly identify as containing earthquakes had substantially higher PGDs and SNRs than the means of the dataset (Fig. 11h,i). For this threshold, the average magnitude associated with the earthquake waveforms the model



correctly identified was 7.1 (only the Ridgecrest mainshock), and the average magnitude associated with the earthquake waveforms the model failed to identify was 4.82. The average PGD for the correctly identified samples was 6.74 cm, and the average PGD for the missed samples was 2.92 cm. The average mean SNR for the correctly identified samples was 4.84, and the average mean SNR for the missed samples was 1.17.

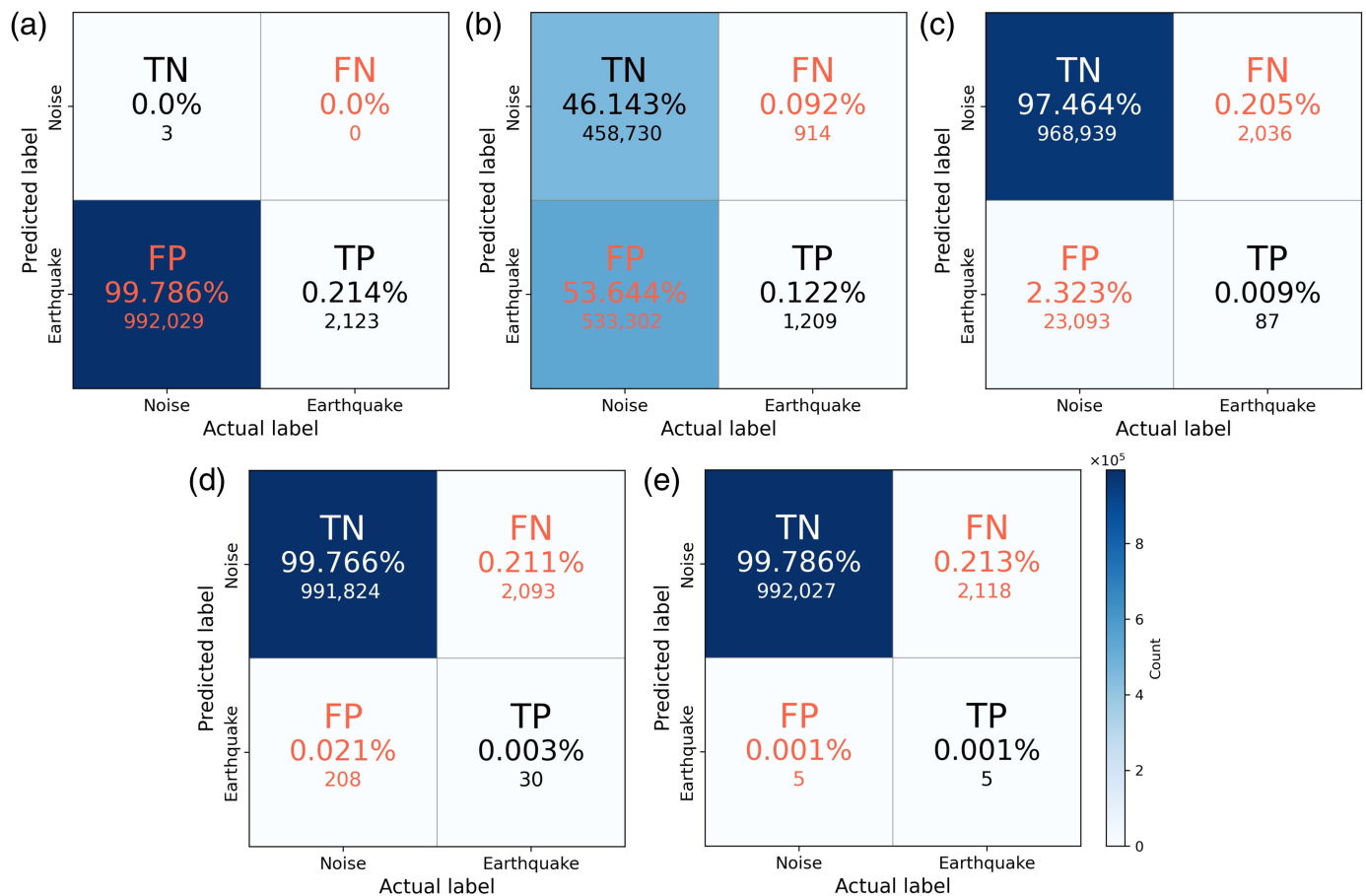
We then examined the model's performance at identifying the correct timing of the *P*-wave arrival in the TP cases with earthquakes for the inference dataset using the five previously discussed thresholds (Fig. 12). We generally observed that the model's performance for the inference data was degraded compared with that of the synthetic FakeQuakes data (Fig. 7), with a much wider distribution of large differences between the true and predicted arrival times. For the inference dataset and the threshold of 0.125, only 72% of the picks fell within 30 s of the true arrival time as opposed to 94% of the FakeQuakes data, 45% fell within 5 s as opposed to 81% with the FakeQuakes data, and 22% fell within 1 s as opposed to 48% with the FakeQuakes data. This difference may be accounted for by the difference in the distribution of magnitudes, PGDs, and SNRs in the FakeQuakes testing dataset compared with the

Figure 9. Plot showing how the number of TP (green), TN (blue), FP (red), and FN (orange) cases in the inference data results changed across the range of confidence thresholds. The inset plot shows a closer cropped look at the region of the larger graph where the lines converge and cross over around a threshold of 0.04. The color version of this figure is available only in the electronic edition.

inference dataset (Fig. S5). For example, in the FakeQuakes dataset, 30% of the waveforms came from earthquakes over *M* 6.5, 10.7% of the waveforms had PGDs over 5 cm, and 5.3% of the waveforms had SNRs over 2, whereas the inference dataset only had 1.5%, 7.1%, and 2% of the waveforms in the same categories, respectively. As shown previously, the model performed better with data from larger earthquakes with higher PGDs and SNRs, and the poorer performance with respect to the timing of the picks in the inference dataset could reflect the relative lack of higher-quality waveforms in it.

DISCUSSION

The model's performance heavily depended on what confidence threshold was chosen to distinguish between samples that were determined to contain earthquakes and those that



did not. The proper choice of threshold would likely vary depending on the use case. Based on the high numbers of FPs observed when lower confidence thresholds were used in this study, we do not find this model to be appropriate for the earthquake detection portion of a theoretical standalone GNSS EEW system. Instead, we suggest that the model could contribute to the existing GFAST framework by acting as a filter to restrict the GNSS data used in the inversion for magnitude to only those waveforms that have high confidence of containing an earthquake signal.

ShakeAlert's GFAST currently computes earthquake magnitudes in real time by calculating PGD for all eligible stations with available data following a trigger from ShakeAlert's seismic algorithms and then using the estimated station hypocentral distance to compute magnitude using the scaling relationship:

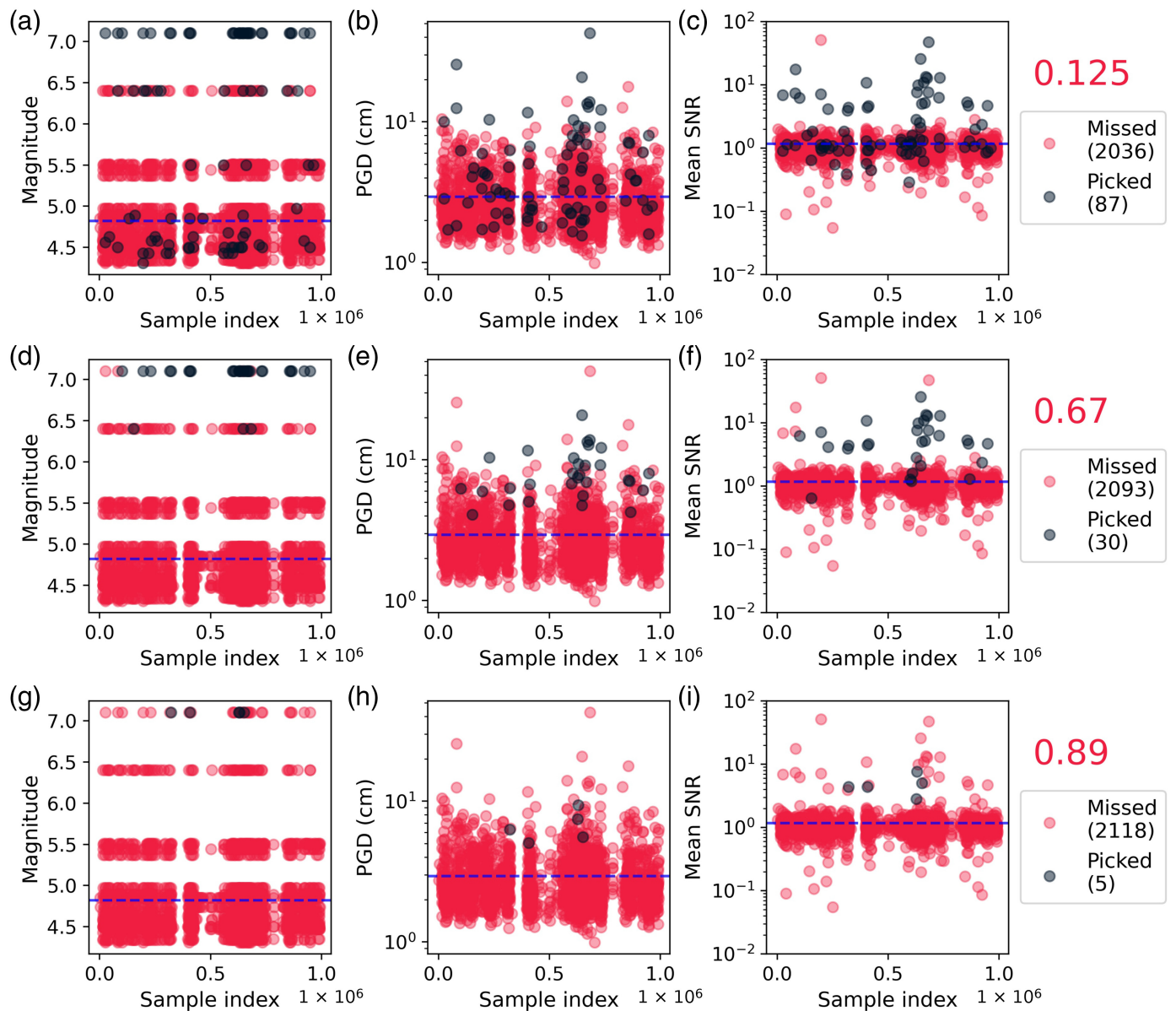
$$\log_{10}(\text{PGD}) = A + BM_w + CM_w \log_{10}(r_{\text{hyp}}), \quad (5)$$

in which PGD is given in centimeters, A , B , and C are coefficients, M_w is the moment magnitude, and r_{hyp} is the hypocentral distance in kilometers (Crowell *et al.*, 2013, 2016; Murray *et al.*, 2023). The coefficients A , B , and C currently used in ShakeAlert are -6.687 , 1.5 , and -0.214 , respectively (Crowell *et al.*, 2016). An exponential epicentral distance weighting is also implemented, where near-source stations are emphasized more strongly in the inversion for event magnitude; note that the

Figure 10. Confusion matrices of the results of the DL model's predictions on the inference data compared with the true sample labels for (a) a threshold of 0.025, where the model captured all of the earthquakes (and recall was maximized), (b) the threshold of 0.04 at which the crossover in Figure 9 occurred, (c) the best accuracy threshold from the FakeQuakes testing results of 0.125, (d) the best F1 score from the inference results of 0.67, and (e) the best accuracy and precision threshold from the inference results of 0.89. TN cases indicate when the sample contained only noise and the model predicted only noise, FN cases indicate when the sample contained a P -wave arrival from an earthquake but the model predicted only noise, FP cases indicate when the sample contained only noise but the model predicted there was an earthquake, and TP cases indicate when the sample contained an earthquake and the model predicted that there was an earthquake. The colorbar indicates how many samples of each case were in a particular box in the matrix. The percentages indicate the percentage of the total number of predictions that each of the four categories contained, and the numbers under the percentages indicate the number of predictions in each category. The FP and FN boxes that represent incorrect predictions by the model have red-colored text. The color version of this figure is available only in the electronic edition.

coefficients from Crowell *et al.* (2016) are regressed using this exponential distance weighting constraint.

GFAST also enforces a travel-time mask that only allows stations to contribute to the magnitude solution once the larger S -waves, wherein the PGD would occur, have already arrived at those stations. This is calculated using a conservative S -wave travel-time estimate of 3 km/s, creating a circle of included stations that grows as the time from the earthquake origin time increases.



GNSS stations that meet these criteria and yet show no clear earthquake signal are still used in GFAST's inversion for earthquake magnitude. Waveforms containing noise alone can return a PGD that corresponds to $M \sim 6-8$ (Murray *et al.*, 2023), leading to substantial overestimations in magnitude for small earthquakes. At the time of this writing, several throttling conditions are in place within ShakeAlert to prevent spuriously high GFAST magnitude estimates caused by noisy data from contributing to alerts. These conditions include minimum magnitudes that both the seismic algorithms and GFAST must meet or exceed as well as checks based on uncertainties in the station positions and maximum PGD, time-dependent minimum PGD, and uncertainty in the GFAST magnitude estimation. For details of these throttling conditions and the current implementation of GFAST in ShakeAlert, refer to Murray *et al.* (2023) and McGuire *et al.* (2025).

As designed and trained, the model presented here could be used to select waveforms that are most likely to contain

Figure 11. Plots showing the outcomes of predictions for the 2123 individual waveforms that included *P*-wave arrivals in the inference dataset. Waveforms where the model successfully identified the earthquakes are colored black, and waveforms where the model did not identify the earthquakes are colored red. For the confidence threshold of 0.125, where accuracy was optimized for the FakeQuakes dataset, results are shown compared with (a) earthquake magnitude, (b) waveform PGD in centimeters, and (c) waveform mean SNR. (d–f) The second row shows the same comparisons for the confidence threshold of 0.67, where the F1 score was optimized for the inference dataset, and (g–i) the third row (g–i) shows the same comparisons for the confidence threshold of 0.89, at which point accuracy and precision were optimized for the inference dataset. The mean values of the parameter shown in each subplot for the dataset are plotted with horizontal blue dashed lines. The color version of this figure is available only in the electronic edition.

earthquake signals for GFAST to analyze, potentially improving GFAST's magnitude estimates by excluding irrelevant noise. Because our model is not currently constructed to operate in

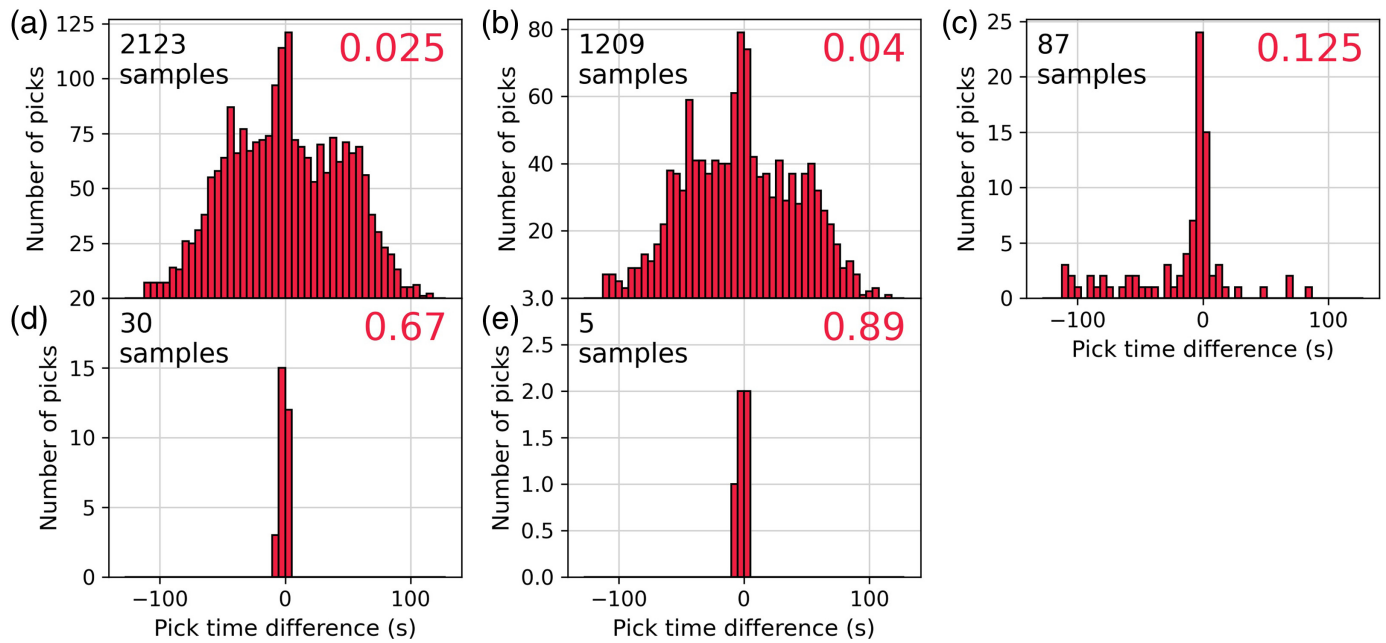


Figure 12. Histograms of the distributions of differences between the actual *P*-wave arrival times and the DL model's predicted arrival times for the inference dataset using confidence thresholds of (a) 0.025, (b) 0.04, (c) 0.125, (d) 0.67, and (e) 0.89 (red text). The color version of this figure is available only in the electronic edition.

real time, only working with 128-s-long waveforms, it would require modification for integration with GFAST; instead, we present a first-order demonstration of its possible utility. For this test, we reformatted the 45,870 earthquake-containing waveforms from the FakeQuakes testing dataset so that instead of the *P*-wave arrival being randomly positioned in the waveform (which included the possibility of it being at the very end), each 128-s-long waveform began at the origin time of the earthquake, increasing the number of waveforms that also contained the *S*-wave arrival. As before, we combined each of these waveforms with a random sample of the real-time GNSS noise. We did not include noise-only waveforms in this test because we were simulating known earthquake events (i.e., we assumed the seismic trigger had occurred).

For a 128-s-long waveform, a 3 km/s travel-time mask like the one used by GFAST would permit stations within 384 km of the hypocenter to contribute to a magnitude estimation. However, the largest hypocentral distance in our FakeQuakes testing dataset was 350 km, so a 3 km/s mask would not actually exclude any waveforms from the magnitude computation. We did, however, implement the distance weighting used by Crowell *et al.* (2016), where the weight for any particular station is

$$w_i = \exp\left(-\frac{r_{\text{epi},i}^2}{8r_{\text{epi},\min}^2}\right), \quad (6)$$

in which $r_{\text{epi},i}$ is the epicentral distance for the i th station, and $r_{\text{epi},\min}$ is the epicentral distance for the station closest to the epicenter. We also included GFAST's requirement that an earthquake must have four stations reporting individual PGD magnitudes to compute the weighted event-averaged magnitude.

As a performance baseline, we computed the event-averaged PGD magnitudes for the entire earthquake-containing

FakeQuakes testing dataset without any kind of GFAST-like modifications or additional filtering. The mean absolute error (MAE) between the true earthquake magnitudes and the calculated event-averaged PGD magnitudes for this unfiltered dataset was 1.312. These data are shown in red for comparison in all panels of Figure 13. Note that for even small true magnitudes, the calculated event-averaged PGD magnitudes are around *M* 7. This is because, as previously mentioned, magnitude estimates made using GNSS noise alone can reach as high as *M* ~6–8 (Murray *et al.*, 2023). Then, in addition to the GFAST-like magnitude computation with epicentral distance weighting described earlier, we also computed event-averaged PGD magnitudes using an SNR filter (where we only allowed waveforms with SNRs > 1, 2, and 3 to contribute; Fig. 13b–d), as well as event-averaged PGD magnitudes using only waveforms that our DL model detected as TPs using both the accuracy-optimized threshold of 0.125 and the precision-optimized threshold of 0.8 from the FakeQuakes testing results (Fig. 13d,e).

In the GFAST-like test (Fig. 13a), because the 3 km/s travel-time mask was irrelevant for this test due to the 350 km maximum hypocentral distance in the dataset, and there were more than four stations available for each event, no events were excluded from the final plot. Although the implementation of the distance weighting did bring some of the smaller magnitude earthquakes down in calculated PGD magnitude closer to the one-to-one line, there were still many earthquakes for which the calculated event-averaged magnitudes were far higher than their true magnitudes. As a result, the MAE for the GFAST-like test was 1.092.

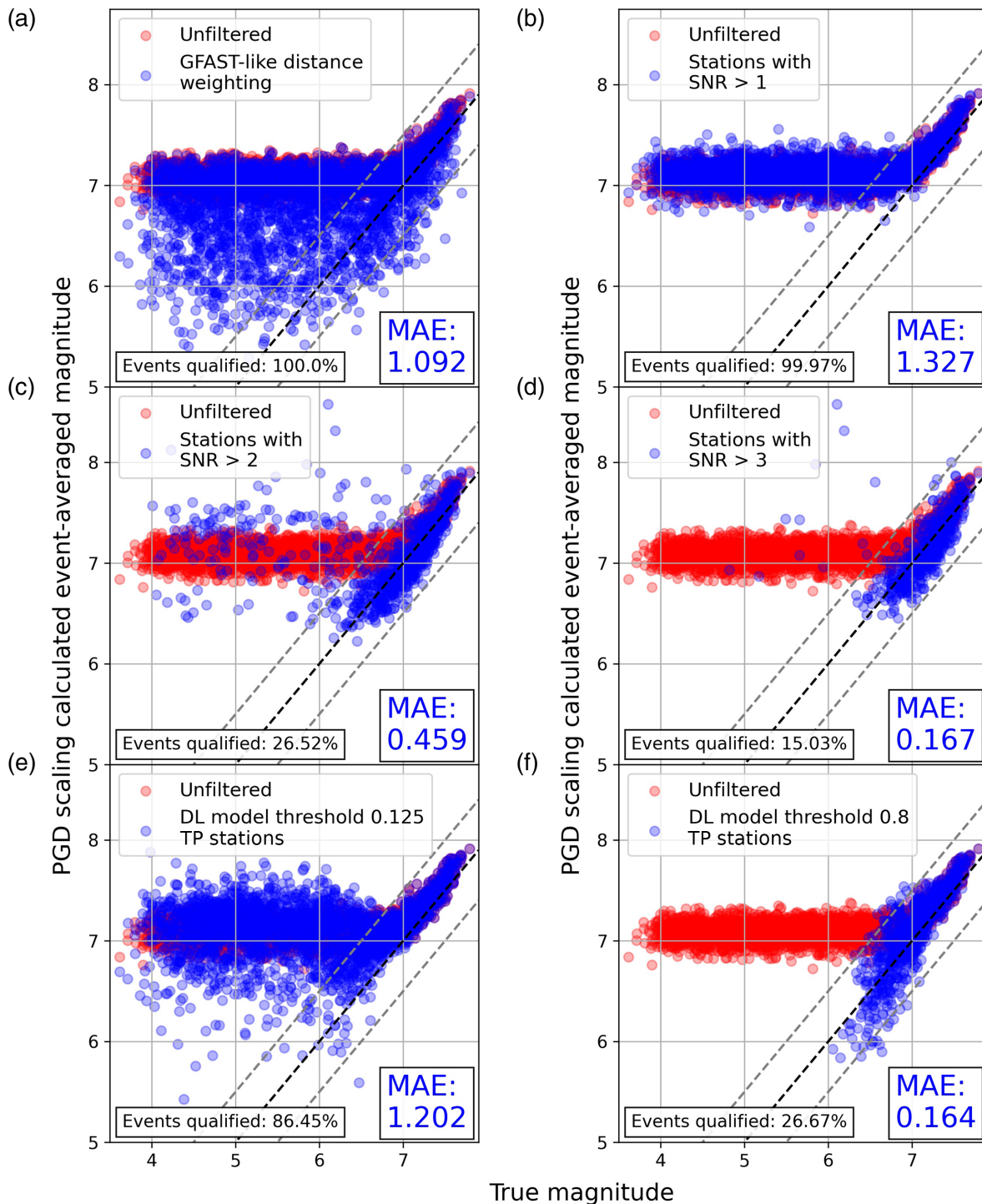


Figure 13. Plots of the event-averaged PGD magnitudes for the FakeQuakes earthquakes in the testing dataset using several different methods to restrict the dataset. The red points indicate event-averaged PGD magnitudes calculated using all the data with no filters, and the blue points indicate event-averaged PGD magnitudes calculated with a restriction on the waveforms permitted to contribute. The black-dashed lines represent the one-to-one relationship between true magnitude and the event-averaged magnitude calculated from the PGD scaling law, with the gray-dashed lines indicating half a magnitude unit above and below the one-to-one line. The mean absolute error (MAE) is shown on the plot for each method. (a) Contributing station magnitudes computed using a Geodetic First

Approximation of Size and Time (GFAST)-like exponential epicentral distance weighting scheme and requiring a minimum of four stations per event. (b) Contributing stations limited to those with waveform SNR > 1. (c) Contributing stations limited to those with waveform SNR > 2. (d) Contributing stations limited to those with waveform SNR > 3. (e) Only stations with TP results as picked by the DL model developed in this study were allowed to contribute, using the accuracy-optimized confidence threshold of 0.125. (f) Only stations with TP results as picked by the DL model developed in this study were allowed to contribute, using the precision-optimized confidence threshold of 0.8. The color version of this figure is available only in the electronic edition.

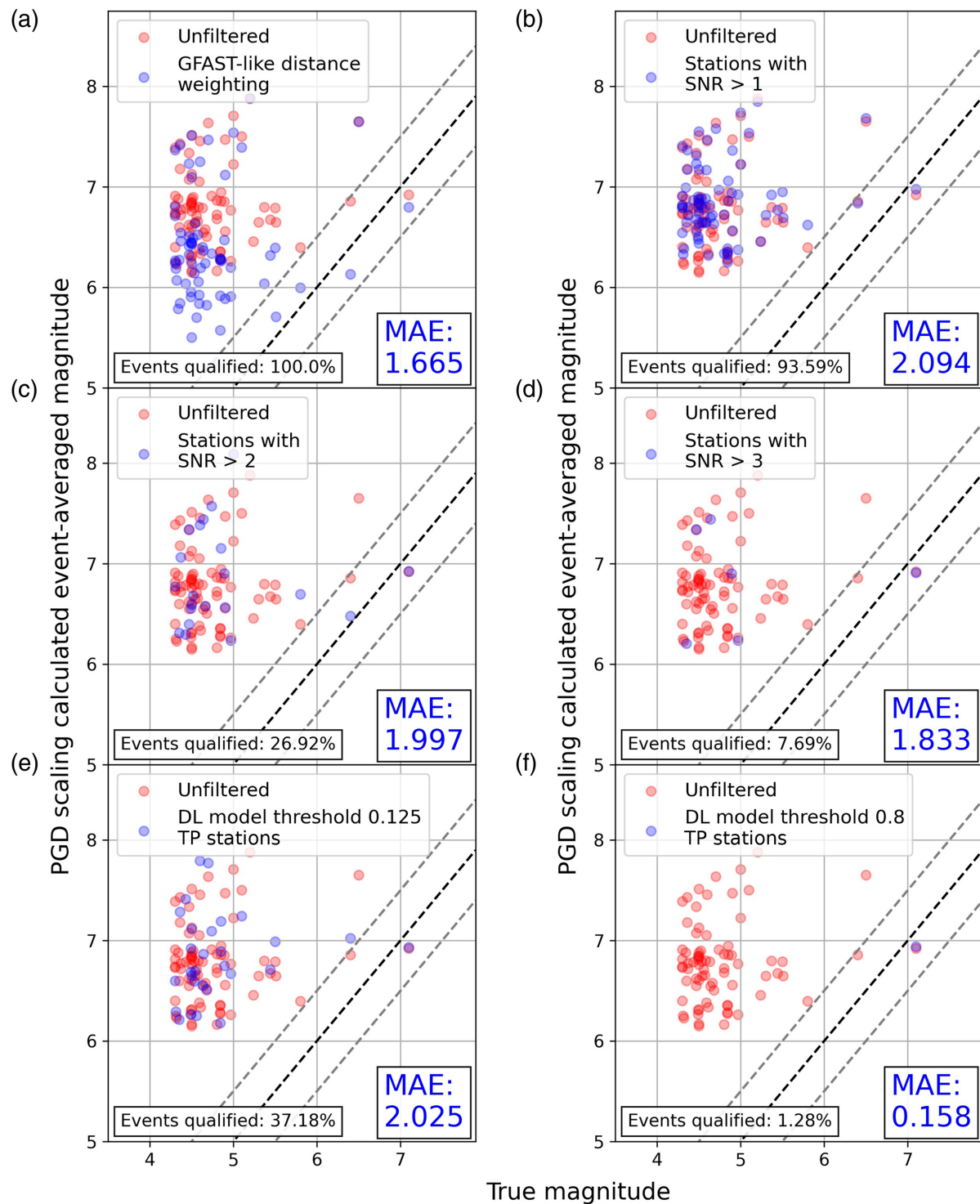


Figure 14. Plots of the event-averaged PGD magnitudes for the earthquakes in the inference dataset using several different methods to restrict the dataset. The red points indicate event-averaged PGD magnitudes calculated using all the data with no filters, and the blue points indicate event-averaged PGD magnitudes calculated with a restriction on the waveforms permitted to contribute. The black-dashed lines represent the one-to-one relationship between true magnitude and the event-averaged magnitude calculated from the PGD scaling law, with the gray dashed lines indicating half a magnitude unit above and below the one-to-one line. The MAE is shown on the plot for each method. (a) Contributing station magnitudes computed using a GFAST-like exponential epicentral distance weighting

scheme and requiring a minimum of four stations per event. (b) Contributing stations limited to those with waveform SNR > 1. (c) Contributing stations limited to those with waveform SNR > 2. (d) Contributing stations limited to those with waveform SNR > 3. (e) Only stations with TP results as picked by the DL model developed in this study were allowed to contribute, using the accuracy-optimized confidence threshold of 0.125 from the FakeQuakes test. (f) Only stations with TP results as picked by the DL model developed in this study were allowed to contribute, using the precision-optimized confidence threshold of 0.8 from the FakeQuakes test. The color version of this figure is available only in the electronic edition.

Including only those waveforms with $\text{SNR} > 1$ in the event-averaged magnitude computation still allowed 99.97% of the earthquakes through and did not improve the performance over the unfiltered dataset; the MAE for this test was even slightly worse than for the unfiltered data at 1.327 (Fig. 13b). When we only permitted use of the earthquake waveforms with $\text{SNR} > 2$, which was approximately when in the FakeQuakes testing dataset the accuracy of the predictions began to show a dramatic improvement (Fig. 6d), only 26.52% of the earthquakes had event-averaged magnitudes computed for them, which removed the majority of the smaller-magnitude events where the difference between the true and computed magnitude was large (Fig. 13c). At magnitudes below $M \sim 7$, performance for the events that remained was still poor, resulting in an MAE of 0.459. Further restricting the permitted waveforms to only those with $\text{SNR} > 3$ reduced the number of events that had computed magnitudes to 15.03% (Fig. 13d). Although some outliers remained, the magnitudes computed for the remaining events above $M \sim 6.5$ fell close to the one-to-one line, resulting in an MAE of 0.167.

Finally, we tested our DL model as a filter, using only those waveforms with TP earthquake detections for the event-averaged PGD magnitude calculations. Using the confidence threshold of 0.125 that maximized the model's performance in terms of accuracy on the FakeQuakes testing dataset permitted magnitude estimations for 86.45% of the earthquakes, but this captured many of the smaller magnitude earthquakes that had high errors compared with the true magnitudes (Fig. 13e). This resulted in an MAE of 1.202, which is better than that of the unfiltered data and the $\text{SNR} > 1$ filter (Fig. 13b) but worse than the GFAST-like distance weighting approach (Fig. 13a).

When we instead used the confidence threshold of 0.8 that maximized precision (Fig. 13f), the percentage of permitted events was reduced to 26.67%, but the MAE improved to 0.164. This was the lowest MAE of any of the filters we used. In addition to MAE, we calculated the mean squared error (MSE) for each of the different filter tests, which is more sensitive to outliers. The MSE for the $\text{SNR} > 3$ filter was 0.097, whereas the MSE for the 0.8 TP confidence threshold was 0.045, further indicating the advantage of the DL model filter. The DL model filter approach with a conservative confidence threshold outperforms the SNR-based filter approach by both metrics and allows for magnitude computation for nearly twice the number of earthquakes compared with the $\text{SNR} > 3$ filter. In addition, most of the magnitude estimates for the DL filter approach for earthquakes over $M 6$ fell within half a magnitude unit of the true magnitude. The scatter in the results also improved with increased earthquake magnitude compared with the $\text{SNR} > 3$ filter test.

We performed these same comparisons with the inference dataset, even though the sample size of earthquakes was very limited (Fig. 14). For the unfiltered inference data, the MAE between the true magnitudes and event-averaged PGD magnitudes was 2.033, which was higher than the MAE for the

unfiltered FakeQuakes data. For simplicity and ease of comparison, we used the same two thresholds of 0.125 and 0.8 for this test on the inference data as we used for the FakeQuakes data. The GFAST-like test on the inference dataset shows more clearly the difference that the epicentral distance weighting makes because the majority of the smaller earthquake magnitude estimates were drawn down toward the one-to-one line using the weighting scheme (Fig. 14a). However, the error was still relatively high with an MAE of 1.665, which was higher than the GFAST test MAE for the FakeQuakes data.

When only waveforms with $\text{SNR} > 1$ were used in the magnitude computation, 93.59% of events were permitted through the filter (Fig. 14b). Similar to the results shown in Figure 13b for the FakeQuakes testing data, this filter resulted in an even higher MAE than that of the unfiltered data at 2.094. Restricting the data further using a filter of $\text{SNR} > 2$ reduced the qualified events to 26.92% but only marginally improved the MAE to 1.997 (Fig. 14c), and restricting the data to $\text{SNR} > 3$ reduced the qualified events further to 7.69% but still only improved the MAE to 1.833 (Fig. 14c). As Figure 14c,d shows, for the inference data using SNR restrictions did not seem to leave only the higher magnitude earthquakes behind as observed in Figure 13c,d.

Next, we allowed only the TP cases detected by the DL model to contribute to the event-averaged PGD magnitude. When we used a threshold of 0.125, similar to the results for the FakeQuakes data (Fig. 13e), many of the smaller magnitude earthquakes that have large differences between true and computed magnitude were still included, with 37.18% of events passing through the filter (Fig. 14e). The MAE was still high for this threshold at 2.025, which was better than the $\text{SNR} > 1$ filter but worse than the $\text{SNR} > 2$ and > 3 filters and the GFAST-like epicentral distance weighting scheme.

Increasing the threshold to 0.8 resulted in only one earthquake, the $M 7.1$ Ridgecrest mainshock, passing through the filter (Fig. 14f). Given the results shown in Figure 11g–i for the high confidence threshold, at which point only that earthquake was successfully picked, this was not necessarily unexpected. Accordingly, the MAE for this event was 0.158, much lower than for the other filtering tests. The PGD magnitude computed for the $M 7.1$ Ridgecrest mainshock using this 0.8 TP threshold was 6.94, but the computed magnitude for the unfiltered dataset was also 6.92—well within the 0.5 magnitude unit lines on the plot. Using the DL model provided a marginal improvement at best for this earthquake.

Ideally, the other two events $M > 6$ in this dataset would have had their magnitude estimates improved using the DL model filter with a threshold of 0.8, as was observed for the FakeQuakes data in Figure 13f, but the DL model failed to identify any stations from those events as TPs at this threshold. A closer investigation of the data explains why the DL model did not improve performance for these events. The $M 6.5$ in the dataset occurred in Nevada (Fig. 1), and there were only six GNSS stations from which we were able to interpolate P -wave

arrivals for that event. The closest of these stations to the hypocenter was over 400 km away, and the SNRs of the earthquake at those stations ranged from 0.8 to 1.7, lower than where we observed the model's best performance in this study. The M 6.4 was the Ridgecrest foreshock, so there were 66 stations from which we had waveforms containing those interpolated P -wave arrival times, and some were quite close to the hypocenter. However, of the 66 waveforms, only two had SNRs > 1.5 (1.71 and 2.28), which were still quite low. Particularly for a higher threshold like 0.8, the model not having high enough confidence that a P -wave arrival existed in the sample with an SNR that low was an expected result.

Our results suggest that using the DL model to select only waveforms in which earthquakes were confidently detected would improve PGD magnitude estimations. This follows the results of the study by Lin *et al.* (2023), which found that the performance of the M-LARGE algorithm that ascertained earthquake source properties such as magnitude, centroid location, and rupture extent was improved by reducing the amount of noise fed into the model. It should be noted, however, that the DL model we presented in this study would require substantial reconfiguration and likely retraining to facilitate real-time operation, and additional tests could help confirm expected performance. The scope of this study was limited regionally and temporally to southern California and a timespan around the 2019 Ridgecrest earthquake sequence, as well as to the real-time GNSS position solutions produced at that time by UNAVCO (now EarthScope). Future work could include expanding to other regions, particularly the rest of the U.S. West Coast, where ShakeAlert and GFAST are active, as well as evaluating the real-time *Fastlane* GNSS position solutions ingested by ShakeAlert (Santillan *et al.*, 2013; Melbourne *et al.*, 2021). In addition, although we included waveforms from smaller earthquakes in the training and testing of this study's DL model, in future work, it may be appropriate to develop a model based only on GNSS data from larger earthquakes (e.g., $M \geq 7$) wherein the PGDs would be larger and earthquake signal more visible in the data because this is the magnitude range where GFAST would be expected to contribute to ShakeAlert. This could potentially improve the performance of a DL model in case the inclusion of waveforms in training where the earthquake signal is below the noise threshold has negative effects on the model's ability to learn to distinguish between earthquakes and noise.

Another potential future avenue for improvement could be the implementation of GNSS denoising into the workflow for computing earthquake magnitudes from GNSS data. Thomas *et al.* (2023) presented a DL model for this application, similarly trained on synthetic data, that showed promise when applied to real data as well. This type of model could potentially be used before a model trained to detect P -wave arrivals in GNSS data, like the one presented in this study, to improve the performance of such a model in terms of the binary classification results, as well as the actual position of the predicted

P -wave arrivals. If successful, this combination may provide a basis for the development of a standalone GNSS EEW system where data are denoised in real time, earthquakes are detected, low-quality data are filtered out, and PGD magnitudes are computed.

CONCLUSIONS

In this study, we presented the construction, training, and testing of a DL model for detecting earthquakes in noisy real-time GNSS data. We trained this model using a suite of synthetic strike-slip earthquakes combined with observed real-time noise and tested it on additional unseen synthetic data. We also performed inference using real earthquake data from southern California over approximately a year around the 2019 Ridgecrest earthquake sequence. When testing the performance of the trained model, we observed that it was generally difficult to pick an appropriate confidence threshold to maximize TPs and TNs while also minimizing FPs and FNs. Maximizing TPs also maximized FPs because to capture the majority of the earthquakes in the dataset, the confidence threshold had to be set so low that many noise-only waveforms were also classified as containing an earthquake signal. Likewise, maximizing TNs also maximized FNs because the confidence threshold had to be set so high that nearly all earthquake samples were missed.

The synthetic testing dataset was composed of equal numbers of noise waveforms and earthquake-containing waveforms. Using a confidence threshold that optimized the accuracy of the model, we found that the model's accuracy was highest ($>75\%$) at detecting earthquakes $M > 7$, and in waveforms with PGDs >10 cm and SNRs above ~ 2 . For the synthetic data with this threshold, the model's predicted P -wave arrival times fell within 5 s of the actual arrival time for 81% of the picks. In contrast, the inference dataset was highly asymmetrical, with only 0.214% of the tested samples containing earthquakes. This made choosing an appropriate threshold even more difficult. Using the same threshold that maximized accuracy for the synthetic testing data, we observed that the model's capability for real earthquake detection depended substantially on the PGD and SNR of the sample as well, and for the inference data, only 45% of the predicted P -wave arrivals were within 5 s of the actual arrival time.

The number of FP results observed in both the synthetic and inference data was high when using a confidence threshold low enough to detect the majority of the actual earthquakes in the data. Current EEW systems that incorporate GNSS data, such as ShakeAlert in the western United States with the GFAST algorithm, require triggering by detections made with seismic data for the GNSS algorithm to begin analysis. Our study results suggest that it may not be realistic to expect a GNSS-only system to detect earthquakes on its own due to the high-noise levels in the data. However, we also explored how our model performed as a filter to prevent waveforms from entering a magnitude estimation algorithm if the

earthquake signal was not confidently detected. We tested the computation of event-averaged PGD magnitudes using all available data with no alterations, as well as an epicentral distance weighting system like the one used in GFAST, wherein stations closest to the epicenter are more strongly weighted in the magnitude computation. We then also tried filtering waveforms by their SNR, allowing only high-SNR waveforms to contribute to the magnitude estimation, and also using this study's DL model to filter out waveforms that the model did not confidently detect as including an earthquake signal. We found that for both the synthetic and inference datasets, the MAE of the event-averaged magnitude estimates compared with the true magnitudes was lowest when the DL model was used compared with using all the unaltered data, the GFAST weighting method, or an SNR filter.

This suggests that using a DL model like the one presented in this study to only allow GNSS waveforms wherein earthquakes are confidently observed above the noise to contribute to the magnitude determination process could improve the accuracy of the magnitude estimates. In future work, this model could be adapted and tested for real-time performance, and data from additional regions, time periods, and GNSS position solution types could be incorporated to explore its applicability to a wider range of scenarios.

DATA AND RESOURCES

The seismic data used for *P*-wave picking in this study were accessed from the National Science Foundation (NSF) National Geophysical Facility data archive operated by EarthScope Consortium (NSF Award 2435260; <https://www.earthscope.org/data/>; last accessed June 2026) under network codes CI (California Institute of Technology and United States Geological Survey Pasadena, 1926) and PB (Plate Boundary Observatory Borehole Seismic Network; operated by EarthScope). The authors appreciate the hard work of the engineers and network operators who make these data available in near-real time. The real-time Global Navigation Satellite Systems position solutions we used were provided by Kathleen Hodgkinson, formerly with UNAVCO (now EarthScope), who at the time of writing is now with Sandia National Laboratories. These data, formatted as used in this study, are available at doi: [10.5281/zenodo.20617754](https://doi.org/10.5281/zenodo.20617754) (Dybing *et al.*, 2026). Codes used to process and analyze data and make the figures for this study are available at <https://github.com/UO-Geophysics/gnss-picker> (last updated November 2024 as of the time of this writing). The supplemental material for this article contains five figures that illustrate characteristics of the waveform datasets, the deep-learning model architecture, and the model training history.

DECLARATION OF COMPETING INTERESTS

The authors acknowledge that there are no conflicts of interest recorded.

ACKNOWLEDGMENTS

The authors would like to acknowledge use of the Broadband Platform software provided by the Southern (now Statewide) California Earthquake Center (<http://scec.org>), which was funded by National

Science Foundation Cooperative Agreement EAR-1600087 and U.S. Geological Survey Cooperative Agreement G17AC00047. The authors thank two anonymous reviewers, Tim Clements, Will Yeck, Brian Shiro, and an additional U.S. Geological Survey reviewer for their feedback, which helped improve this article.

Any use of trade, firm, or product names is for descriptive purposes only and does not imply endorsement by the U.S. Government.

REFERENCES

- Abadi, M., A. Ashish, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, *et al.* (2015). TensorFlow: Large-scale machine learning on heterogeneous systems [Software], *Zenodo*, doi: [10.5281/zenodo.4724125](https://doi.org/10.5281/zenodo.4724125).
- Beyreuther, M., R. Barsch, L. Krischer, T. Megies, Y. Behr, and J. Wassermann (2010). ObsPy: A Python toolbox for seismology, *Seismol. Res. Lett.* **81**, no. 3, 530–533, doi: [10.1785/gssrl.81.3.530](https://doi.org/10.1785/gssrl.81.3.530).
- Blaser, L., F. Krüger, M. Ohrnberger, and F. Scherbaum (2010). Scaling relations of earthquake source parameter estimates with special focus on subduction environment, *Bull. Seismol. Soc. Am.* **100**, no. 6, 2914–2926, doi: [10.1785/0120100111](https://doi.org/10.1785/0120100111).
- California Institute of Technology and U.S. Geological Survey Pasadena (1926). Southern California Seismic Network [Data set], *International Federation of Digital Seismograph Networks*, doi: [10.7914/SN/CI](https://doi.org/10.7914/SN/CI).
- Colombelli, S., A. Zollo, G. Festa, and H. Kanamori (2012). Early magnitude and potential damage zone estimates for the great M_w 9 Tohoku-Oki earthquake, *Geophys. Res. Lett.* **39**, no. 22, L22306, doi: [10.1029/2012GL053923](https://doi.org/10.1029/2012GL053923).
- Crowell, B. W. (2021). Near-field strong ground motions from GPS-derived velocities for 2020 Intermountain Western United States earthquakes, *Seismol. Res. Lett.* **92**, no. 2A, 840–848, doi: [10.1785/0220200325](https://doi.org/10.1785/0220200325).
- Crowell, B. W. (2024). Earthquake and tsunami early warning with GNSS data, in *GNSS Monitoring of the Terrestrial Environment*, Y. Aoki and C. Kreemer (Editors) Elsevier, Amsterdam, The Netherlands, 111–127, doi: [10.1016/B978-0-323-95507-2.00004-9](https://doi.org/10.1016/B978-0-323-95507-2.00004-9).
- Crowell, B. W., D. Melgar, Y. Bock, J. S. Haase, and J. Geng (2013). Earthquake magnitude scaling using seismogeodetic data, *Geophys. Res. Lett.* **40**, no. 23, 6089–6094, doi: [10.1002/2013gl058391](https://doi.org/10.1002/2013gl058391).
- Crowell, B. W., D. A. Schmidt, P. Bodin, J. E. Vidale, J. Gomberg, J. R. Hartog, V. C. Kress, T. I. Melbourne, M. Santillan, S. E. Minson, *et al.* (2016). Demonstration of the Cascadia G-FAST geodetic earthquake early warning system for the Nisqually, Washington, earthquake, *Seismol. Res. Lett.* **87**, no. 4, 930–943, doi: [10.1785/0220150255](https://doi.org/10.1785/0220150255).
- Dittmann, T., Y. Liu, Y. Morton, and D. Mencin (2022). Supervised machine learning of high rate GNSS velocities for earthquake strong motion signals, *J. Geophys. Res. Solid Earth* **127**, no. 11, e2022JB024854, doi: [10.1029/2022JB024854](https://doi.org/10.1029/2022JB024854).
- Dittmann, T., Y. J. Morton, B. Crowell, D. Melgar, J. DeGrande, and D. Mencin (2023). Characterizing high rate GNSS velocity noise for synthesizing a GNSS strong motion learning catalog, *Seismica* **2**, no. 2, 978, doi: [10.26443/seismica.v2i2.978](https://doi.org/10.26443/seismica.v2i2.978).
- Dybing, S. N., D. Melgar, A. M. Thomas, D. E. Goldberg, D. Mencin, and B. W. Crowell (2026). Ridgecrest sequence FakeQuakes synthetic displacement waveforms and real GNSS data [Data set], *Zenodo*, doi: [10.5281/zenodo.20617754](https://doi.org/10.5281/zenodo.20617754).

- Geller, R. J. (1976). Scaling relations for earthquake source parameters and magnitudes, *Bull. Seismol. Soc. Am.* **66**, no. 5, 1501–1523, doi: [10.1785/BSSA0660051501](https://doi.org/10.1785/BSSA0660051501).
- Geng, J., Y. Pan, X. Li, J. Guo, J. Liu, X. Chen, and Y. Zhang (2018). Noise characteristics of high-rate multi-GNSS for subdaily crustal deformation monitoring, *J. Geophys. Res. Solid Earth* **123**, no. 2, 1987–2002, doi: [10.1002/2018JB015527](https://doi.org/10.1002/2018JB015527).
- Goldberg, D. E., and K. L. Haynie (2022). Ready for real time: Performance of global navigation satellite system in 2019 M_w 7.1 Ridgecrest, California, rapid response products, *Seismol. Res. Lett.* **93**, no. 2A, 517–530, doi: [10.1785/0220210278](https://doi.org/10.1785/0220210278).
- Goldberg, D. E., D. Melgar, V. J. Sahakian, A. M. Thomas, X. Xu, B. W. Crowell, and J. Geng (2020). Complex rupture of an immature fault zone: A simultaneous kinematic model of the 2019 Ridgecrest, CA earthquakes, *Geophys. Res. Lett.* **47**, no. 3, e2019GL086382, doi: [10.1029/2019GL086382](https://doi.org/10.1029/2019GL086382).
- Graves, R., and A. Pitarka (2015). Refinements to the Graves and Pitarka (2010) broadband ground-motion simulation method, *Seismol. Res. Lett.* **86**, no. 1, 75–80, doi: [10.1785/0220140101](https://doi.org/10.1785/0220140101).
- Graves, R. W., and A. Pitarka (2010). Broadband ground-motion simulation using a hybrid approach, *Bull. Seismol. Soc. Am.* **100**, no. 5A, 2095–2123, doi: [10.1785/0120100057](https://doi.org/10.1785/0120100057).
- Gutenberg, B., and C. F. Richter (1949). *Seismicity of the Earth and Associated Phenomena*. Princeton University Press, Princeton, New Jersey.
- Hoshiba, M., and T. Ozaki (2014). Earthquake early warning and tsunami warning of the Japan Meteorological Agency, and their performance in the 2011 off the Pacific coast of Tohoku earthquake (M_w 9.0), in *Early Warning for Geological Disasters: Advanced Technologies in Earth Sciences*, F. Wenzel and J. Zschau (Editors), Springer, 1–12, doi: [10.1007/978-3-642-12233-0_1](https://doi.org/10.1007/978-3-642-12233-0_1).
- Kanamori, H. (2005). Real-time seismology and earthquake damage mitigation, *Annu. Rev. Earth Planet. Sci.* **33**, 195–214, doi: [10.1146/annurev.earth.33.092203.122626](https://doi.org/10.1146/annurev.earth.33.092203.122626).
- Kawamoto, S., Y. Ohta, Y. Hiyama, M. Todoriki, T. Nishimura, T. Furuya, Y. Sato, T. Yahagi, and K. Miyagawa (2017). REGARD: A new GNSS-based real-time finite fault modeling system for GEONET, *J. Geophys. Res. Solid Earth* **122**, no. 2, 1324–1349, doi: [10.1002/2016JB013485](https://doi.org/10.1002/2016JB013485).
- Kingma, D. P., and J. Ba (2015). Adam: A method for stochastic optimization, *3rd International Conference on Learning Representations*, San Diego, California, doi: [10.48550/arXiv.1412.6980](https://doi.org/10.48550/arXiv.1412.6980).
- LeVeque, R. J., K. Waagan, F. I. González, D. Rim, and G. Lin (2016). Generating random earthquake events for probabilistic tsunami hazard assessment, *Pure Appl. Geophys.* **173**, 3671–3692, doi: [10.1007/s00024-016-1357-1](https://doi.org/10.1007/s00024-016-1357-1).
- Licciardi, A., Q. Blettery, B. Rouet-Leduc, J.-P. Ampuero, and K. Juhel (2022). Instantaneous tracking of earthquake growth with elastogravity signals, *Nature* **606**, 319–324, doi: [10.1038/s41586-022-04672-7](https://doi.org/10.1038/s41586-022-04672-7).
- Lin, J.-T., D. Melgar, V. J. Sahakian, A. M. Thomas, and J. Searcy (2023). Real-time fault tracking and ground motion prediction for large earthquakes with HR-GNSS and deep learning, *J. Geophys. Res. Solid Earth* **128**, no. 12, e2023JB027255, doi: [10.1029/2023JB027255](https://doi.org/10.1029/2023JB027255).
- Lin, J.-T., D. Melgar, A. M. Thomas, and J. Searcy (2021). Early warning for great earthquakes from characterization of crustal deformation patterns with deep learning, *J. Geophys. Res. Solid Earth* **126**, no. 10, e2021JB022703, doi: [10.1029/2021JB022703](https://doi.org/10.1029/2021JB022703).
- Lomax, A., A. Michelini, and D. Jozinović (2019). An investigation of rapid earthquake characterization using single-station waveforms and a convolutional neural network, *Seismol. Res. Lett.* **90**, no. 2A, 517–529, doi: [10.1785/0220180311](https://doi.org/10.1785/0220180311).
- Maechling, P. J., F. Silva, S. Callaghan, and T. H. Jordan (2015). SCEC broadband platform: System architecture and software implementation, *Seismol. Res. Lett.* **86**, no. 1, 27–38, doi: [10.1785/0220140125](https://doi.org/10.1785/0220140125).
- McGuire, J. J., C. W. Ulberg, A. I. Lux, M. Böse, J. R. Andrews, D. E. Smith, B. W. Crowell, J. R. Murray, I. Henson, R. Hartog, et al. (2025). ShakeAlert® version 3: Expected performance in large earthquakes, *Bull. Seismol. Soc. Am.* **115**, no. 2, 533–561, doi: [10.1785/0120240189](https://doi.org/10.1785/0120240189).
- Melbourne, T. I., W. M. Szeliga, V. Marcelo Santillan, and C. W. Scrivner (2021). Global navigational satellite system seismic monitoring, *Bull. Seismol. Soc. Am.* **111**, no. 3, 1248–1262, doi: [10.1785/0120200356](https://doi.org/10.1785/0120200356).
- Melgar, D., and G. P. Hayes (2017). Systematic observations of the slip pulse properties of large earthquake ruptures, *Geophys. Res. Lett.* **44**, no. 19, 9691–9698, doi: [10.1002/2017GL074916](https://doi.org/10.1002/2017GL074916).
- Melgar, D., B. W. Crowell, J. Geng, R. M. Allen, Y. Bock, S. Riquelme, E. M. Hill, M. Protti, and A. Ganas (2015). Earthquake magnitude calculation without saturation from the scaling of peak ground displacement, *Geophys. Res. Lett.* **42**, no. 13, 5197–5205, doi: [10.1002/2015GL064278](https://doi.org/10.1002/2015GL064278).
- Melgar, D., B. W. Crowell, T. I. Melbourne, W. Szeliga, M. Santillan, and C. Scrivner (2020). Noise characteristics of operational real-time high-rate GNSS positions in a large aperture network, *J. Geophys. Res. Solid Earth* **125**, no. 7, e2019JB019197, doi: [10.1029/2019JB019197](https://doi.org/10.1029/2019JB019197).
- Melgar, D., R. J. LeVeque, D. S. Dreger, and R. M. Allen (2016). Kinematic rupture scenarios and synthetic displacement data: An example application to the Cascadia subduction zone, *J. Geophys. Res. Solid Earth* **121**, no. 9, 6658–6674, doi: [10.1002/2016JB013314](https://doi.org/10.1002/2016JB013314).
- Melgar, D., T. I. Melbourne, B. W. Crowell, J. Geng, W. Szeliga, C. Scrivner, M. Santillan, and D. E. Goldberg (2019). Real-time high-rate GNSS displacements: Performance demonstration during the 2019 Ridgecrest, California, earthquakes, *Seismol. Res. Lett.* **91**, no. 4, 1943–1951, doi: [10.1785/0220190223](https://doi.org/10.1785/0220190223).
- Mena, B., P. M. Mai, K. B. Olsen, M. D. Purvance, and J. N. Brune (2010). Hybrid broadband ground-motion simulation using scattering Green's functions: Application to large-magnitude events, *Bull. Seismol. Soc. Am.* **100**, no. 5A, 2143–2162, doi: [10.1785/0120080318](https://doi.org/10.1785/0120080318).
- Mousavi, S. M., and G. C. Beroza (2020a). Bayesian-deep-learning estimation of earthquake location from single-station observations, *IEEE Trans. Geosci. Remote Sens.* **58**, no. 11, 8211–8224, doi: [10.1109/TGRS.2020.2988770](https://doi.org/10.1109/TGRS.2020.2988770).
- Mousavi, S. M., and G. C. Beroza (2020b). A machine-learning approach for earthquake magnitude estimation, *Geophys. Res. Lett.* **47**, no. 1, e2019GL085976, doi: [10.1029/2019GL085976](https://doi.org/10.1029/2019GL085976).
- Mousavi, S. M., and G. C. Beroza (2023). Machine learning in earthquake seismology, *Annu. Rev. Earth Planet. Sci.* **51**, no. 1, 105–129, doi: [10.1146/annurev-earth-071822-100323](https://doi.org/10.1146/annurev-earth-071822-100323).
- Mousavi, S. M., W. L. Ellsworth, W. Zhu, L. Y. Chuang, and G. C. Beroza (2020). Earthquake transformer—An attentive deep-learning model for simultaneous earthquake detection and phase picking, *Nat. Commun.* **11**, 3952, doi: [10.1038/s41467-020-17591-w](https://doi.org/10.1038/s41467-020-17591-w).

- Münchmeyer, J., D. Bindi, U. Leser, and F. Tilmann (2021). Earthquake magnitude and location estimation from real time seismic waveforms with a transformer network, *Geophys. J. Int.* **226**, no. 2, 1086–1104, doi: [10.1093/gji/ggab139](https://doi.org/10.1093/gji/ggab139).
- Murray, J. R., B. W. Crowell, M. H. Murray, C. W. Ulberg, J. J. McGuire, M. A. Aranha, and M. T. Hagerty (2023). Incorporation of real-time earthquake magnitudes estimated via peak ground displacement scaling in the ShakeAlert earthquake early warning system, *Bull. Seismol. Soc. Am.* **113**, no. 3, 1286–1310, doi: [10.1785/0120220181](https://doi.org/10.1785/0120220181).
- Perol, T., M. Gharbi, and M. Denolle (2018). Convolutional neural network for earthquake detection and location, *Sci. Adv.* **4**, no. 2, e1700578, doi: [10.1126/sciadv.1700578](https://doi.org/10.1126/sciadv.1700578).
- Ristea, N.-C., and A. Radoi (2021). Complex neural networks for estimating epicentral distance, depth, and magnitude of seismic waves, *IEEE Trans. Geosci. Remote Sens.* **19**, 7502305, doi: [10.1109/LGRS.2021.3059422](https://doi.org/10.1109/LGRS.2021.3059422).
- Ronneberger, O., P. Fischer, and T. Brox (2015). U-Net: Convolutional networks for biomedical image segmentation, *Medical Image Computing and Computer-Assisted Intervention: Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI 2015)*, N. Navab, J. Hornegger, W. Wells, and A. Frangi (Editors), Springer, 234–241, doi: [10.1007/978-3-319-24574-4_28](https://doi.org/10.1007/978-3-319-24574-4_28).
- Ross, Z. E., B. Idini, Z. Jia, O. L. Stephenson, M. Zhong, X. Wang, Z. Zhan, M. Simons, E. J. Fielding, S.-H. Yun, *et al.* (2019). Hierarchical interlocked orthogonal faulting in the 2019 Ridgecrest earthquake sequence, *Science* **366**, no. 6463, 346–351, doi: [10.1126/science.aaz0109](https://doi.org/10.1126/science.aaz0109).
- Ross, Z. E., M.-A. Meier, E. Hauksson, and T. H. Heaton (2018). Generalized seismic phase detection with deep learning, *Bull. Seismol. Soc. Am.* **108**, no. 5A, 2894–2901, doi: [10.1785/0120180080](https://doi.org/10.1785/0120180080).
- Ruhl, C. J., D. Melgar, J. Geng, D. E. Goldberg, B. W. Crowell, R. M. Allen, Y. Bock, S. Barrientos, S. Riquelme, J. C. Baez, *et al.* (2018). A global database of strong-motion displacement GNSS recordings and an example application to PGD scaling, *Seismol. Res. Lett.* **90**, no. 1, 271–279, doi: [10.1785/0220180177](https://doi.org/10.1785/0220180177).
- Santillan, V. M., T. I. Melbourne, W. M. Szeliga, and C. W. Scrivner (2013). A fast-convergence stream editor for real-time precise point positioning, *Abstract G53B-0930 presented at American Geophysical Union Fall Meeting*, San Francisco, California, 9–13 December 2013.
- Shelly, D. R. (2020). A high-resolution seismic catalog for the initial 2019 Ridgecrest earthquake sequence: Foreshocks, aftershocks, and faulting complexity, *Seismol. Res. Lett.* **91**, no. 4, 1971–1978, doi: [10.1785/0220190309](https://doi.org/10.1785/0220190309).
- Southern California Earthquake Data Center (SCEDC) (2013). Southern California Earthquake Center [Data set], *Caltech*, doi: [10.7909/C3WD3xH1](https://doi.org/10.7909/C3WD3xH1).
- Thomas, A., D. Melgar, S. N. Dybing, and J. R. Searcy (2023). Deep learning for denoising high-rate Global Navigation Satellite System data, *Seismica* **2**, no. 1, 240, doi: [10.26443/seismica.v2i1.240](https://doi.org/10.26443/seismica.v2i1.240).
- Thomas, A. M., A. Inbal, J. Searcy, D. R. Shelly, and R. Bürgmann (2021). Identification of low-frequency earthquakes on the San Andreas fault with deep learning, *Geophys. Res. Lett.* **48**, no. 13, e2021GL093157, doi: [10.1029/2021GL093157](https://doi.org/10.1029/2021GL093157).
- Trugman, D. T., M. T. Page, S. E. Minson, and E. S. Cochran (2019). Peak ground displacement saturates exactly when expected: Implications for earthquake early warning, *J. Geophys. Res. Solid Earth* **124**, no. 5, 4642–4653, doi: [10.1029/2018JB017093](https://doi.org/10.1029/2018JB017093).
- van den Ende, M. P. A., and J.-P. Ampuero (2020). Automated seismic source characterization using deep graph neural networks, *Geophys. Res. Lett.* **47**, no. 17, e2020GL088690, doi: [10.1029/2020GL088690](https://doi.org/10.1029/2020GL088690).
- Zhang, X., J. Zhang, C. Yuan, S. Liu, Z. Chen, and W. Li (2020). Locating induced earthquakes with a network of seismic stations in Oklahoma via a deep learning method, *Sci. Rep.* **10**, 1941, doi: [10.1038/s41598-020-58908-5](https://doi.org/10.1038/s41598-020-58908-5).
- Zhu, L. P., and L. A. Rivera (2002). A note on the dynamic and static displacements from a point source in multilayered media, *Geophys. J. Int.* **148**, no. 3, 619–627, doi: [10.1046/j.1365-246X.2002.01610.x](https://doi.org/10.1046/j.1365-246X.2002.01610.x).
- Zhu, W., and G. C. Beroza (2019). PhaseNet: A deep-neural-network-based seismic arrival-time picking method, *Geophys. J. Int.* **216**, no. 1, 261–273, doi: [10.1093/gji/ggy423](https://doi.org/10.1093/gji/ggy423).

Manuscript received 10 September 2025

Published online 23 July 2026