



BIROn - Birkbeck Institutional Research Online

Lapins, S. and Goitom, B. and Kendall, J.-M. and Werner, M.J. and Cashman, K.V. and Hammond, James O.S. (2021) A little data goes a long way: automating seismic phase arrival picking at Nabro Volcano with transfer learning. *Journal of Geophysical Research: Solid Earth* 126 (7), e2021JB021910. ISSN 0148-0227.

Downloaded from: <https://eprints.bbk.ac.uk/id/eprint/45241/>

Usage Guidelines:

Please refer to usage guidelines at <https://eprints.bbk.ac.uk/policies.html>
contact lib-eprints@bbk.ac.uk.

or alternatively

JGR Solid Earth

RESEARCH ARTICLE

10.1029/2021JB021910

Special Section:

Machine learning for Solid Earth observation, modeling and understanding

Key Points:

- Transfer learning using existing model trained on California earthquake data produces effective new model for monitoring at Nabro volcano
- Nabro transfer learning model shows improved S-wave picking resulting in smaller location errors than even manual phase picks
- Changing task from classification to segmentation results in more efficient model processing 14 months of data from seven stations in 4 h

Supporting Information:

Supporting Information may be found in the online version of this article.

Correspondence to:

S. Lapins,
sacha.lapins@bristol.ac.uk

Citation:

Lapins, S., Goitom, B., Kendall, J.-M., Werner, M. J., Cashman, K. V., & Hammond, J. O. S. (2021). A little data goes a long way: Automating seismic phase arrival picking at Nabro volcano with transfer learning. *Journal of Geophysical Research: Solid Earth*, 126, e2021JB021910. <https://doi.org/10.1029/2021JB021910>

Received 23 FEB 2021

Accepted 21 JUN 2021

© 2021. The Authors.

This is an open access article under the terms of the [Creative Commons Attribution License](https://creativecommons.org/licenses/by/4.0/), which permits use, distribution and reproduction in any medium, provided the original work is properly cited.

A Little Data Goes a Long Way: Automating Seismic Phase Arrival Picking at Nabro Volcano With Transfer Learning

Sacha Lapins¹ , Berhe Goitom¹, J-Michael Kendall² , Maximilian J. Werner¹ , Katharine V. Cashman¹, and James O. S. Hammond³ 

¹School of Earth Sciences, University of Bristol, Bristol, UK, ²Department of Earth Sciences, University of Oxford, Oxford, UK, ³Department of Earth and Planetary Sciences, Birkbeck, University of London, London, UK

Abstract Supervised deep learning models have become a popular choice for seismic phase arrival detection. However, they do not always perform well on out-of-distribution data and require large training sets to aid generalization and prevent overfitting. This can present issues when using these models in new monitoring settings. In this work, we develop a deep learning model for automating phase arrival detection at Nabro volcano using a limited amount of training data (2,498 event waveforms recorded over 35 days) through a process known as transfer learning. We use the feature extraction layers of an existing, extensively trained seismic phase picking model to form the base of a new all-convolutional model, which we call U-GPD. We demonstrate that transfer learning reduces overfitting and model error relative to training the same model from scratch, particularly for small training sets (e.g., 500 waveforms). The new U-GPD model achieves greater classification accuracy and smaller arrival time residuals than off-the-shelf applications of two existing, extensively-trained baseline models for a test set of 800 event and noise waveforms from Nabro volcano. When applied to 14 months of continuous Nabro data, the new U-GPD model detects 31,387 events with at least four P-wave arrivals and one S-wave arrival, which is more than the original base model (26,808 events) and our existing manual catalog (2,926 events), with smaller location errors. The new model is also more efficient when applied as a sliding window, processing 14 months of data from seven stations in less than 4 h on a single graphics processing unit.

Plain Language Summary Seismic monitoring increasingly relies on automated signal processing as the rate of data acquisition grows. Supervised deep learning models have proven to be effective for detecting and characterizing seismic events, but training such highly parameterized models generally requires large amounts of manually labeled data. Once trained, however, these models extract general seismic waveform features that can be used to train new models with more limited training data. In this work, we use the generalized knowledge of seismic data from a model trained on millions of earthquakes in California to train a new model for detecting volcanic earthquakes at Nabro volcano, Eritrea, a recently active and, prior to its 2011 eruption, poorly monitored volcano. Using a small training set of waveforms, the new model more accurately detects phase arrivals and noise than off-the-shelf applications of two baseline models. The new model is efficient, processing 14 months of data in less than 4 h. It is also effective, detecting more volcanic events and showing improved levels of S-wave arrival picking. The result is smaller event location errors than even our manual picks. This level of efficiency and consistency highlights the role that machine learning can play in volcano-seismic monitoring.

1. Introduction

Seismic monitoring plays a fundamental part in mitigating hazards at volcanoes. During periods of unrest, thousands of earthquakes can occur each day, producing a diverse range of seismic signals that reflect a multitude of interlinked volcanic processes (e.g., migrating fluids, fault movement, explosions, rockfalls). These earthquakes are generally recorded by broadband seismometers, which are highly sensitive to ground motion across a wide range of frequencies and record signals at high sample rates (typically 100 times or more per second). This level of detail, however, comes at the cost of generating vast amounts of data. Many seismic networks utilize tens or even hundreds of seismometers at a given time (e.g., Hansen & Schmandt, 2015), making real-time manual inspection of these time series practically infeasible. Previous

seismic deployments have also generated extensive legacy datasets that can offer insights into historical volcanic activity and opportunities to further our understanding of volcanic processes. The main challenge is therefore to identify and characterize volcanic earthquakes in a robust and timely manner so as to provide vital clues regarding the state of a volcano and the likelihood or impact of an eruption or hazard, as well as be able to accurately and efficiently process large existing datasets for further analysis within a reasonable timeframe.

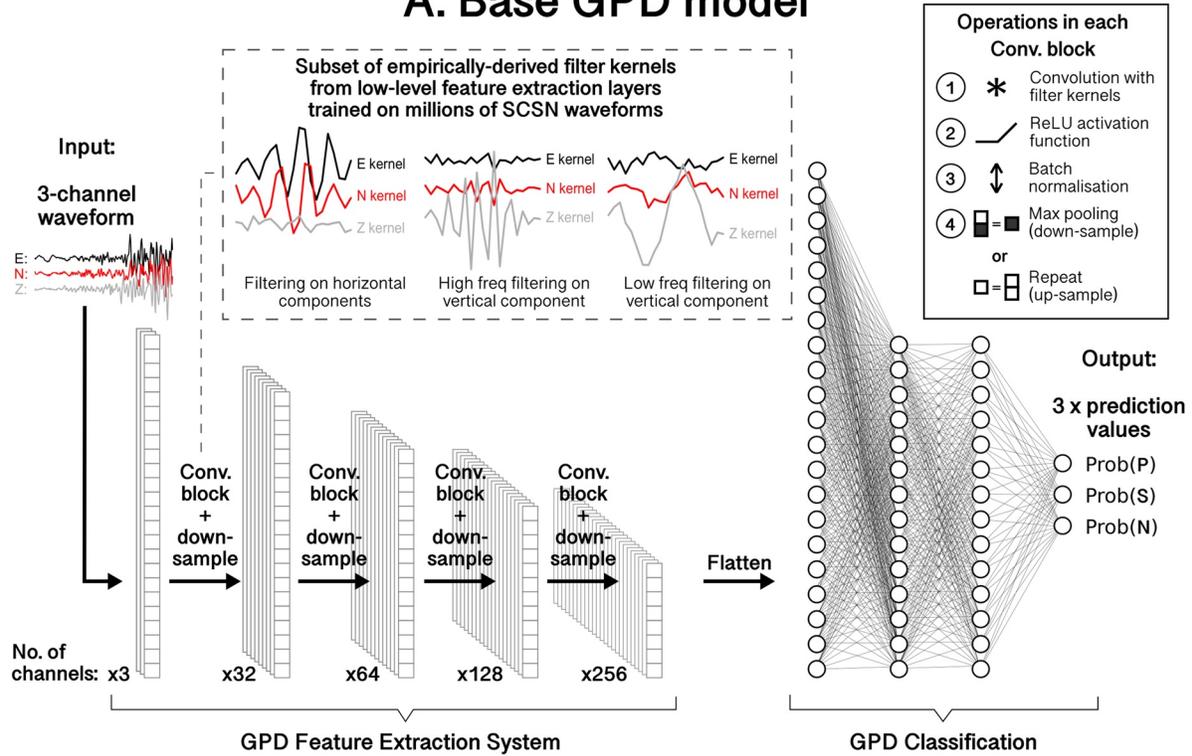
Identifying earthquake phase arrivals, particularly the initial primary (P-) and secondary/shear (S-) wave arrivals, forms the basis of most seismic processing tasks (e.g., determining locations, magnitudes and source parameters). Manually identifying these phase arrivals yields greater accuracy and estimates of arrival time uncertainty than automated approaches but is extremely time-consuming. Alternatively, most automated approaches are orders of magnitude quicker but typically require clear phase arrivals, existing “templates” of previously cataloged earthquakes (e.g., Gibbons & Ringdal, 2006; Lengliné et al., 2016; Shelly et al., 2007), or pre-processing/feature extraction steps calibrated for a small set of earthquake characteristics (e.g., trigger algorithms based on the ratio of short-term average to long-term average signal amplitude, STA/LTA; Withers et al., 1998). A challenge for application to volcanology is that volcanic earthquakes can exhibit widely varying time-frequency characteristics, often with low amplitudes or obscured phase arrivals, and new phases of unrest can produce previously unseen seismic signals that differ from existing earthquake templates. Furthermore, methods based on existing seismic catalogs are unsuitable for new seismic deployments where a catalog of events has not been collected.

A recently successful approach for seismic phase arrival detection is the use of supervised deep learning models (e.g., Dokht et al., 2019; Mousavi et al., 2019; Ross, Meier, Hauksson, & Heaton, 2018; Woollam et al., 2019; Zhu & Beroza, 2019). These methods are based on convolutional neural networks (CNN), a variant of classical neural networks that employ convolution operations, as opposed to matrix multiplication, in at least part of the model. These operations are employed in “hidden” convolutional layers that allow the network to learn a large set of filters to extract useful features from the input data and map them to a desired output (e.g., to identify phase arrivals in earthquake waveforms; Figure 1). Typically, multiple convolutional layers are applied in succession and in combination with other operations, such as nonlinear “activation,” down-sampling and normalization, to extract complex patterns from the data using a hierarchy of simpler filter kernels. These extracted features can then be fed into a standard fully-connected neural network or other machine learning architecture for classification, segmentation, regression, clustering, or inference (e.g., Mousavi et al., 2019; Ross, Meier, Hauksson, & Heaton, 2018; van den Ende & Ampuero, 2020). As such, the “convolutional” part of CNNs act as the model’s feature extraction system. With each successive convolutional layer, the extracted features move from lower-level, general signal features (resembling, e.g., long/short period wavelets in seismological waveform models; Figure 1a inset) to more task specific, high-level features (Yosinski et al., 2014). The final “classification” layers of the model map these features to the desired output and can be considered the most task specific part of the model, empirically tuned to the distribution of the training data (Yosinski et al., 2014).

Such an approach gives supervised deep learning models a strong advantage over traditional algorithms that require considerable manual intervention or rely on a small set of manually determined characteristics and simple threshold criteria. In general, however, these models require substantial amounts of labeled data during training to generalize to out-of-sample data (the amount dependent on various factors, such as network architecture, number of network parameters and training hyperparameters; e.g., D’souza et al., 2020; He et al., 2019; Sun et al., 2017). In the case of seismological supervised models, these models can demonstrate impressive levels of generalization to phase arrival detection in other geographic and tectonic settings, if trained with sufficient data (e.g., Mousavi et al., 2020; Tan et al., 2021). However, as with practically any deep learning model, they can also suffer significant loss in performance when faced with data that differs in source or distribution from their training data (e.g., Barbedo, 2018; Zech et al., 2018, Figure 7). As such, the requirement for extensive training sets can place the traditional paradigm of supervised learning (i.e., using a large amount of hand-labeled data to train a single model for a desired domain or problem) out of reach for many real-world applications.

Transfer learning is based on the idea of knowledge transfer from one task to another (Pan & Yang, 2010; Zhuang et al., 2020) and can be a powerful tool when we do not have sufficient labeled data to train a reliable

A. Base GPD model



B. U-GPD Transfer Learning model

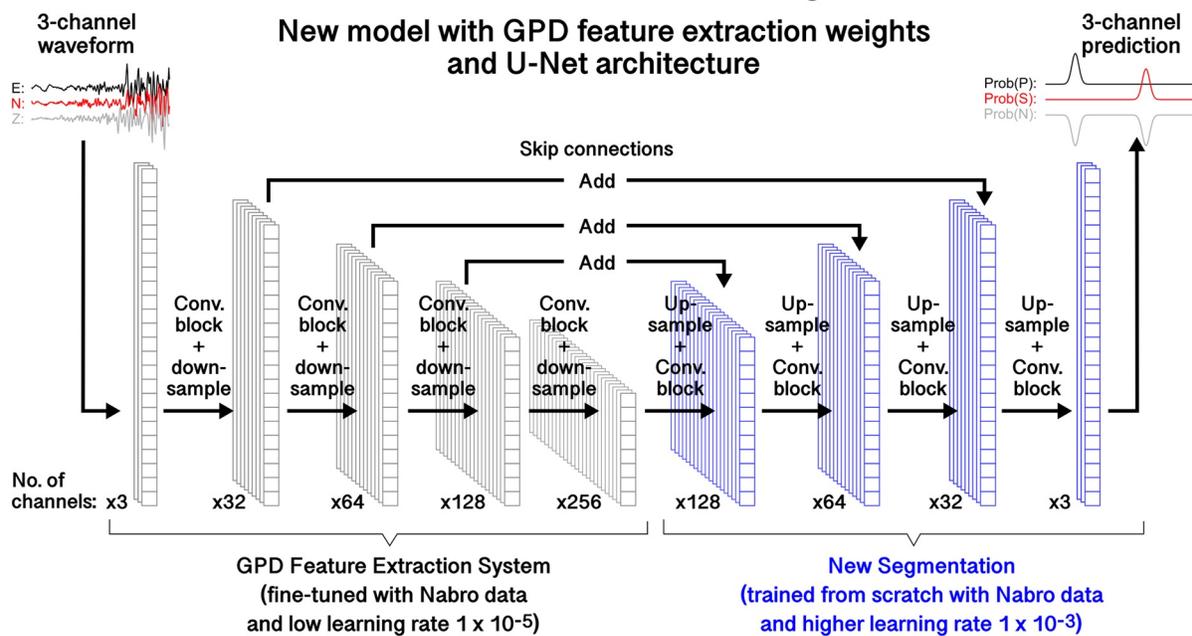


Figure 1

model from scratch, or when existing models perform poorly. At its simplest, the first n convolutional layers and their weights from the feature extraction part of an existing model are copied to the first n layers of a new model for a related or similar task, with the remaining layers either re-initialized with randomized weights or replaced (e.g., Razavian et al., 2014; Yosinski et al., 2014). These tasks need not be near-identical or even superficially related, as long as low-level data characteristics are shared between tasks (e.g., Efremova et al., 2019; Tran et al., 2020; Zamir et al., 2018). The intuition is that generalized knowledge of data structure and properties from one model trained with abundant labeled data (or “big data”) can guide a learning algorithm toward a good solution for a new task with far more limited, or even no, labeled data.

In this study, we evaluate the utility of inductive transfer learning (i.e., when labeled data are available for both the source and target tasks) for small seismic training sets and produce a deep learning model that accurately and robustly picks phase arrivals from a deployment at Nabro volcano in Eritrea, a region with little or no prior seismic monitoring. We leverage the knowledge acquired from training a model on millions of seismic waveforms recorded by the Southern California Seismic Network (SCSN), hereby referred to as the GPD model (Generalized seismic Phase Detection; Ross, Meier, Hauksson, & Heaton, 2018), and apply it to seismograms from Nabro volcano in Eritrea, for which we have limited hand-labeled data (manual phase arrival picks) from the first couple of months of a 14-months seismic deployment (Goitom, 2017; Hamlyn et al., 2014). The new model task differs from the original GPD model task in that it is modified from one of *classification* (assigning a single class label P-wave, S-wave or *noise* to an entire 4-second waveform; Figure 1a) to one of *segmentation* (assigning a class label P-wave, S-wave or *noise* to *each datapoint* within that 4-second waveform; Figure 1b). We achieve this by replacing the fully-connected uppermost layers of the original GPD model with further convolutional layers, creating an all-convolutional model commonly referred to as a U-Net (Ronneberger et al., 2015). We refer to this specific model design as the U-GPD model, utilizing GPD model weights within a U-Net architecture. The new data from Nabro volcano also exhibit differences in instrument calibration and sample rates from the original GPD model training data, as well as differing waveform characteristics between tectonic and volcanic event types (Lahr et al., 1994; Lapins et al., 2020; McNutt & Roman, 2015).

In the following section, we introduce transfer learning and recent applications in seismological deep learning. In Sections 3 and 4, we present our proposed transfer learning method, U-GPD model architecture and seismic data recorded at Nabro volcano. In Section 5, we present a series of model comparisons. We first use common training metrics to demonstrate that transfer learning reduces overfitting and model error, particularly for very small training sets (<1,000 waveforms), when compared with a model reinitialized with randomized weights before training (i.e., trained from scratch with no transfer learning). We then apply these new models to a test data set of known P-/S-wave arrivals and sections of noise and compare performance with off-the-shelf applications of the base GPD model and another extensively-trained phase-picking model, PhaseNet (Zhu & Beroza, 2019). We find that the U-GPD transfer learning model yields improved phase arrival identification, particularly for S-waves, and false detection rate at Nabro volcano. Altering the model task from classification to segmentation also improves pick time residuals over the base GPD model for these test data. Finally, we apply both our new U-GPD transfer learning model and the original base GPD model to the full 14-month seismic deployment at Nabro volcano through a sliding window approach. The new U-GPD model identifies more useable S-wave arrivals than the base GPD model, yielding smaller subsequent location errors than even our manual analyst’s phase arrival picks. The new model also runs an order of magnitude faster, processing 14 months of data from 7 broadband seismometers in less than 4 h on a single GPU. Our findings indicate that transfer learning can be extremely useful for volcano seismic monitoring, even with limited computing resources and data. We conclude this study with a discussion of

Figure 1. (a) Model architecture for Generalized seismic Phase Detection (GPD) CNN model (Ross, Meier, Hauksson, & Heaton, 2018). Model can be considered as two parts: a feature extraction system (convolutional layers) and classification part (fully connected layers). GPD model outputs $3 \times$ prediction values (probability of P, S, or noise) for an entire 400-sample 3-component waveform (i.e., output dimensions: 1×3). Examples of filter kernels (dashed line inset) from lowest convolutional layer that extract generalized seismic waveform features determined through model training on extensive SCSN dataset. These indicate that the GPD model has learned to extract different features from vertical and horizontal components. (b) Proposed transfer learning model architecture (“U-GPD”). GPD model feature extraction system is copied to new model and fine-tuned with new Nabro data and low learning rate. Low learning rate ensures that useful features are not “unlearned.” New convolutional layers replace the GPD classification layers and are trained using new Nabro data and higher learning rate. Model outputs $3 \times$ prediction values for each datapoint in 400-sample three-component waveform (i.e., output dimensions: 400×3).

our findings, methodology and practical considerations of transfer learning in Section 6. All data and code used throughout this study are made fully and publicly available (see *Data Availability Statement*).

2. Transfer Learning

There are many approaches to transfer learning (see Pan & Yang, 2010; Zhuang et al., 2020 for comprehensive surveys), including using “off-the-shelf” feature extraction systems from existing state-of-the-art CNNs (e.g., Maqsood et al., 2019; Razavian et al., 2014), learning domain-invariant or global representations across multiple tasks (e.g., Glorot et al., 2011; W. Li et al., 2014; Tzeng et al., 2015; Zhuang et al., 2015), applying pre-processing steps to make input data representations more similar between datasets (e.g., Daumé, 2007; Sun et al., 2016) and the use of domain-adversarial models (e.g., Ganin et al., 2016). Here we employ the first of these approaches for P- and S-wave arrival time picking at Nabro volcano, utilizing pre-trained filters from an existing, extensively trained CNN model (the GPD model; Ross, Meier, Hauksson, & Heaton, 2018) to train a new model with different output dimension and task type (see Section 3.1, *U-GPD Model Architecture*). Most seismological studies that have employed transfer learning in this way have used pre-trained filters from models designed for nonseismological tasks, such as image recognition. For example, filters trained to recognize photographic images or handwritten characters have been used to detect earthquakes and classify volcano-seismic event types from spectrograms (Huot et al., 2018; Lara et al., 2020; Titos et al., 2020) and interpret seismic facies (Dramsich & Lüthje, 2018).

Some studies have chosen to fine-tune entire seismic deep learning models, essentially updating the models with new data (or equivalently “pre-training” the models with larger datasets, depending on perspective). El Zini et al. (2020) pre-train an autoencoder with abundant unlabeled data to learn compressed data representations of 2D seismic images. These model weights then serve as a starting point for a model that segments seismic images, with weights fine-tuned using limited labeled training data. This approach was shown to outperform the transfer of weights from image recognition models and training a model from scratch. Bueno et al. (2020) fine-tune a Bayesian neural network (BNN) to improve classification of volcano-seismic event characteristics between datasets and time periods. They show that this approach increases model accuracy and reduces epistemic uncertainty when applied to new volcanic systems or phases of activity. With a similar aim but different approach to the work of this study, Chai et al. (2020) utilize pre-trained weights from another existing phase arrival detection model, PhaseNet (Zhu & Beroza, 2019), to pick phase arrivals from hydraulic fracturing experiments. They use the entirety of the PhaseNet model and its pre-trained weights as a starting point for training and then fine-tune all model weights equally using just 3,500 seismograms. They present improved results over the original PhaseNet model, which was trained using 700,000 seismograms of regional Californian seismicity, when applied to higher sample rate data (100 kHz) from a very different setting (i.e., hydraulic fracturing). While these studies show that fine-tuning entire models can be an effective strategy, poor hyperparameter choices (model learning rate, number of training epochs, etc.) can inadvertently retrain the model (also known as “catastrophic forgetting”; e.g., Kirkpatrick et al., 2017) or lead to settling on a nonglobal minimum within the parameter space, reopening the potential for overfitting when the number of model parameters is large and the training dataset is small (El Zini et al., 2020; Yosinski et al., 2014). The work in this study differs from that of Chai et al. (2020) in that only the weights from the feature extraction part (i.e., the first “half”) of the GPD model are transferred to our new U-GPD model. These weights are fine-tuned using a much lower learning rate (weight update step size) to retain useful learned knowledge from the original model but optimize cohesion with the rest of the new model, which is redesigned to reduce the total number of trainable parameters, among other optimizations (see Section 3.1, *Model Architecture*), and initialized with randomized weights (Figure 1).

3. Proposed Model

3.1. U-GPD Model Architecture

As outlined briefly above, we utilize pre-trained parameters from the convolutional layers of the GPD model as a starting point for our U-GPD transfer learning model. The original GPD model was trained using 4.5 million hand-labeled seismograms (1.5 million of each class P, S, and *noise*) recorded by the Southern California Seismic Network (SCSN) between the years 2000 and 2017. These training data were all 400-sample

(4 s) 3-component waveforms, high-pass filtered above 2 Hz and (re)sampled at 100 Hz. All events had epicentral distances less than 100 km and magnitudes between -0.81 and $5.7 M$ (various magnitude scales). The GPD model was chosen as a base for our transfer learning model as these data characteristics (magnitude range, sample rate and event distances) are comparable to those observed and recorded by volcano observatories. Furthermore, the short input length of 4 s (400 samples at 100 Hz sample frequency) means there is less chance of erroneously labeling or missing relatively small magnitude or overlapping phase arrivals. Finally, the GPD model's "sequential" architecture, with each layer being solely connected to the layers directly before and after, also means the model is more interpretable and makes it easier to isolate its feature extraction system.

During model training, we fine-tune these pre-trained parameters using a very small learning rate (1×10^{-5}), rather than keep them fixed (e.g., Yosinski et al., 2014). Learning rate effectively controls how much model weights can change and a small learning rate will keep adjustments to the pre-trained GPD feature extraction weights small. The aim of this fine-tuning step is to modify any highly specific features from the source domain (particularly in the higher-level feature extraction layers) and overcome optimization difficulties arising from splitting the GPD convolutional layers from co-adapted classification layers (Yosinski et al., 2014), without unlearning the important generalized waveform features we wish to exploit.

We then replace the GPD model's fully-connected layers (i.e., the task-specific classification part of the model) with further convolutional layers and up-sampling operations, combined with ReLU activation function (Nair & Hinton, 2010) and batch normalization (Ioffe & Szegedy, 2015), to produce a model output with the same dimensions as model input (400 samples \times 3 channels; Figure 1b). Each of the three output channels represents the model's prediction (or "probability") of a P-wave arrival, S-wave arrival or neither (hereby referred to as *noise*), respectively, at each datapoint in the waveform. This all-convolutional approach has been adopted by other phase arrival picking models (e.g., Woollam et al., 2019; Zhu & Beroza, 2019) and has several distinct advantages when applied to seismic phase arrival detection: (i) it provides less ambiguous labeling of phase arrivals when compared to the original GPD model's approach of assigning a single class prediction (P, S or *noise*) to an entire 400-sample 3-channel waveform; (ii) convolutional layers tend to have fewer parameters than fully connected neural network layers so less training data is required to avoid overfitting; (iii) by producing a model with input and output traces of same dimension, we require less overlap when applied as a rolling window method, producing a model that runs orders of magnitude faster on continuous sections of data.

The new convolutional layers are initialized with completely randomized weights and trained with a higher learning rate (1×10^{-3}) than the pre-trained GPD weights. A higher learning rate effectively allows the randomized weights in the new model layers to be adjusted much more than the pre-trained GPD weights. The learning rates used for each part of the model were determined through experimentation, insight from previous works (e.g., Ross, Meier, & Hauksson, 2018; Ross, Meier, Hauksson, & Heaton, 2018), and on the basis that the learning rate for fine-tuning pre-trained weights should be orders of magnitude lower than that used for tuning randomized weights (e.g., Yosinski et al., 2014). We note that there are more formal strategies (e.g., grid/random search, Bayesian optimization, bandit strategies, gradient reversal; Bergstra & Bengio, 2012; Feurer et al., 2015; Klein et al., 2017; Maclaurin et al., 2015; Snoek et al., 2015) for determining optimal model hyperparameters. Such strategies, however, add significant computational cost as they generally require repeatedly training models with differing hyperparameter choices, producing a much greater search space. The aim in this study is not to present the absolute best possible model architecture and set of hyperparameters specific to this deployment at Nabro (as these choices will likely be specific to application and training set size) but to illustrate how existing models can be tailored to new datasets to improve performance in those settings. Furthermore, it would prove more difficult to attribute any observed improvements to the use of transfer learning and U-Net architecture, as opposed to the hyperparameter optimization strategy. We do, however, implement two further hyperparameter choices that were found to improve performance. First, we use dilated filter kernels in the new convolutional layers (e.g., van den Oord et al., 2016; Yu & Koltun, 2016) to increase the size of the model's receptive field (or "field of view") and aggregate multi-scale context. Second, the new layers are subjected to spatial dropout (Tompson et al., 2015), where 30% of the feature maps (output of filter operations) in each convolutional layer are effectively dropped (set to zero) at the start of each training epoch. This step promotes independence between the features the model

extracts and prevents overfitting (Tompson et al., 2015). Precise details of U-GPD model dimensions and hyperparameters are provided in Supplementary Materials (Figure S1).

The overall network architecture outlined above is sometimes referred to as a U-Net (Ronneberger et al., 2015). With each step through the network, the input data are progressively downsampled with an increasing number of features extracted, creating a contracting network path that is forced to sacrifice detail and learn a more compressed, general representation of the input waveform to discriminate between classes (P, S, or *noise*). The model then follows a symmetrically expanding path, where the data are progressively upsampled and the number of features reduced, to regain precise temporal or spatial detail and return an output with equal dimension to the model input (Ronneberger et al., 2015). Skip connections (addition operators), which act as direct, one-way pathways between layers in the contracting and expanding sides of the model (Figure 1b), are used to retain precise waveform details that may be lost through this contraction/expansion process and have been shown to greatly improve the likelihood of model parameters settling on the global minimum during training (H. Li et al., 2017).

3.2. Phase Arrival Labels and Model Hyperparameters

Each 3-component waveform in our training dataset has a corresponding 3-channel “mask” that provides a ground truth label (P, S, or *noise*) for each waveform datapoint. During training, the model aims to minimize the difference between its predictions and these ground truth labels. Labels are presented as binary values (0's or 1's), with P-wave arrivals indicated by a ± 0.14 s boxcar function, centered on the manually picked P-wave arrival time, and S-wave arrivals indicated by ± 0.19 s boxcar function, also centered on the manually picked S-wave arrival time. These boxcar widths provide a good balance between phase arrival detection rate and arrival time precision and compensate for human error in the ground truth labels. Previous studies have used Gaussian-style probability masks, with values ranging between 0 and 1, for labeling phase arrivals (e.g., Woollam et al., 2019; Zhu & Beroza, 2019). We find that label accuracy on our test data (e.g., Figures 5–7) and event location error distributions from the full deployment (e.g., Figure 10c and 10d) are near-identical when using either approach but training with boxcar masks produces a model that detects $\sim 10\%$ more events when run over continuous data.

As with the original GPD model, our new U-GPD model was trained using a categorical cross entropy loss function (Supporting Information Text S7) and the Adam optimization algorithm (Kingma & Ba, 2014). The model weights that produced lowest loss value on the validation dataset during training were selected as our final model weights. Other loss functions that address the imbalance between arrival and noise labels (as the majority of labels in any given waveform are not a phase arrival), such as a focal loss function that effectively adds weighting parameters to cross entropy loss (Lin et al., 2017), were trialed but yielded no improvement in model performance.

4. Data

Nabro volcano is one of two calderas that form the Bidu Volcanic Massif on the Eritrea-Ethiopia international border (Figure 2). Located in the Afar region at the northern end of the Main Ethiopian Rift, it erupted unexpectedly for the first time in recorded history on June 12th, 2011, disrupting continental aviation and initiating a significant humanitarian crisis (Bojanowski, 2011; Donovan et al., 2018; Goitom et al., 2015). At the time, there were no seismic or other monitoring networks operating in Eritrea but earthquakes were felt around the volcano several hours and days prior to eruption, prompting evacuation (Goitom et al., 2015). This seismicity is the first of note in global catalogs for the region (Goitom et al., 2015). Despite this fortuitous warning, at least seven people were tragically killed and about 12,000 were displaced (Bojanowski, 2011; Goitom et al., 2015; Hamlyn et al., 2014). The eruption is particularly notable for the vast amount of SO₂ emitted into the atmosphere, one of the largest eruptive SO₂ masses globally since the eruption of Mount Pinatubo in 1991 (Fromm et al., 2014; Goitom et al., 2015; Theys et al., 2013), and the comparative rarity of recorded historical eruptions in the region (Goitom et al., 2015; Hamlyn et al., 2014).

In August 2011, approximately two months after the eruption began, eight 3-component broadband seismometers ($5 \times$ Guralp CMG-6T, $3 \times$ Guralp CMG-40T; Figure 2) were deployed around the volcano to monitor ongoing activity (Hamlyn et al., 2014; Hammond et al., 2011). These stations remained operational

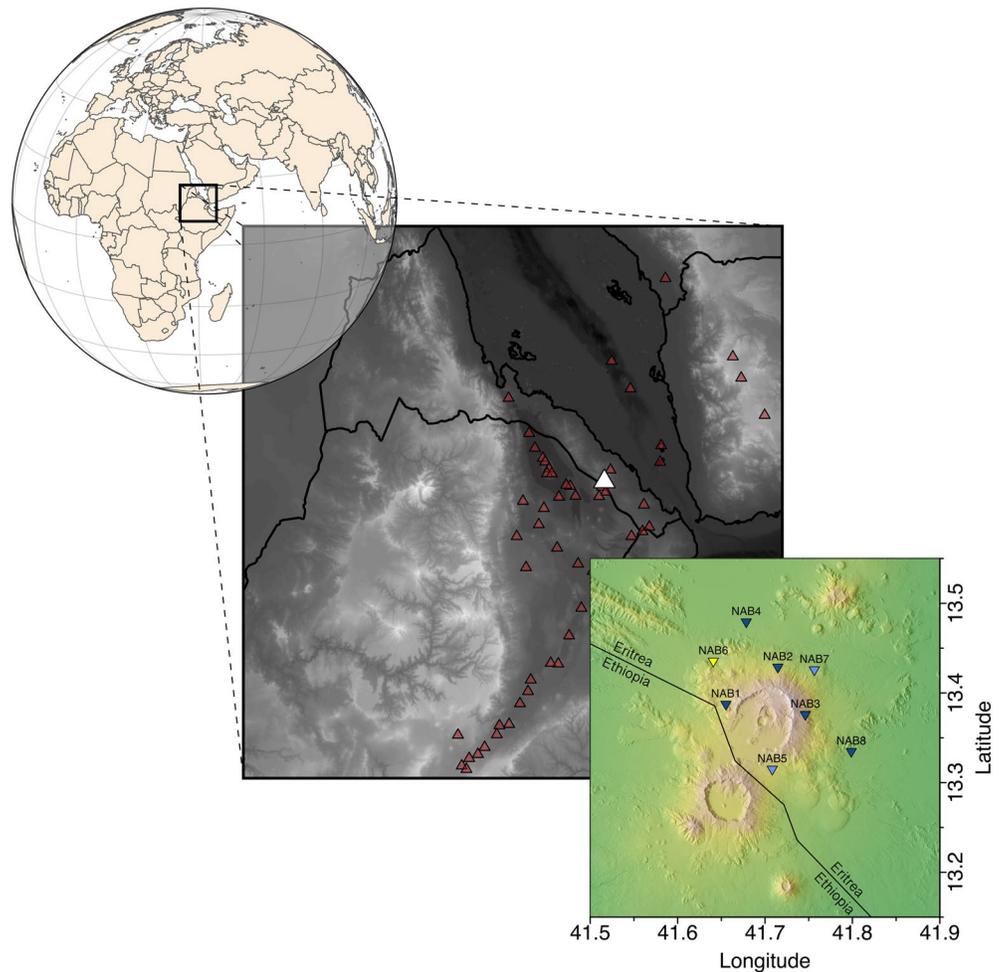


Figure 2. Regional topographic map (90 m CGIAR Shuttle Radar Topography Mission and GEBCO bathymetry model, gray-scale map center) and seismic deployment (30 m ALOS Digital Surface Model, color map bottom right) around Nabro volcano. Red triangles (center map) indicate Holocene volcanoes (Global Volcanism Program, 2013) with Nabro volcano highlighted in white. Inverted blue triangles (bottom right map) indicate operational broadband seismic stations deployed around Nabro volcano from August 2011 to October 2012 (station NAB6, inverted yellow triangle, was flooded shortly after deployment and not operational). Training and validation data were taken from dark blue stations only (NAB1, NAB2, NAB3, NAB4, and NAB8).

for 14 months until October 2012. The first two months of data were collected at a sample rate of 100 Hz before dataloggers were switched to a sample rate of 50 Hz for the remainder of the deployment to maximize data recovery while minimizing service runs. Data from the full deployment occupies 70 GB of disk space (miniSEED format). Manual phase arrival picking conducted on the first four months of data (2011-08-30 to 2011-12-31; Goitom, 2017; Hamlyn et al., 2014) identified a total of 2,926 events, from which the first 35 days of data (all 100 Hz sample rate) were quality checked and used for training and validating our transfer learning model. Five subsequent days of data (2×100 Hz days, 3×50 Hz days) were selected and quality checked to serve as test data. The reason to exclude 50 Hz data from model training is to emulate data availability in the early stages of this seismic deployment and demonstrate that changes in sample rate can be overcome without compiling new training datasets through a process known as data augmentation. The raw data for all datasets (training, validation, and testing) were self-normalized, with linear trend removed, and left unfiltered.

A total of 2,921 waveforms with labeled P- and S-wave arrivals from 978 events (2011-08-30 to 2011-10-03) and five stations were used as training and validation data (only five stations were consistently operational during this time; dark blue stations in Figure 2 bottom right map). Training and validation data were

grouped and divided so that no event appeared in both datasets to avoid data leakage (the model being trained on event data that also appears in validation or testing). Eight hundred fifty-seven events (2,498 waveforms) were used for model training and 121 events (423 waveforms) were used for model validation, a training-validation split of approximately 85%–15%. Six hundred twenty-four sections of noise (20 s length) were manually identified across all five stations (2011-08-31 to 2011-09-27), with 500 sections (2,500 waveforms) and 85 sections (425 waveforms) used for model training and validation, respectively. Two noise waveforms were randomly dropped from each dataset so that the training and validation noise data comprise 2,498 and 423 waveforms, respectively, to match the number of event waveforms.

A separate test dataset of 400 event waveforms with labeled P- and S-wave arrivals (132 events) and 400 noise waveforms (80 sections of noise) was also produced for subsequent model testing. These data come from a different time period than those used for training and validation data, with 200 waveforms from a period where data were recorded at 100 Hz sample rate (2011-10-04 and 2011-10-05) and 200 waveforms from a period with 50 Hz sample rate (2011-10-14, 2011-10-15 and 2011-11-27) for each category. All training, validation and test data were manually identified and quality checked.

The success of U-Net architectures relies on an effective data augmentation strategy when working with smaller datasets (Ronneberger et al., 2015). This allows the network to learn invariance to certain changes in input signal without them needing to appear in the annotated dataset. Here we outline a data augmentation strategy that improves performance of our U-GPD transfer learning model (Figure S2). First, as all stations were switched from 100 Hz sample frequency to 50 Hz sample frequency part way through the seismic deployment, we randomly select subsets of the training data (all originally sampled at 100 Hz) to be decimated to 50 Hz sample frequency throughout training. Each training sample (i.e., each 3-component waveform) has a probability of 0.5 of being selected for decimation before each training epoch, with an anti-aliasing, low-pass finite impulse response (FIR) filter applied and linear phase shift removed. Second, we randomly time-shift our P- and S-wave arrivals relative to the model input “window,” so that our waveforms differ slightly from epoch to epoch and the model must learn signal features that indicate arrivals rather than where they occur within the input window (i.e., arrivals don’t need to occur in the center of the window for the model to detect them). With our noise data, a random 400-sample window is chosen at each training epoch from our 20-s noise sections, introducing more waveform variety between training epochs.

All data processing and model training/testing were performed in Python using the ObsPy (Beyreuther et al., 2010; Krischer et al., 2015; Megies et al., 2011), TensorFlow (Abadi et al., 2015; <https://tensorflow.org>), and Keras (Chollet, 2015; <https://keras.io>) libraries.

5. Results

5.1. Training Metrics (Transfer Learning Versus no Transfer Learning)

To examine the impact of transfer learning and determine how much training data is required to produce an effective model, we use varying sized subsets of the training data throughout model training (i.e., 250, 500, 750, ..., 2,000, 2,250, and 2,498 training samples). Figure 3 compares how model loss (measure of distance between model predictions and ground truth labels) on training and validation data evolves throughout training between our transfer learning model and the same model with completely re-initialized weights (i.e., with no transfer learning) for our smallest and largest subsets of training data (250 and 2,498 training samples, respectively). The learning rate is set to be equal (1×10^{-3}) across the whole re-initialized model as we are no longer fine-tuning existing knowledge. All other hyperparameters, including dropout rate, are kept the same. The models trained without transfer learning (Figure 3b and 3d) show a much greater degree of overfitting: the model loss on the training data continues to decrease with more training while the loss on validation data (data that the model does not use during training) hits an inflection point and starts increasing, reflecting that the model is “memorizing” the precise features of the training data at the cost of generalization (Shorten & Khoshgoftaar, 2019). By contrast, the validation loss continues to decrease for the models trained with transfer learning (Figure 3a and 3c). Furthermore, the minimum validation loss achieved by the transfer learning models for each training dataset size is lower than when transfer learning is not employed (Figure 3 horizontal dashed lines). Such diagnostics indicate that transfer learning

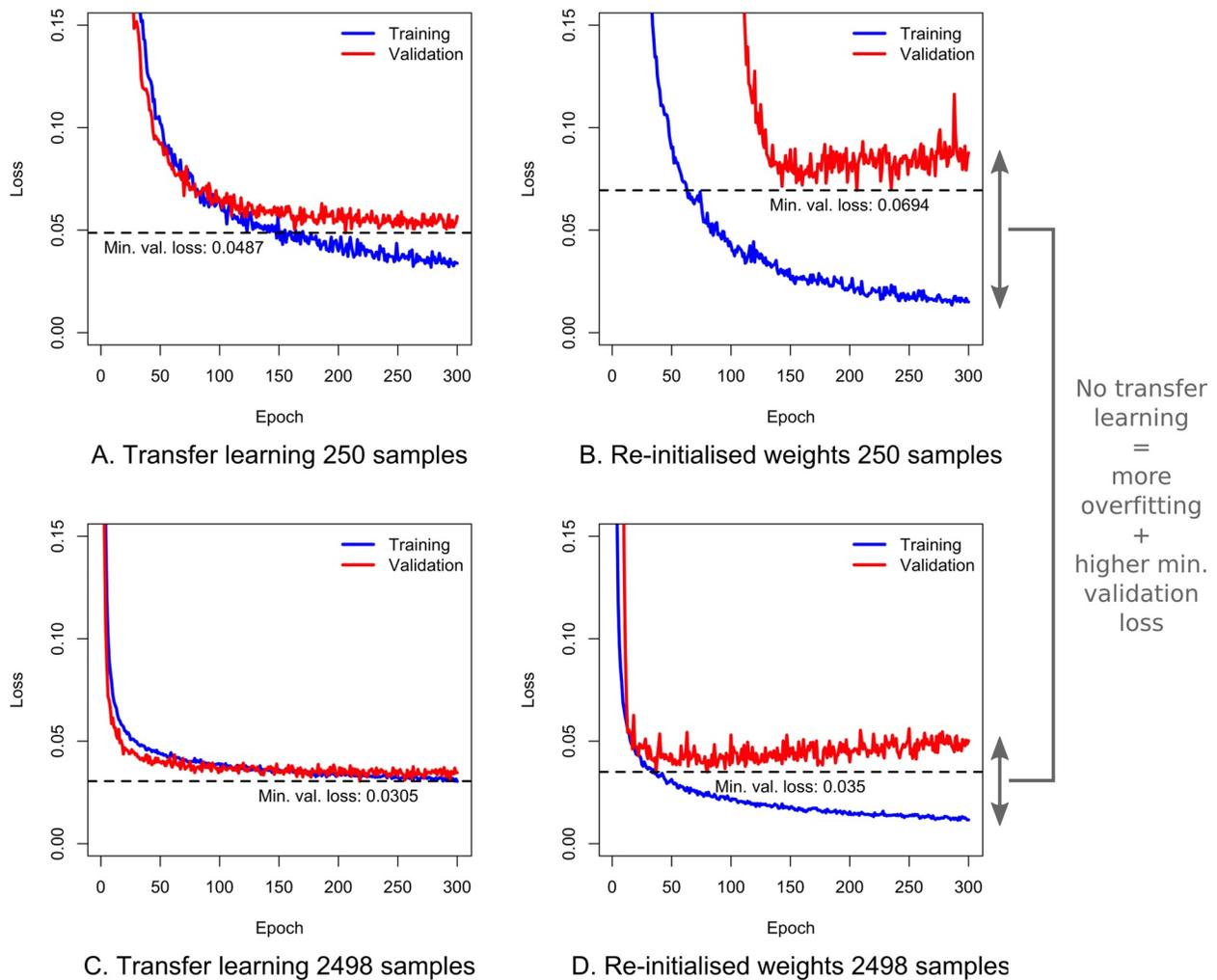


Figure 3. Model loss versus training epoch number. (a) Transfer learning model and 250 training samples of each class (P, S, or neither). (b) Model trained without transfer learning (i.e., initially randomized weights) and 250 training samples of each class. (c) Transfer learning model and full training dataset (2,498 training samples of each class). (d) Model trained without transfer learning (i.e., initially randomized weights) and full training dataset. Blue curve shows model loss for training data, red curve shows model loss for validation data (not seen during training). A lower model loss on training data (blue) than validation data (red) means the model shows signs of overfitting. The degree of overfitting (gap between blue and red curves) is much greater for the models without transfer learning (b and d) with validation loss hitting an inflection point then increasing while training loss continues to decrease. The transfer learning models also achieve a smaller minimum validation loss (horizontal dashed line) for each training set size.

is successfully preventing overfitting to the training data and will likely produce a model that generalizes better to nontraining data (Shorten & Khoshgoftaar, 2019). The greatly improved performance on validation data using the smallest subset of training data (Figure 3a and 3b) shows that transfer learning is particularly useful for reducing overfitting and model loss when training data are very limited, but this advantage is progressively diminished with increasing training dataset size (Figures 3 and 4).

Figure 4 shows the highest model accuracy (the proportion of labels the model classifies correctly) and lowest model loss achieved by our transfer learning and re-initialized models on validation data when trained using each subset size of training data. The transfer learning model achieves lower model loss regardless of training dataset size (Figure 4b). As training dataset size increases, the difference between the lowest loss achieved by the two models (gap between red circles and red triangles, Figure 4b) decreases and the advantages of transfer learning diminish. Generally, loss is considered a more robust metric than accuracy for model performance on future data as it measures the distance between model predictions and ground truth labels, whereas accuracy simply measures a binary true/false score. However, accuracy still provides useful information regarding model performance. In particular, the transfer learning model shows a stable

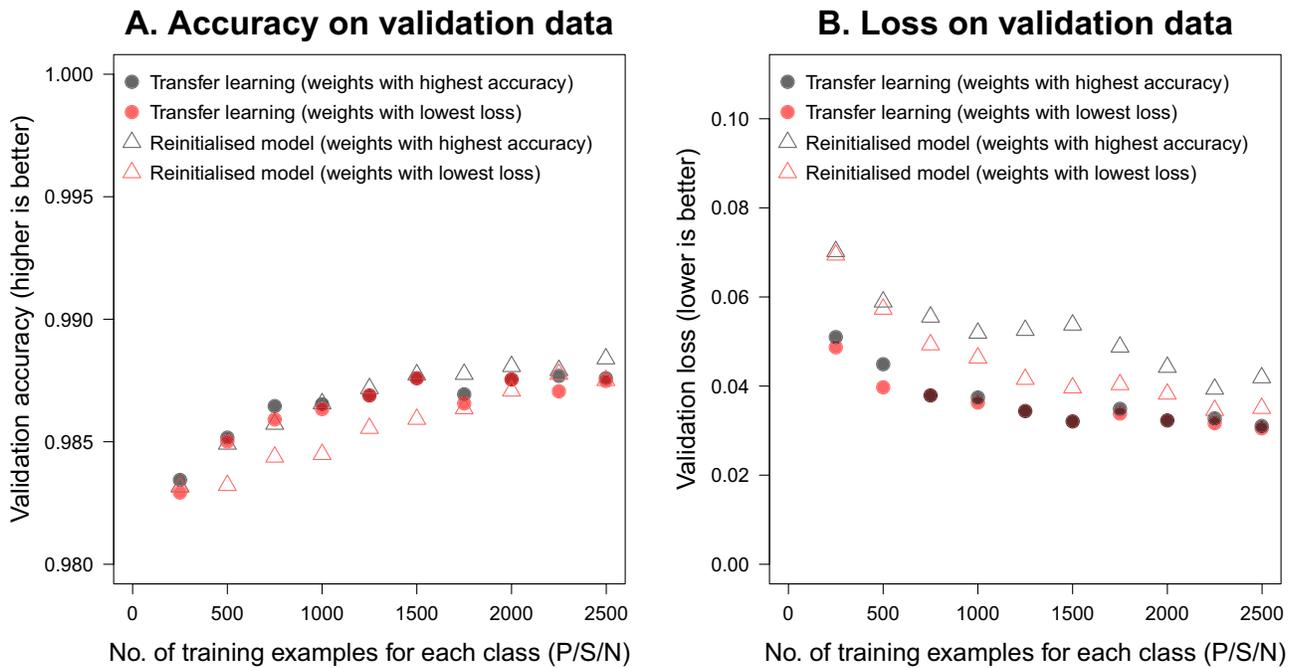


Figure 4. Model accuracy (a) and loss (b) for various subsets of training data. Open red circles are transfer learning model weights from epoch that achieves lowest validation loss (e.g., dashed horizontal lines in Figure 3), open black circles are transfer learning model weights from epoch that achieves highest validation accuracy, solid red triangles are re-initialized model (no transfer learning) weights from epoch that achieves lowest validation loss, and solid black triangles are re-initialized model weights from epoch that achieves highest validation accuracy.

relationship between maximizing model accuracy and minimizing model loss (gap between black and red circles is very small for all training subset sizes), where the training strategy of minimizing model loss appears to achieve the same goal as maximizing model accuracy, again a sign of reduced overfitting. The re-initialized model (black and red triangles), on the other hand, shows a much less stable relationship in this regard, with diverging training scores (Figure 4) indicating that high model accuracy comes at the cost of higher model loss and low model loss comes at the cost of lower model accuracy for these small training set sizes when transfer learning is not employed. The increased model loss for model weights with highest model accuracy (black triangles) also suggests that the model has become overconfident in its predictions (it has large errors on the small proportion of labels it gets wrong) and is therefore likely to perform worse on out-of-distribution data, with more false or missed phase arrival detections (e.g., a phase arrival being labeled as noise with high model confidence, or vice versa).

Model performance between the two approaches (transfer learning vs. re-initialization) converges as training set size increases, indicating that the need for transfer learning decreases with increased training set size, as expected. In fact, model performance with transfer learning appears to plateau, or possibly even degrade, at training subset sizes of more than 1,500 samples. This suggests that, with enough training data, transfer learning could potentially inhibit the model's ability to learn useful features in the new data that are absent in the original GPD training data. This apparent variance in performance may also simply be a result of the stochasticity arising from training using randomized weights in the new part of our transfer learning model.

5.2. Test Dataset (Known Arrival Times)

Following model training, we test the above models (i.e., new model with and without transfer learning) and two baseline models (GPD and PhaseNet) using the test dataset outlined in Section 4. We examine the proportion of correct class predictions (Figure 5) and the residuals between model and manually determined phase arrival pick times (Figure 6). Due to differences in model task types (classification vs. segmentation), we apply all models as sliding windows over 1000-sample waveforms (note that the PhaseNet model takes a

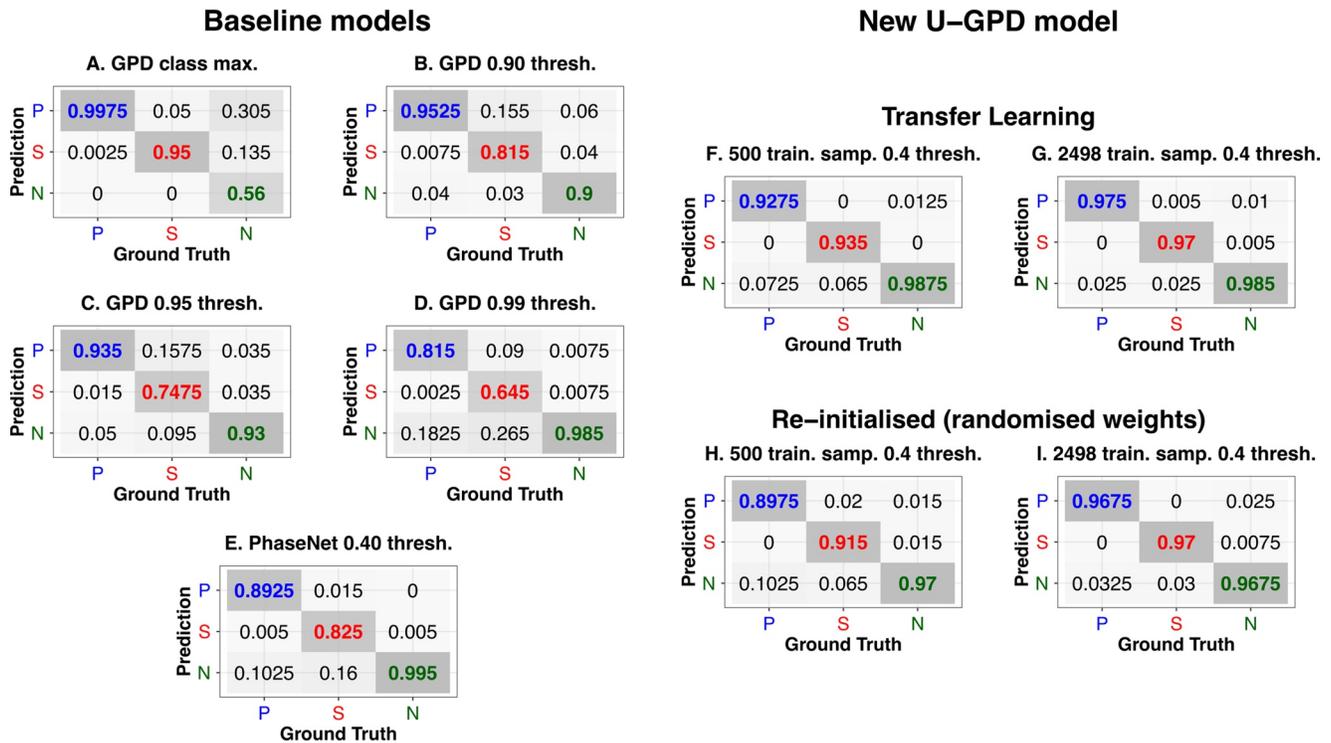


Figure 5. Confusion matrices for base Generalized seismic Phase Detection (GPD) model (a–d), PhaseNet model (e), U-GPD transfer learning model (f), 500 training samples, and (g), 2,498 training samples and re-initialized model (h), 500 training samples, and (i), 2,498 training samples. Values in matrices are proportion of ground truth phase arrivals (test set) assigned by each model to a given class (values of 1 along diagonal from top left to bottom right means all phase arrivals and sections of noise correctly identified).

3,000-sample waveform as input so we examine only the middle 1,000 samples for this model). To account for human picking error in collating our test set, we define a true positive for each phase arrival type (P or S) as the model prediction exceeding a given threshold value for that arrival type within 0.5 s of the manually determined arrival, such that predicted arrival times very close to the manually determined arrival time are considered accurate. A true positive for sections of noise is defined as no phase arrival prediction exceeding a given threshold value at any point within that section of data. The test data are pre-processed as per the training data for each model (i.e., GPD model tested on 2 Hz high-pass filtered data and all other models, including PhaseNet, tested on raw data; all detrended and self-normalized).

The GPD model is tested using four different threshold values (Figure 5a–5d) as this value strongly controls the number of false or missed phase arrival detections generated by this model. When the threshold is set to be whichever class label (P, S, or N) has the highest predicted value for a given waveform, nearly all P- and S-wave arrivals are detected by the GPD model (99.75% and 95% detection rate, respectively; Figure 5a). However, this threshold criterion makes the GPD model extremely prone to false phase arrival detections in sections of noise, with 44% of 1,000-sample noise waveforms in our test dataset containing at least one false phase arrival detection (Figure 5a, bottom right square) and many of our 1,000-sample event waveforms containing multiple phase arrival triggers (e.g., Figure 7b and 7e). When this threshold criterion is applied to continuous sections of data from Nabro, the number of false phase arrival detections overwhelmingly outweighs the number of true phase arrival detections and becomes unmanageable in terms of correctly associating phases, identifying true events and processing the data within computational memory constraints.

One way to lower the number of false phase arrival detections is to use a higher threshold value for P- and S-wave predictions. Figure 5b shows the GPD model's performance on our test data using a 0.9 threshold value (i.e., a P or S prediction “probability” must exceed 0.9 to be included). The number of false detections in sections of noise is greatly reduced (down from 44% of waveforms to 10% of waveforms) but at the cost of reduced true phase arrival detections (~95% and ~82% of P- and S-wave arrivals, respectively). Part of

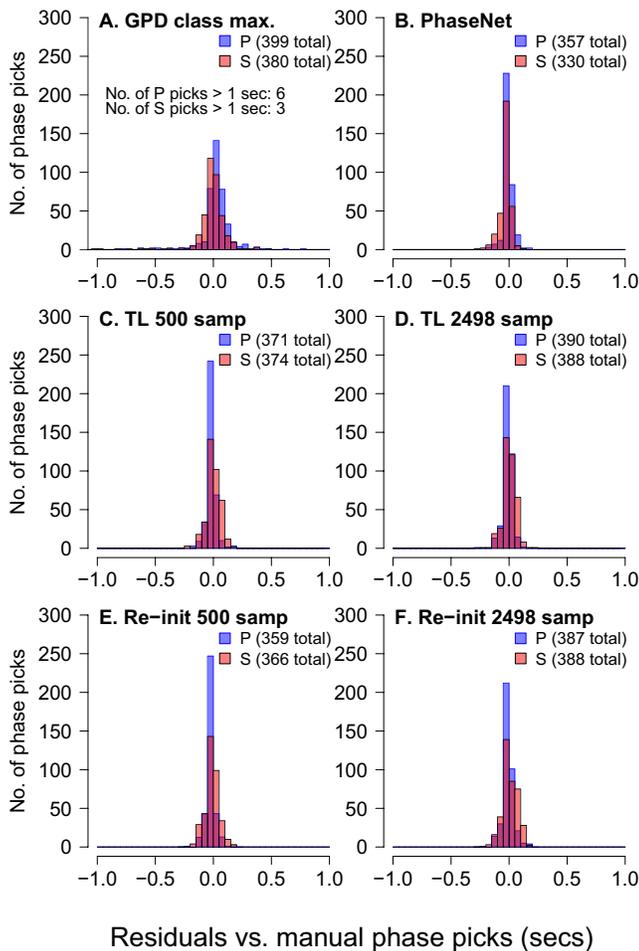


Figure 6. Model phase pick residuals versus manual phase picks for base Generalized seismic Phase Detection (GPD) model (a), PhaseNet model (b), U-GPD transfer learning model (c), 500 training samples, and (d), 2,498 training samples, and reinitialized model (e), 500 training samples, and (f), 2,498 training samples. The models based on semantic segmentation (b–f) yield smaller phase pick residuals.

this performance dip is undoubtedly due to the difference in sample rates between one half of the test data (50 Hz) and the GPD model's training data (all 100 Hz). When the threshold value is increased further (i.e., P or S prediction must exceed 0.95 or 0.99; Figure 5c and 5d), the GPD model yields even fewer false phase arrival detections in noise sections but at the cost of fewer P- and S-wave arrivals.

Figure 5e shows the performance of the PhaseNet model on our test dataset. This model is included as it adopts the same U-Net segmentation approach as our new model and is trained on data from a variety of instrument types, although the training data is still exclusively from California. The PhaseNet model is much less prone to false phase arrival detections than the GPD model (Figure 5e, bottom right square); as such, a much lower threshold value (0.4) can be used to maximize the number of true phase arrival detections. This model accurately identifies ~89% and ~83% of P- and S-wave arrivals in our test dataset, which is better than the GPD model with a threshold value that achieves a similar false detection rate (e.g., Figure 5d), but detects fewer phase arrivals than our transfer learning and reinitialized models trained with Nabro data (Figure 5f–5i).

When trained using a subset of just 500 training samples for each class (P/S/N) and evaluated using a prediction threshold value of 0.4, the transfer learning approach correctly detects ~93% and ~94% of P- and S-wave arrivals with very few false phase arrival detections in sections of noise (~1%; Figure 5f), a clear improvement over our model trained with re-initialized weights and the same training subset (Figure 5h). When our full training dataset is used (2,498 samples for each class), model performance converges between transfer learning (Figure 5g) and re-initialization (Figure 5i), with a similar number of correctly identified phase arrivals and false detections in noise, although the transfer learning model still performs marginally better, particularly on sections of noise. In essence, the transfer learning model strikes a better balance between high phase arrival detection rate (~97%–98% for each phase arrival type; Figure 5g, top left and center squares) and low false detection rates in sections of noise (~1%; Figure 5g, bottom right square) on our test data from Nabro volcano than any of the existing baseline models (Figure 5a–5e) or training a model from scratch (Figure 5i).

Figure 6 shows the residuals for each model between their predicted phase arrival times and the original manual pick times for these test waveforms. Predicted phase arrival times were determined using a simple trigger algorithm (e.g., Withers et al., 1998) on each model's probability time series with the time series index that yields maximum predicted value chosen as the pick time for a given phase arrival type (Figure 7). The models that employ semantic segmentation (i.e., PhaseNet, our U-GPD transfer learning model and our re-initialized model; Figure 6b–6f) show comparable pick time precision (root mean square deviation [RMSD] of 0.036, 0.038, and 0.044 s, respectively, for each model's P-wave predictions and RMSD of 0.053, 0.053 and 0.065 s, respectively, for each model's S-wave predictions). The GPD model (Figure 6a), by comparison, has a more diffuse range of phase arrival pick times (RMSD of 0.217 s for P-waves and 0.188 s for S-waves), with some model picks made more than 1 s before or after the manually determined arrival time. This is almost certainly a result of its more ambiguous class labeling (Figure 1) and the broad phase arrival probability peaks it generates (Figure 7).

Figure 7 shows three example waveforms from the test set with corresponding model predictions for the U-GPD transfer learning, GPD and PhaseNet models. These waveforms were chosen as they have low SNR phase arrivals. Prediction labels for the U-GPD model resemble the boxcar labels of the training set (Figure 7a, 7d and 7g), whereas prediction labels produced by the PhaseNet model resemble the model's

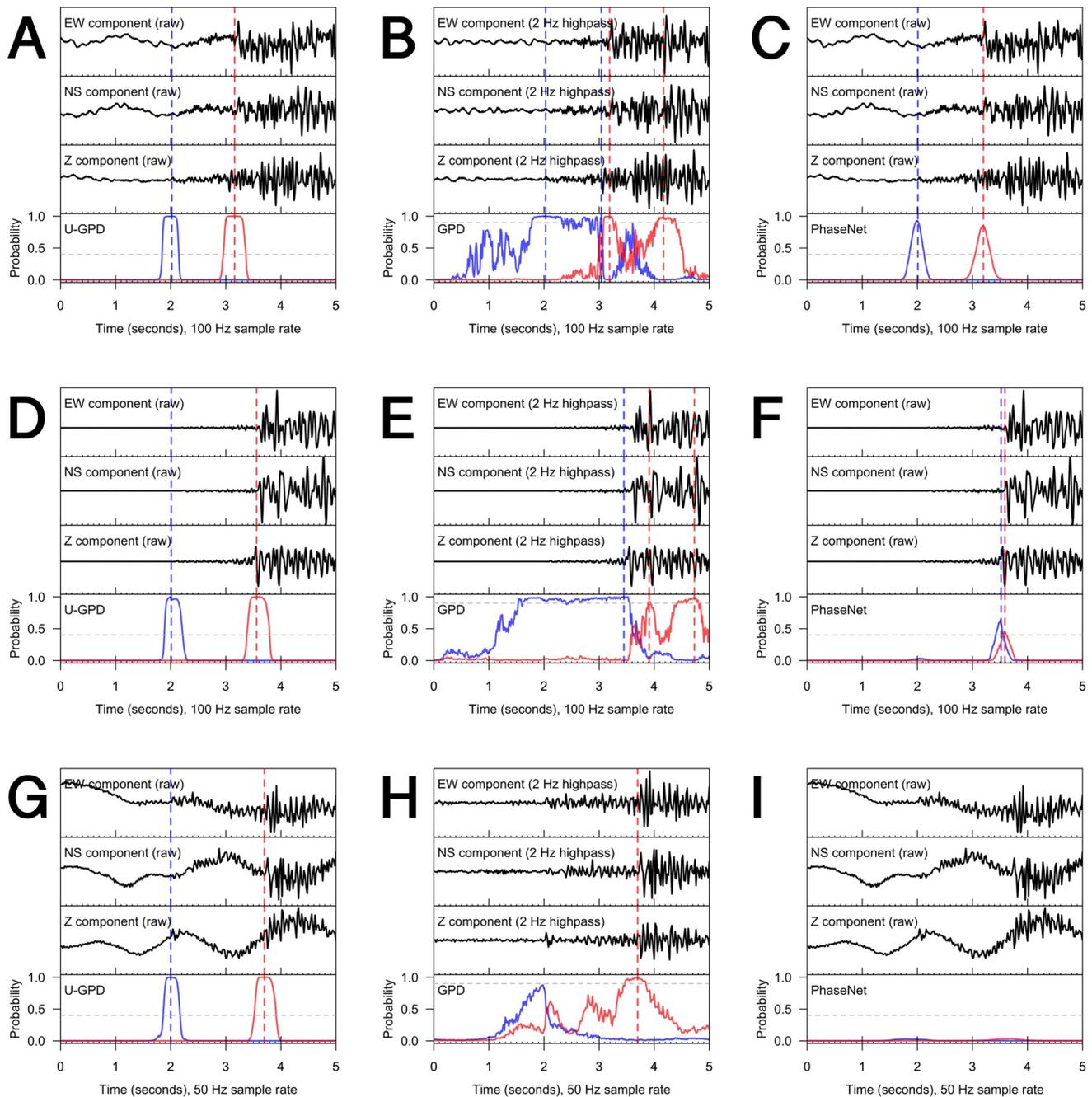


Figure 7. Three example waveforms from our test set. Phase arrival prediction trigger thresholds (horizontal dashed lines) are 0.4, 0.9 and 0.4 for U-GPD (left), Generalized seismic Phase Detection (GPD) (center) and PhaseNet (right), respectively. (a–c), Test waveform with substantial high frequency background noise. All models accurately detect P- and S-wave arrivals but GPD model makes multiple phase detections. (d–f), Test waveform with low amplitude P-wave arrival. Existing GPD and PhaseNet models incorrectly label S-wave arrival as close combination of P-wave arrival and S-wave arrival. U-GPD transfer learning model correctly detects both P- and S-wave arrivals. (g–i), Test waveform with substantial low frequency background noise. P-wave arrival prediction is below trigger thresholds for both GPD and PhaseNet models, although GPD model accurately detects S-wave arrival. U-GPD transfer learning model correctly identifies both P- and S-wave arrivals.

truncated Gaussian-style training labels (Figure 7c, 7f and 7i). Despite these boxcar shapes, the U-GPD model's maximum predicted value for each phase arrival consistently and accurately picks both P- and S-wave arrivals (Figure 7a, 7d and 7g). On the other hand, the base GPD model's prediction labels are considerably broader and noisier (Figure 7b, 7e and 7h). The U-GPD model appears to have benefitted from

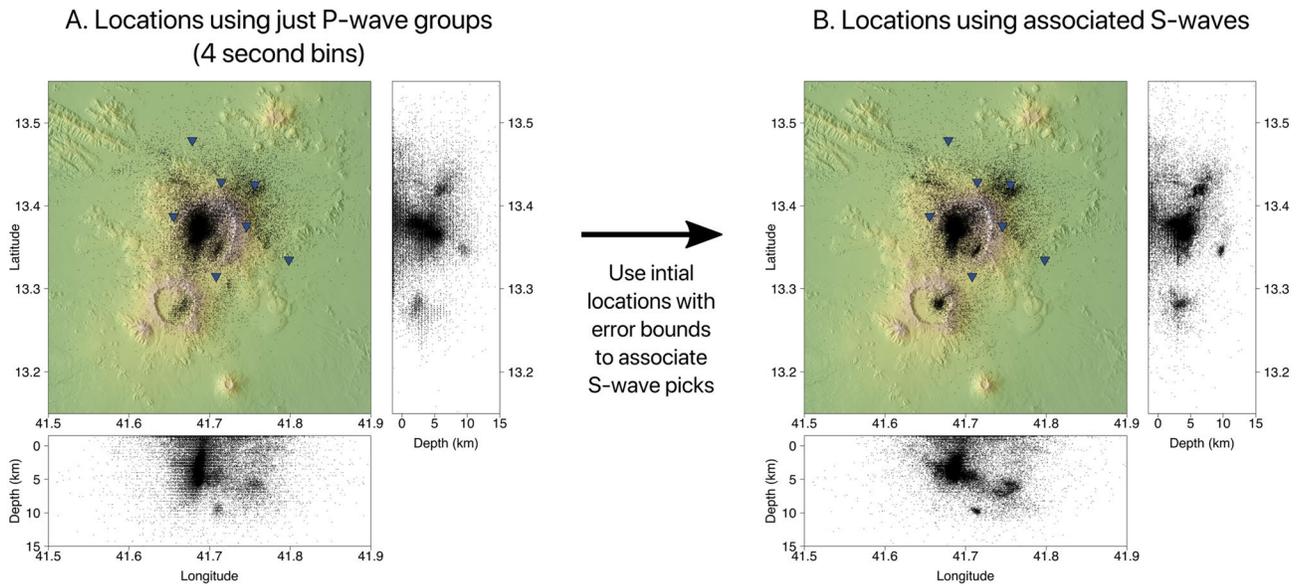


Figure 8. U-GPD transfer learning model event locations (total no. of events = 33,950) using automated phase association strategy. (a) P-wave phase arrival triggers are grouped into 4 s bins and these groupings are used to obtain initial event hypocenters and origin times. (b) S-wave phase arrival triggers are associated to P-waves in (a) using initial origin times, a V_p/V_s ratio of 1.76 and a travel-time error of 25%. Events are then located again using all included P-wave and S-wave arrivals.

retraining using Nabro-specific data, as it performs much better than the existing models on these challenging waveforms.

5.3. Full 14-Month Deployment (Unknown Arrival Times)

While evaluating model performance on individual, manually scrutinized waveforms is useful for benchmarking and yielding estimates of model efficacy, the model's performance in a "real-world" setting is ultimately of most importance to seismic analysts. Evaluating such performance is inherently more challenging, however, as the number of events in long sections of monitoring data and their respective phase arrival times are unknown, and other considerations, such as computational time and resources (e.g., memory requirements and availability of optimized hardware), affect model feasibility as a monitoring tool.

In this section, we present results of our best performing model in the prior section (U-GPD transfer learning model trained with full training dataset of 2,498 samples of each class) and the original base GPD model when run over the full 14-month Nabro seismic deployment (Figure 8). As with the test dataset in Section 5.2, phase arrivals are detected at individual stations through a simple trigger algorithm, where an arrival is detected if the probability assigned to that class label (P or S) exceeds a given threshold (e.g., 0.4 for our U-GPD transfer learning model). The phase arrival time is determined as the waveform sample with the highest probability for that phase (Figure 7).

The U-GPD transfer learning model was applied to the data as a sliding window with 50% overlap (i.e., applied at "time shifts" of 200 samples) over 24-h sections of data from each individual station. The model takes 5 s to process 24 h of 3-component data at 100 Hz sample rate (or 3 s per day at 50 Hz sample rate) on a single graphics processing unit (GPU; NVIDIA GeForce RTX 2080 Ti), a rate many orders of magnitude faster than "real-time" even when run on hundreds of stations. To avoid poor predictions due to window edge effects, only the middle 200 sample predictions out of 400 from each window are used to predict phase arrivals and are concatenated to produce one long continuous prediction trace without overlap or gaps and with the same sample rate as that of the input signal (i.e., 100 or 50 Hz). With all other processing steps (e.g., software initialization, data read/write, signal windowing, running trigger algorithm, etc.), the U-GPD transfer learning model picks phase arrivals at all seven available stations from the full 14-months

deployment in less than 4 h using a single GPU (greatly reduced when parallelized over multiple GPUs), indicating that it could easily be used within real-time monitoring constraints.

Conversely, as the GPD model produces only one class prediction per window (Figure 1a), we apply this model with much greater overlap (97.5%; every 10 samples of data) and with varying threshold values (0.9, 0.95 and 0.99) for phase arrival detection triggering. This generates a prediction trace with a much coarser sample rate than the original input signal (i.e., from 100 or 50 Hz to 10 or 5 Hz, respectively) and takes 26 s per 24 h of 3-component data at 100 Hz sample rate (or 15 s per day at 50 Hz sample rate) on the same NVIDIA GPU, approximately a five-fold increase in computational time with a tenth of the temporal detail. With all other processing steps, the GPD model took almost 50 h to run over the full 14-month deployment using a single GPU, more than a ten-fold increase in computational time over the transfer learning model, due to more (pre-)processing required (e.g., more signal windows generated and subsequent processing). Assuming a linear increase in computational time, running the model as a sliding window over every sample of data would take ~ 260 s per 24 h of 3-component data at 100 Hz sample rate and ~ 500 h (nearly three weeks) for the full 14-months deployment and seven stations. While this is still faster than real-time, these timescales for a single or limited number of station(s) could become limiting when applied at hundreds of stations, particularly without high performance computing resources.

5.3.1. Phase Association Method

Both models detect P- and S-wave phase arrivals but do not associate them to the same event. To assess the number of locatable events detected, we group P-wave phase arrival triggers into 4-s bins and keep only bins with arrivals detected at four or more stations. This bin size was chosen to encompass the maximum plausible travel time between any two stations. If multiple arrivals were detected at the same station within a 4-s bin, the detection threshold was increased for all arrivals in that particular bin to retain only the highest probability phase picks. If any of these bins now had arrivals at less than four stations, as a result of removing lower probability phase picks, they were discarded as there would be too few stations to constrain event location. If there were still multiple arrivals present at any given station, only the arrivals with highest probability for each station were kept. Finally, if phase arrival bins intersected (a subset of one bin was contained in another), the bin with highest mean probability was kept. This association method is clearly quite crude, and only works for small, very local arrays, but allows a broad evaluation of model performance at detecting phase arrivals. Use of a more rigorous phase association method (e.g., Ross et al., 2019; Yeck et al., 2019) would obviously be better at eliminating false arrival picks or identifying multiple events within a 4 s window, which is a common feature of seismicity during volcanic unrest. However, this will mask underlying model performance; for example, the inclusion of false arrival picks is likely to generate greater estimated location errors (Figure 10).

We associate S-wave arrivals to their corresponding P-wave arrivals by first locating events using NonLinLoc (e.g., Lomax et al., 2000), a widely used software package for probabilistic earthquake location, using the P-wave arrival bins outlined above (Figure 8a) and a simple 1D linear gradient velocity model from previous seismic studies at Nabro (Table S8; Goitom et al., 2015; Hamlyn et al., 2014). The difference between P-wave arrival and event origin times were used to predict which S-wave arrival detections should be associated with each P-wave arrival using a V_p/V_s ratio of 1.76 (Goitom et al., 2015) and S-wave travel time error of 0.25 (25%). S-wave arrival triggers that lay within this error bound for each detected P-wave arrival were associated to that event. S-wave arrivals at stations without a detected P-wave arrival were not included. All events were then located again in NonLinLoc using all included phase arrivals (Figure 8b).

5.3.2. Detected Events and Location Errors

Figure 9 shows the cumulative number of events detected by the U-GPD transfer learning model (threshold value of 0.4; black solid line) and the original GPD model (threshold values of 0.9, 0.95 and 0.99; gray lines). The cumulative number of events from an existing manual catalog for this deployment (Goitom, 2017; Hamlyn et al., 2014), some of which provided the transfer learning model training data, is also given for reference. Event locations for each model and the manual catalog are provided in Supplementary Materials (Figures S3–S6). When only P-wave arrivals are used (Figure 9a), the GPD model with detection threshold of 0.9 appears to detect the most events (total no. of events detected by GPD model = 41,007; total no. of events detected by transfer learning model = 33,950). A threshold of 0.95 also detects more events than

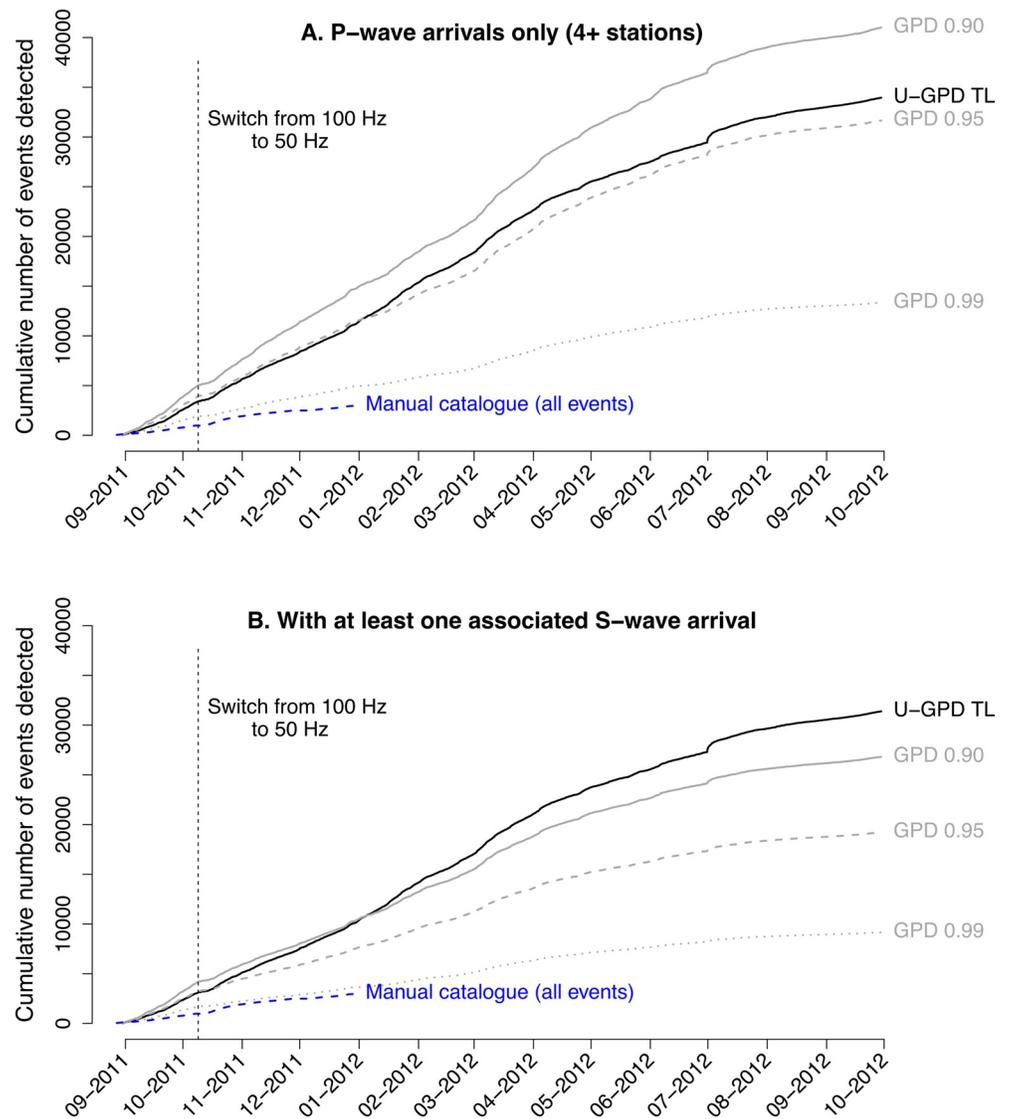


Figure 9. Cumulative number of events detected by Generalized seismic Phase Detection model (various thresholds, gray lines) and transfer learning model trained on full Nabro dataset (2,498 samples of each class, 0.4 threshold, black line). Blue dashed line is existing manual catalog (Goitom, 2017). All training/validation waveforms are from dates before switch in sample frequency (vertical dashed line). (a) Cumulative number of events detected using P-wave arrivals only (see main text for event binning procedure). (b) Cumulative number of events with at least one associated S-wave arrival.

the transfer learning model until shortly after the switch in instrument sample rates from 100 to 50 Hz. However, when we consider events with at least one associated S-wave arrival, the transfer learning model detects more events overall (Figure 9b; no. of events detected by transfer learning model = 31,387; no. of events detected by GPD model with 0.9 threshold = 26,808). This is consistent with the results from our test dataset in Section 5.2, with the proportion of S-wave arrivals accurately detected by the GPD model at these threshold values much lower than the proportion of P-wave arrivals detected (Figure 5b–5d). Furthermore, 6% of noise waveforms and 16% of S-wave arrivals from our test data were mislabeled by the GPD model (0.9 threshold value) as P-wave arrivals (Figure 5b), a higher rate of false detections or labels than the transfer learning model (1% of noise sections and 0.5% of S-waves, respectively; Figure 5g). This means that a higher proportion of the P-wave groupings detected by this model with 0.9 threshold value are likely to include mislabeled S-waves or false arrivals, which is reflected in subsequent event location errors (Figure 10c–10d).

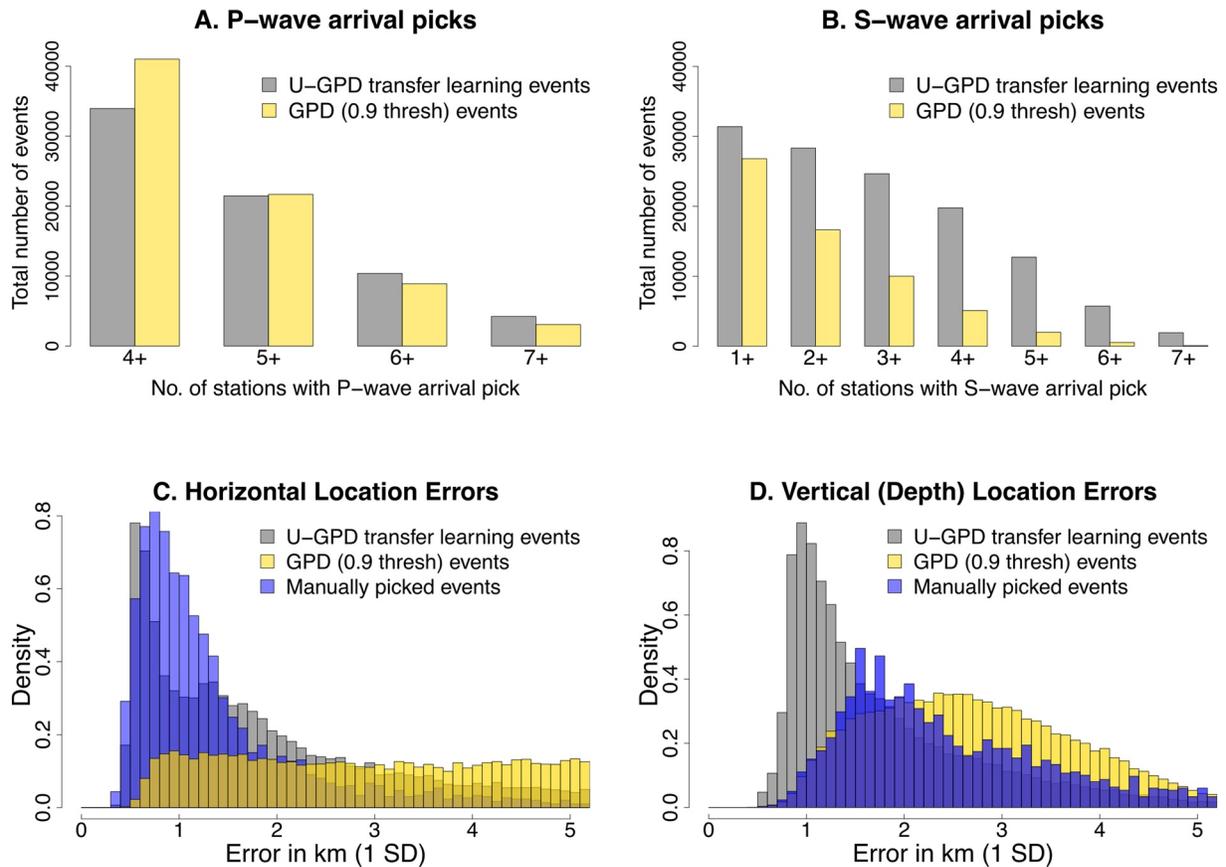


Figure 10. (a) Number of P-wave arrival picks per event for transfer learning model (gray) and base Generalized seismic Phase Detection (GPD) model (gold). (b) Number of S-wave arrival picks per event. (c) Histogram of Gaussian horizontal location errors (one standard deviation) for events picked by transfer learning model (gray) and base GPD model (gold), and those in the existing manual catalog (blue). (d) Histogram of Gaussian vertical (depth) location errors (one standard deviation).

To scrutinize these results further, we examine the number of stations with P- and S-wave arrival detections per event (Figure 10a–10b). In general, the events detected and picked by the U-GPD transfer learning model include more stations and considerably more S-wave arrivals than those picked by the GPD model, although the number detected by the GPD model may have been reduced by using a coarser prediction trace (every 10 samples, a requirement to reduce model run time to a reasonable timeframe). This increase in the number of stations and S-wave arrivals per event will constrain event locations, as seen in the location errors derived from the models' phase arrival picks (Figure 10c–10d).

Location errors are estimated by NonLinLoc using multi-dimensional Gaussian estimators and subsequent confidence intervals (e.g., Lomax et al., 2000). The horizontal errors (Figure 10c) for the locations produced using the transfer learning model pick times are comparable to the existing manually picked events. Furthermore, vertical (depth) errors are much improved over the manual catalog (Figure 10d), likely reflecting more consistency in S-wave arrival picking than that of a manual analyst. The GPD model, by comparison, produces a more diffuse range of horizontal and vertical errors, which is likely to be a combination of coarser prediction trace, poorer pick precision (Figure 6a), lack of S-wave arrivals (Figure 10b) and false/mis-labeled P-wave arrival detections (Figure 5b). This interpretation is further supported when we look at the number of event locations lying within the array (i.e., event locations lying within the convex hull of station coordinates) for each model: NonLinLoc locates more events within the array using the transfer learning picks ($n = 23,859$) than using the GPD model with 0.9 threshold value ($n = 22,826$). While we expect many events to occur outside of the array (e.g., at neighboring faults or volcanic centers), this metric shows that a much larger proportion of event locations detected by the GPD model lie away from the volcanic edifice,

which may reflect poorer pick precision, false/mislabeled arrivals or coarser prediction trace, but may also reflect the event types (i.e., regional tectonic) that the original model was trained on.

6. Discussion

Transfer learning using existing seismological deep learning models can be a highly effective strategy to automate phase arrival picking in settings with little or no prior monitoring. We demonstrate that, with a limited number of hand-labeled waveforms (on the order of hundreds to low thousands) and a few minutes of training time, one can produce a consistent and effective deep learning model for phase arrival detection that requires no other manual intervention or tuning and can process years of data in a matter of hours.

For small training datasets, the use of pre-existing, generalized CNN filters greatly reduces model overfitting (i.e., model parameters “memorizing” the training data) when compared with training a model from scratch (Figure 3) and yields a more stable relationship between maximizing model accuracy and minimizing model error (Figure 4). Furthermore, when combined with a good data augmentation strategy, transfer learning can also address the issue of processing data when instrument sample rates differ from those used to train existing models. When applied to data from Nabro volcano, augmenting our training set with decimated waveforms greatly improves model performance on lower sample rate data (Figure S2). As such, hand-labeled training data from the first 35 days of the deployment (all 100 Hz sample rate) were sufficient to detect phase arrivals throughout the duration of the deployment, even after instrument sample rates were switched to 50 Hz (Figure 9). Without this data augmentation step, model performance on lower sample rate data declines dramatically (Figure S2). This shows that where sample rates are altered or new instruments added during a seismic deployment, data augmentation can overcome the cost of collecting further hand-labeled data and allow models to be adapted cheaply and quickly throughout the deployment.

The introduction of new, task-specific data and the change in model task from one of classification to one of segmentation also improves our U-GPD model pick time precision (Figure 6), the number of stations per detected event (Figure 10a), the number of S-wave arrivals detected (Figures 5 and 10b) and computational efficiency over the original base GPD model, as well as potentially reducing the number of false/mislabeled P-wave detections (Figure 5) and increasing the number of identified events that relate directly to volcanic activity (evidenced by the increased number of events located within the array). Without manual intervention or sophisticated phase association, phase arrival picks from the U-GPD transfer learning model produce locations with smaller depth errors than the base GPD model and even manually determined phase arrival times (Figure 10d). This is likely a result of more consistent picking and labeling, particularly for S-wave arrivals, which is difficult even for manual analysts to perform consistently, and suggests that very few of the events detected are false.

Given the greatly improved computational time over the base GPD model, the small number of training events required and the use of a high-level, user-focused programming library (Keras), this approach is well within the reach of volcano observatories and research groups. Previous studies that analyze the pre-, syn- and post-eruptive periods at Nabro volcano have relied on manually-produced seismic catalogs comprising hundreds of events (e.g., Goitom et al., 2015; Hamlyn et al., 2014; the latter locating 658 events over 38 days, a rate of <18 events per day). Our U-GPD transfer learning model yields a seismic catalog that is, order of magnitudes larger (33,950 events over 396 days, a rate of >85 events per day; Figures 8 and 9), with smaller location errors (Figure 10), in a matter of hours. Furthermore, as the model processes 1D waveform data, as opposed to 2D spectrogram images in some other existing models (e.g., Dokht et al., 2019; Lara et al., 2020; Titos et al., 2020), it runs quickly on high resolution data without using a GPU optimized for deep learning frameworks (32 s per 24 h of 100 Hz data on an Intel Core i7 desktop CPU) and so could easily be deployed for real-time monitoring with limited computing resources or at much larger arrays. The methods and computational times in this study have relied on standard, generic libraries (ObsPy, TensorFlow and Keras); the use of more optimized, compiled code or higher-performance/lower-level languages (e.g., Julia and C) could greatly improve computational times further.

Beyond phase arrival picking, the generalized waveform features extracted by existing, extensively trained models, such as the GPD model (Figure 1a), could serve as a useful feature extraction system for models designed for other waveform processing tasks. For example, information regarding frequency content and

orientation of seismic energy extracted by the GPD model (Figure 1a inset) could reasonably provide useful features for a new model designed to automatically classify volcano seismic event types (e.g., Bueno et al., 2020; Hibert et al., 2017; Lara et al., 2020), particularly when available annotated datasets are small or unbalanced. However, with larger datasets, there is the potential for transfer learning to inhibit learning of new, useful features, particularly if the source and target tasks or data distributions differ considerably.

The number of seismological studies to date that employ transfer learning is relatively low (e.g., Bueno et al., 2020; Chai et al., 2020; El Zini et al., 2020; Huot et al., 2018; Titos et al., 2020). This is undoubtedly, in part, due to the lack of extensively trained, well-documented, publicly available seismological models. However, the number is likely to grow as more extensive datasets and models are developed and released into the public domain. We credit the availability of the GPD model in the public domain and use of a popular, user-focused machine learning framework (Keras) as the foundation of the work presented in this study. Such availability facilitates adaptation and experimentation; development of other publicly available models and extensive datasets would aid progress in the field of seismological machine learning.

While the application of transfer learning can overcome the perception that deep learning models require a “large upfront cost” in terms of data and computational resources, the development and benchmarking of large-scale, extensive models and datasets are still imperative to push the field of seismological machine learning forwards and extend applications to all aspects of seismic processing and inference. However, it is hoped that applications such as the one presented in this study will motivate the initial investment in the development of such models so that the cost of producing effective task-specific models (e.g., through transfer learning) is progressively reduced.

Data Availability Statement

All seismic data from the Nabro Urgency Array (Hammond et al., 2011; https://doi.org/10.7914/SN/4H_2011) are publicly available through IRIS Data Services (<http://service.iris.edu/fdsnws/dataselect/1/>). See Hammond et al. (2011) for further details on waveform data access and availability. Model training, validation and test sets/metadata are archived and available through Zenodo (Lapins et al., 2021; <https://doi.org/10.5281/zenodo.4498549>). Full code to reproduce our U-GPD transfer learning model, perform model training, run the U-GPD model over continuous sections of data and use model picks to locate events in NonLinLoc (Lomax et al., 2000) are available at <https://github.com/sachalapins/U-GPD>, with the release (v1.0.0) associated with this study also archived and available through Zenodo (Lapins, 2021; <https://doi.org/10.5281/zenodo.4558121>).

References

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., & Zheng, X. (2015). *TensorFlow: Large-scale machine learning on heterogeneous distributed systems*. Retrieved from <https://www.tensorflow.org>

Barbedo, J. G. A. (2018). Impact of dataset size and variety on the effectiveness of deep learning and transfer learning for plant disease classification. *Computers and Electronics in Agriculture*, 153, 46–53. <https://doi.org/10.1016/j.compag.2018.08.013>

Bergstra, J., & Bengio, Y. (2012). Random search for hyper-parameter optimization. *Journal of Machine Learning Research*, 13(10), 281–305.

Beyreuther, M., Barsch, R., Krischer, L., Megies, T., Behr, Y., & Wassermann, J. (2010). ObsPy: A python toolbox for seismology. *Seismological Research Letters*, 81(3), 530–533. <https://doi.org/10.1785/gssrl.81.3.530>

Bojanowski, A. (2011). Volcano mix-up. *Nature Geoscience*, 4(8), 495. <https://doi.org/10.1038/ngeo1222>

Bueno, A., Benitez, C., De Angelis, S., Diaz Moreno, A., & Ibanez, J. M. (2020). Volcano-seismic transfer learning and uncertainty quantification with Bayesian neural networks. *IEEE Transactions on Geoscience and Remote Sensing*, 58(2), 892–902. <https://doi.org/10.1109/TGRS.2019.2941494>

Chai, C., Maceira, M., Santos-Villalobos, H. J., Venkatakrishnan, S. V., Schoenball, M., Zhu, W., et al. (2020). Using a deep neural network and transfer learning to bridge scales for seismic phase picking. *Geophysical Research Letters*, 47(16). e2020GL088651. <https://doi.org/10.1029/2020GL088651>

Chollet, F. (2015). *Keras*. Retrieved from <https://keras.io>

Daumé, H. (2007). Frustratingly easy domain adaptation. In *45th annual meeting of the association for computational linguistics (ACL)* (pp. 256–263). Retrieved from <https://arxiv.org/abs/0907.1815>

Dokht, R. M. H., Kao, H., Visser, R., & Smith, B. (2019). Seismic event and phase detection using time–frequency representation and convolutional neural networks. *Seismological Research Letters*, 90(2A), 481–490. <https://doi.org/10.1785/0220180308>

Donovan, A., Blundy, J., Oppenheimer, C., & Buisman, I. (2018). The 2011 eruption of Nabro volcano, Eritrea: Perspectives on magmatic processes from melt inclusions. *Contributions to Mineralogy and Petrology*, 173(1), 1–23. <https://doi.org/10.1007/s00410-017-1425-2>

Dramsich, J. S., & Lüthje, M. (2018). Deep-learning seismic facies on state-of-the-art CNN architectures. In *SEG technical program expanded abstracts* (pp. 2036–2040). Anaheim, CA, USA.

Acknowledgments

The seismic data were collected with funding from the Natural Environment Research Council (NERC) project NE/J012297/1 (“Mechanisms and implications of the 2011 eruption of Nabro volcano, Eritrea”). The UK seismic instruments and data management facilities were provided under loan number 976 by SEIS-UK at the University of Leicester. The facilities of SEIS-UK are supported by NERC under Agreement R8/H10/64. Author SL was supported by a GW4+ Doctoral Training Partnership studentship from the Natural Environment Research Council (NERC) [NE/L002434/1]. Author BG was funded by the Engineering and Physical Sciences Research Council (EPSRC) and the School of Earth Sciences at the University of Bristol. Author MJW was funded by UKRI GCRF EP/P028233/1 (“PREPARE”) and NERC NE/R017956/1 (“EQUIPT4RISK”). Author JMK was funded by NERC grant NE/R018006/1. Author KVC was supported by the AXA Research Fund. The authors gratefully acknowledge support from the sponsors of the Bristol University Microseismicity ProjectS (BUMPS) and the NERC Centre for the Observation and Modelling of Earthquakes, volcanoes and Tectonics (COMET). The authors also gratefully acknowledge the cooperation we received from the Eritrea Institute of Technology, Eritrean government, Southern and Northern Red Sea Administrations, local sub-zones and village administrations. The authors thank the Department of Mines, Ministry of Energy and Mines for their continued support throughout the project. Special thanks go to Zerai Berhe, Mebrahtu Fisseha, Michael Eyob, Ahmed Mohammed, Kibrom Nerayo, Asresehey Ogbatsien, Andemichael Solomon and Isaac Tuum. The authors thank Alem Kibreab and Prof. Ghebrebrhan Ogubazghi for their vital help in facilitating the field-work. IRIS Data Services are funded through the Seismological Facilities for the Advancement of Geoscience (SAGE) Award of the National Science Foundation under Cooperative Support Agreement EAR-1851048.

- D'souza, R. N., Huang, P.-Y., & Yeh, F.-C. (2020). Structural analysis and optimization of convolutional neural networks with a small sample size. *Scientific Reports*, *10*(834). <https://doi.org/10.1038/s41598-020-57866-2>
- Efremova, D. B., Sankupellay, M., & Kononov, D. A. (2019). Data-efficient classification of birdcall through convolutional neural networks transfer learning. In *2019 Digital image computing: Techniques and applications (DICTA)* (pp. 1–8). Perth, Australia. <https://doi.org/10.1109/DICTA47822.2019.8946016>
- El Zini, J., Rizk, Y., & Awad, M. (2020). A deep transfer learning framework for seismic data analysis: A case study on bright spot detection. *IEEE Transactions on Geoscience and Remote Sensing*, *58*(5), 3202–3212. <https://doi.org/10.1109/TGRS.2019.2950888>
- Feurer, M., Klein, A., Eggenberger, K., Springenberg, J., Blum, M., & Hutter, F. (2015). Efficient and robust automated machine learning. In *Advances in Neural Information Processing Systems NIPS 2015* (Vol. 28, pp. 2962–2970).
- Fromm, M., Kablick, G., Nedoluha, G., Carboni, E., Grainger, R., Campbell, J., & Lewis, J. (2014). Correcting the record of volcanic stratospheric aerosol impact: Nabro and Sarychev peak. *Journal of Geophysical Research: Atmospheres*, *119*, 10343–10364. <https://doi.org/10.1002/2014JD021507>
- Ganin, Y., Ustinova, E., Ajakan, H., Germain, P., Larochelle, H., Laviolette, F., & Lempitsky, V. (2016). Domain-adversarial training of neural networks. *Journal of Machine Learning Research*, *17*(59), 1–35. Retrieved from <https://arxiv.org/abs/1505.07818>
- Gibbons, S. J., & Ringdal, F. (2006). The detection of low magnitude seismic events using array-based waveform correlation. *Geophysical Journal International*, *165*(1), 149–166. <https://doi.org/10.1111/j.1365-246X.2006.02865.x>
- Global Volcanism Program (2013). Volcanoes of the world. (01 Feb 2021). In E. Venke (Ed.), *Smithsonian Institution*. Downloaded 10 Feb 2021. <https://doi.org/10.5479/si.GVP.VOTW4-2013>
- Glorot, X., Bordes, A., & Bengio, Y. (2011). Domain adaptation for large-scale sentiment classification: A deep learning approach. In *Proceedings of the 28th International Conference on Machine Learning ICML 2011*. (pp. 513–520). Bellevue, WA, USA.
- Goitom, B. (2017). *The Nabro volcano, tectonic framework and seismic hazard assessment of Eritrea [doctoral thesis]*. University of Bristol.
- Goitom, B., Oppenheimer, C., Hammond, J. O. S., Grandin, R., Barnie, T., Donovan, A., & Berhe, S. (2015). First recorded eruption of Nabro volcano, Eritrea, 2011. *Bulletin of Volcanology*, *77*(85). <https://doi.org/10.1007/s00445-015-0966-3>
- Hamlyn, J. E., Keir, D., Wright, T. J., Neuberg, J. W., Goitom, B., Hammond, J. O. S., & Grandin, R. (2014). Seismicity and subsidence following the 2011 Nabro eruption, Eritrea: Insights into the plumbing system of an off-rift volcano. *Journal of Geophysical Research: Solid Earth*, *119*, 8267–8282. <https://doi.org/10.1002/2014JB011395>
- Hammond, J., Goitom, B., Kendall, J. M., & Ogubazghi, G. (2011). *Nabro Urgency array [data set]*. International Federation of Digital Seismograph Networks. https://doi.org/10.7914/SN/4H_2011
- Hansen, S. M., & Schmandt, B. (2015). Automated detection and location of microseismicity at Mount St. Helens with a large-N geophone array. *Geophysical Research Letters*, *42*, 7390–7397. <https://doi.org/10.1002/2015GL064848>
- He, T., Zhang, Z., Zhang, H., Zhang, Z., Xie, J., & Li, M. (2019). Bag of tricks for image classification with convolutional neural networks. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. (pp. 558–567). Long Beach, CA, USA. <https://doi.org/10.1109/CVPR.2019.00065>
- Hibert, C., Provost, F., Malet, J. P., Maggi, A., Stumpf, A., & Ferrazzini, V. (2017). Automatic identification of rockfalls and volcano-tectonic earthquakes at the Piton de la Fournaise volcano using a Random Forest algorithm. *Journal of Volcanology and Geothermal Research*, *340*, 130–142. <https://doi.org/10.1016/j.jvolgeores.2017.04.015>
- Huot, F., Biondi, B., & Beroza, G. C. (2018). Jump-starting neural network training for seismic problems. In *SEG Technical Program Expanded Abstracts* (pp. 2191–2195).
- Ioffe, S., & Szegedy, C. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *32nd International Conference on Machine Learning (ICML)* (pp. 448–456). France. Retrieved from <https://arxiv.org/abs/1502.03167>
- Kingma, D. P., & Ba, J. (2014). Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations (ICLR)* (pp. 1–15). Retrieved from <http://arxiv.org/abs/1412.6980>
- Kirkpatrick, J., Pascanu, R., Rabinowitz, N., Veness, J., Desjardins, G., Rusu, A. A., & Hadsell, R. (2017). Overcoming catastrophic forgetting in neural networks. *Proceedings of the National Academy of Sciences*, *114*(13), 3521–3526. <https://doi.org/10.1073/pnas.1611835114>
- Klein, A., Falkner, S., Bartels, S., Henning, P., & Hutter, F. (2017). Fast bayesian optimization of machine learning hyperparameters on large datasets. *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics, PMLR*, *54*, 528–536. Retrieved from <https://arxiv.org/abs/1605.07079>
- Krischer, L., Megies, T., Barsch, R., Beyreuther, M., Lecocq, T., Caudron, C., & Wassermann, J. (2015). ObsPy: A bridge for seismology into the scientific Python ecosystem. *Computational Science & Discovery*, *8*, 014003. <https://doi.org/10.1088/1749-4699/8/1/014003>
- Lahr, J. C., Chouet, B. A., Stephens, C. D., Power, J. A., & Page, R. A. (1994). Earthquake classification, location, and error analysis in a volcanic environment: Implications for the magmatic system of the 1989–1990 eruptions at redboubt volcano, Alaska. *Journal of Volcanology and Geothermal Research*, *62*(1–4), 137–151. [https://doi.org/10.1016/0377-0273\(94\)90031-0](https://doi.org/10.1016/0377-0273(94)90031-0)
- Lapins, S. (2021). *Python notebooks to accompany paper 'A little data goes a long way: Automating seismic phase Arrival picking at Nabro volcano with transfer learning'*. (Version v1.0.0) [Archived GitHub repository]. Zenodo. <https://doi.org/10.5281/zenodo.4558121>
- Lapins, S., Goitom, B., Kendall, J.-M., Werner, M. J., Cashman, K. V. & Hammond, J. O. S. (2021). *Training, validation and test sets for paper 'A little data goes a long way: Automating seismic phase Arrival picking at Nabro volcano with transfer learning'*. Zenodo. <https://doi.org/10.5281/zenodo.4498549>
- Lapins, S., Roman, D. C., Rougier, J., De Angelis, S., Cashman, K. V., & Kendall, J.-M. (2020). An examination of the continuous wavelet transform for volcano-seismic spectral analysis. *Journal of Volcanology and Geothermal Research*, *389*, 106728. <https://doi.org/10.1016/j.jvolgeores.2019.106728>
- Lara, F., Lara-Cueva, R., Larco, J. C., Carrera, E. V., & León, R. (2020). A deep learning approach for automatic recognition of seismic-volcanic events at the cotopaxi volcano. *Journal of Volcanology and Geothermal Research*. *409*, 107142. <https://doi.org/10.1016/j.jvolgeores.2020.107142>
- Leung, O., Duputel, Z., & Ferrazzini, V. (2016). Uncovering the hidden signature of a magmatic recharge at Piton de la Fournaise volcano using small earthquakes. *Geophysical Research Letters*, *43*(9), 4255–4262. <https://doi.org/10.1002/2016GL068383>
- Li, H., Xu, Z., Taylor, G., Studer, C., & Goldstein, T. (2017). Visualizing the loss landscape of neural nets. *Advances in Neural Information Processing Systems (NIPS)*, 6389–6399. Retrieved from <https://arxiv.org/abs/1712.09913>
- Li, W., Duan, L., Xu, D., & Tsang, I. W. (2014). Learning with augmented features for supervised and semi-supervised heterogeneous domain adaptation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *36*(6), 1134–1148. <https://doi.org/10.1109/TPAMI.2013.167>
- Lin, T.-Y., Goyal, P., Girshick, R., He, K., & Dollár, P. (2017). Focal loss for dense object detection. In *2017 IEEE International Conference on Computer Vision (ICCV)* (pp. 2999–3007). Venice, Italy: IEEE. <https://doi.org/10.1109/ICCV.2017.324>

- Lomax, A., Virieux, J., Volant, P., & Berge, C. (2000). Probabilistic earthquake location in 3D and layered models: Introduction of a Metropolis-Gibbs method and comparison with linear locations. In C. H. Thurber, & N. Rabinowitz (Eds.), *Advances in seismic event location* (pp. 101–134). Amsterdam: Kluwer. https://doi.org/10.1007/978-94-015-9536-0_5
- Maclaurin, D., Duvenaud, D., & Adams, R. (2015). Gradient-based hyperparameter optimization through reversible learning. *Proceedings of the 32nd International Conference on Machine Learning, PMLR*, 37, 2113–2122. Retrieved from <https://arxiv.org/abs/1502.03492>
- Maqsood, M., Nazir, F., Khan, U., Aadil, F., Jamal, H., Mehmood, L., & Song, O. (2019). Transfer learning assisted classification and detection of alzheimer's disease stages using 3D MRI scans. *Sensors*, 19(11), 2645. <https://doi.org/10.3390/s19112645>
- McNutt, S. R., & Roman, D. C. (2015). Volcanic seismicity. In *The encyclopedia of volcanoes* (2nd ed., pp. 1011–1034). Elsevier. <https://doi.org/10.1016/B978-0-12-385938-9.00059-6>
- Megies, T., Beyreuther, M., Barsch, R., Krischer, L., & Wassermann, J. (2011). ObsPy—What can it do for data centers and observatories? *Annals of Geophysics*, 54(1), 47–58. <https://doi.org/10.4401/ag-4838>
- Mousavi, S. M., Ellsworth, W. L., Zhu, W., Chuang, L. Y., & Beroza, G. C. (2020). Earthquake transformer—An attentive deep-learning model for simultaneous earthquake detection and phase picking. *Nature Communications*, 11, 3952. <https://doi.org/10.1038/s41467-020-17591-w>
- Mousavi, S. M., Zhu, W., Sheng, Y., & Beroza, G. C. (2019). CRED: A deep residual network of convolutional and recurrent units for earthquake signal detection. *Scientific Reports*, 9, 10267. <https://doi.org/10.1038/s41598-019-45748-1>
- Nair, V., & Hinton, G. E. (2010). Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th International Conference on Machine Learning* (pp. 807–814). Haifa, Israel. <https://doi.org/10.5555/3104322.3104425>
- Pan, S. J., & Yang, Q. (2010). A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering*, 22(10), 1345–1359. <https://doi.org/10.1109/TKDE.2009.191>
- Razavian, A. S., Azizpour, H., Sullivan, J., & Carlsson, S. (2014). CNN features off-the-shelf: An astounding baseline for recognition. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops* (pp. 512–519). Columbus, OH, USA: IEEE. <https://doi.org/10.1109/CVPRW.2014.131>
- Ronneberger, O., Fischer, P., & Brox, T. (2015). U-Net: Convolutional networks for biomedical image segmentation. In N. Navab, J. Hornegger, W. M. Wells, & A. F. Frangi (Eds.), *Medical image computing and computer-assisted intervention (MICCAI) 2015, part III, Lecture notes in computer science* (pp. 234–241). Springer International Publishing. https://doi.org/10.1007/978-3-319-24574-4_28
- Ross, Z. E., Meier, M., & Hauksson, E. (2018). P-wave arrival picking and first-motion polarity determination with deep learning. *Journal of Geophysical Research: Solid Earth*, 123, 5120–5129. <https://doi.org/10.1029/2017JB015251>
- Ross, Z. E., Meier, M., Hauksson, E., & Heaton, T. H. (2018). Generalized seismic phase detection with deep learning. *Bulletin of the Seismological Society of America*, 108(5A), 2894–2901. <https://doi.org/10.1785/0120180080>
- Ross, Z. E., Yue, Y., Meier, M., Hauksson, E., & Heaton, T. H. (2019). PhaseLink: A deep learning approach to seismic phase association. *Journal of Geophysical Research: Solid Earth*, 124, 856–869. <https://doi.org/10.1029/2018JB016674>
- Shelly, D. R., Beroza, G. C., & Ide, S. (2007). Non-volcanic tremor and low-frequency earthquake swarms. *Nature*, 446(7133), 305–307. <https://doi.org/10.1038/nature05666>
- Shorten, C., & Khoshgoftaar, T. M. (2019). A survey on image data augmentation for deep learning. *Journal of Big Data*, 6(1). <https://doi.org/10.1186/s40537-019-0197-0>
- Snoek, J., Rippel, O., Swersky, K., Kiros, R., Satish, N., Sundaram, N., et al. (2015). Scalable bayesian optimization using deep neural networks. *Proceedings of the 32nd International Conference on Machine Learning, PMLR*, 37, 2171–2180. Retrieved from <https://arxiv.org/abs/1502.05700>
- Sun, B., Feng, J., & Saenko, K. (2016). Return of frustratingly easy domain adaptation. In *30th AAAI Conference on Artificial Intelligence (AAAI)* (pp. 2058–2065). Retrieved from <https://arxiv.org/abs/1511.05547>
- Sun, C., Shrivastava, A., Singh, S., & Gupta, A. (2017). Revisiting unreasonable effectiveness of data in deep learning era. In *2017 IEEE International Conference on Computer Vision (ICCV)* (pp. 843–852). Venice. <https://doi.org/10.1109/ICCV.2017.97>
- Tan, Y. J., Waldhauser, F., Ellsworth, W. L., Zhang, M., Zhu, W., Michele, M., et al. (2021). Machine-learning-based high-resolution earthquake catalog reveals how complex fault structures were activated during the 2016–2017 central Italy sequence. *The Seismic Record*, 1(1). <https://doi.org/10.1785/0320210001>
- Theys, N., Campion, R., Clarisse, L., Brenot, H., van Gent, J., Dils, B., & Ferrucci, F. (2013). Volcanic SO₂ fluxes derived from satellite data: A survey using OMI, GOME-2, IASI and MODIS. *Atmospheric Chemistry and Physics*, 13, 5945–5968. <https://doi.org/10.5194/acp-13-5945-2013>
- Titos, M., Bueno, A., Garcia, L., Benitez, C., & Segura, J. C. (2020). Classification of isolated volcano-seismic events based on inductive transfer learning. *IEEE Geoscience and Remote Sensing Letters*, 17(5), 869–873. <https://doi.org/10.1109/LGRS.2019.2931063>
- Tompson, J., Goroshin, R., Jain, A., LeCun, Y., & Bregler, C. (2015). Efficient object localization using convolutional networks. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 648–656). Boston, MA, USA: IEEE. <https://doi.org/10.1109/CVPR.2015.7298664>
- Tran, K. T., Griffin, L. D., Chetty, K., & Vishwakarma, S. (2020). Transfer learning from audio deep learning models for micro-doppler activity recognition. In *2020 IEEE International Radar Conference, (RADAR)* (pp. 584–589). Washington, DC, USA. <https://doi.org/10.1109/RADAR42522.2020.9114643>
- Tzeng, E., Hoffman, J., Darrell, T., & Saenko, K. (2015). Simultaneous deep transfer across domains and tasks. In *2015 IEEE International Conference on Computer Vision (ICCV)* (pp. 4068–4076). Santiago, Chile: IEEE. <https://doi.org/10.1109/ICCV.2015.463>
- van den Ende, M. P. A., & Ampuero, J. P. (2020). Automated seismic source characterization using deep graph neural networks. *Geophysical Research Letters*, 47, 1–11. <https://doi.org/10.1029/2020GL088690>
- van den Oord, A., Dieleman, S., Zen, H., Simonyan, K., Vinyals, O., Graves, A., & Kavukcuoglu, K. (2016). *WaveNet: A generative model for raw audio*. Retrieved from <https://arxiv.org/abs/1609.03499>
- Withers, M., Aster, R., Young, C., Beiriger, J., Harris, M., Moore, S., & Trujillo, J. (1998). A comparison of select trigger algorithms for automated global seismic phase and event detection. *Bulletin of the Seismological Society of America*, 88(1), 95–106.
- Woollam, J., Rietbrock, A., Bueno, A., & De Angelis, S. (2019). Convolutional neural network for seismic phase classification, performance demonstration over a local seismic network. *Seismological Research Letters*, 1–12. <https://doi.org/10.1785/0220180312>
- Yeck, W. L., Patton, J. M., Johnson, C. E., Kragness, D., Benz, H. M., Earle, P. S., et al. (2019). GLASS3: A standalone multiscale seismic detection associator. *Bulletin of the Seismological Society of America*, 109(4), 1469–1478. <https://doi.org/10.1785/0120180308>
- Yosinski, J., Clune, J., Bengio, Y., & Lipson, H. (2014). How transferable are features in deep neural networks? *Advances in Neural Information Processing Systems*, 27. Retrieved from <https://arxiv.org/abs/1411.1792>

- Yu, F., & Koltun, V. (2016). Multi-scale context aggregation by dilated convolutions. In *4th International Conference on Learning Representations (ICLR)*. Retrieved from <https://arxiv.org/abs/1511.07122>
- Zamir, A. R., Sax, A., Shen, W., Guibas, L., Malik, J., & Savarese, S. (2018). Taskonomy: Disentangling task transfer learning. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 3712–3722). Salt Lake City, UT, USA: IEEE. <https://doi.org/10.1109/CVPR.2018.00391>
- Zech, J. R., Badgeley, M. A., Liu, M., Costa, A. B., Titano, J. J., & Oermann, E. K. (2018). Variable generalization performance of a deep learning model to detect pneumonia in chest radiographs: A cross-sectional study. *PLoS Medicine*, *15*(11), 1–17. <https://doi.org/10.1371/journal.pmed.1002683>
- Zhu, W., & Beroza, G. C. (2019). PhaseNet: A deep-neural-network-based seismic arrival-time picking method. *Geophysical Journal International*, *216*(1), 261–273. <https://doi.org/10.1093/gji/ggy423>
- Zhuang, F., Cheng, X., Luo, P., Pan, S. J., & He, Q. (2015). Supervised representation learning: Transfer learning with deep autoencoders. In *IJCAI International Joint Conference on Artificial Intelligence* (pp. 4119–4125). Buenos Aires, Argentina.
- Zhuang, F., Qi, Z., Duan, K., Xi, D., Zhu, Y., Zhu, H., & He, Q. (2020). A comprehensive survey on transfer learning. *Proceedings of the IEEE*, 1–34. <https://doi.org/10.1109/JPROC.2020.3004555>