Seismic signal augmentation to improve generalization of deep neural networks

Weiqiang Zhu*, S. Mostafa Mousavi, and Gregory C. Beroza
Department of Geophysics, Stanford University, Stanford, CA, United States
*Corresponding author: e-mail address: zhuwq@stanford.edu

Contents

1. Introduction

Deep learning has been successfully applied to a wide range of seismic problems, such as earthquake detection (Dokht, Kao, Visser, & Smith, 2019; Mousavi, Zhu, Sheng, & Beroza, 2019; Perol, Gharbi, & Denolle, 2018; Wu et al., 2018; Zhou, Yue, Kong, & Zhou, 2019; Zhu et al., 2019), clustering (Mousavi, Zhu, Ellsworth, & Beroza, 2019), phase detection and picking (Ross, Meier, Hauksson, & Heaton, 2018; Wang, Xiao, Liu, Zhao, & Yao, 2019; Zheng, Lu, Peng, & Jiang, 2018; Zhu & Beroza, 2018), polarity determination (Ross, Meier, & Hauksson, 2018), earthquake

Advances in Geophysics
ISSN 0065-2687
https://doi.org/10.1016/bs.agph.2020.07.003

location (Mousavi & Beroza, 2019; Zhang et al., 2020), magnitude estimation (Mousavi & Beroza, 2020), denoising (Si & Yuan, 2018; Zhu, Mousavi, & Beroza, 2019), phase association (McBrearty, Delorey, & Johnson, 2019; Ross, Yue, Meier, Hauksson, & Heaton, 2019), among others. Based on deep neural network layers, deep learning algorithms exploit structures in seismic data, extract useful features and representations of seismic waveforms, and learn a map to a target distribution of interest, such as probabilities to separate earthquake from non-earthquake signals. Because the performance of a deep neural network improves with the number of diverse training samples, large seismic datasets like STEAD (Mousavi, Sheng, Zhu, & Beroza, 2019) have been created for deep-learning-based research. However, large-scale training datasets do not exist for every problem, e.g., there is a limited number of very large earthquakes, which due to the power-law distribution of earthquake magnitudes are (fortunately) rare. Moreover, building a high-quality large training dataset, with sufficient labeling and quality control, requires significant effort and time. One way to circumvent these problems is through data augmentation, which consists of various techniques to generate new training samples based on collected datasets to expand the size and variety of training samples. Data augmentation has proven successful in avoiding overfitting and improving the generalization of deep learning models trained on small training datasets and thus shows potential for applications on seismic datasets.

"Generalization" commonly is used to refer to the process of recognizing that a specific feature belongs to a larger category. In deep learning, "generalization" denotes the ability of a trained neural network to perform well on data that were not used in its training or validation (e.g., a unique seismic source or seismograms from a region not used in training). Among the factors that affect generalization are the network architectures, optimization techniques, and training datasets. The size, accuracy (of labels), and completeness of the training datasets are key elements for developing a well-performing model. A training dataset may lack any or all of these properties; for this situation, data augmentation can provide an effective option to improve a model's performance. In high-dimension data space (Fig. 1), the limited training samples, including signals and noise, may only span a small subspace and provide weak constraints on the possible decision boundaries learned by the neural network. This can result in poor performance either in the form of low true positives or high false positives. Data augmentation is designed to increase the training sample size and complexity, thus expanding the sampled feature space such that the neural network

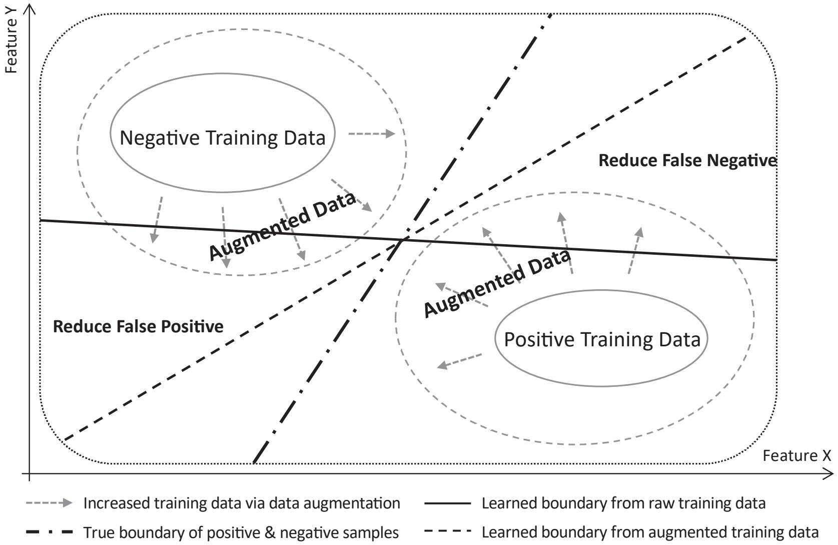
Feature Y
Feature X
Negative Training Data
Augmented Data
Reduce False Negative
Augmented Data
Positive Training Data
Reduce False Positive
Increased training data via data augmentation
Learned boundary from raw training data
True boundary of positive & negative samples
Learned boundary from augmented training data

Fig. 1 Data augmentation expands the data space (small dashed arrows) spanned by collected seismic signals and noise, and results in improved constraints on the decision boundary (the shift from solid to dashed lines). With a broader data space exploited by data augmentation, the trained deep neural networks can generalize better, with reduced false negative and reduced false positive rates, to unseen signals and noise.

can learn an improved decision boundary to reduce both false positives and false negatives and to improve generalization on unseen samples.

Data augmentation is commonly used in training deep neural networks to boost performance in classification and recognition problems for computer vision (Krizhevsky, Sutskever, & Hinton, 2012; Mikołajczyk & Grochowski, 2018; Perez & Wang, 2017; Simard, Steinkraus, Platt, et al., 2003), audio processing (Cui, Goel, & Kingsbury, 2015; Salamon & Bello, 2017) and other areas (Fadaee, Bisazza, & Monz, 2017; Frid-Adar et al., 2018; Um et al., 2017). When implemented correctly, data augmentation reduces the risk of overfitting by increasing the number and variability of training data. It can be especially effective for applications with scarce labeled data. Most data augmentation methods are designed for images, for which semantic meaning can be easily preserved. However, some augmentations appropriate for vision, e.g., horizontal flipping, rotating, and shearing, are inappropriate for seismic data because these processes violate the physical properties of the waveform data of interest. Very few studies exist on augmentation techniques for seismic data. Because seismic data are usually collected and organized in a standard way, i.e., using a fixed time window around the P-wave arrival, obvious augmentations are needed to prevent the neural networks from learning and memorizing any artifacts in how the training data are organized. Less obvious augmentations, such as random shifting, recombining events and noise, and channel (and station) dropout, are consistent with the character of seismic signals and can mitigate bias in training data and increase model performance. In this paper, we demonstrate these techniques and explain how they help to improve generalization of the model to cases of interest including low quality data, complex noise, and closely recorded earthquakes in earthquake swarms. Our examples demonstrate that with appropriate augmentation, we can expand the application of deep learning to smaller datasets than would otherwise be possible for purposes of earthquake monitoring.

2. Benchmark data and training procedure

We collected a small, high signal-to-noise ratio (SNR) training dataset with accurate manual labels of 500 earthquake waveforms from the Northern California Earthquake Data Center (NCEDC) (NCEDC, 2014) recorded before 2018 to demonstrate the application of augmentation for training deep learning models on seismic data. In the same way, we created a validation dataset with another 500 high SNR earthquake waveforms before 2018, used to choose the best model during training. We evaluated the augmentation methods with a much larger test dataset of 10,000 earthquake waveforms recorded in 2018. This choice of ratios between training, validation, and test datasets, is purposely designed to evaluate the effect of augmentation on small datasets. For real applications, we would need to choose a larger sample size for the training dataset than the validation and test datasets. The SNR distribution of the data is shown in Fig. 2. Here the SNR is calculated based on the standard deviations of waveforms before and after the first manually picked P-wave arrival. The distribution of epicentral distances of the test dataset is shown in Fig. 2D, with source-station paths mainly ranging from 0 to 120 km. The training and validation sets (not illustrated) have similar distributions of epicentral distances.

We used the now well-studied phase picking problem as an exemplar for the effects of data augmentation for training deep neural networks. We used the same neural network architecture as PhaseNet (Zhu & Beroza, 2018), i.e., a fully convolutional neural network designed for seismic phase picking problems. To avoid the complex effects of hyperparameter tuning, we removed dropout, learning rate decay, and weight decay (regularization) and keep only batch normalization in the architecture.

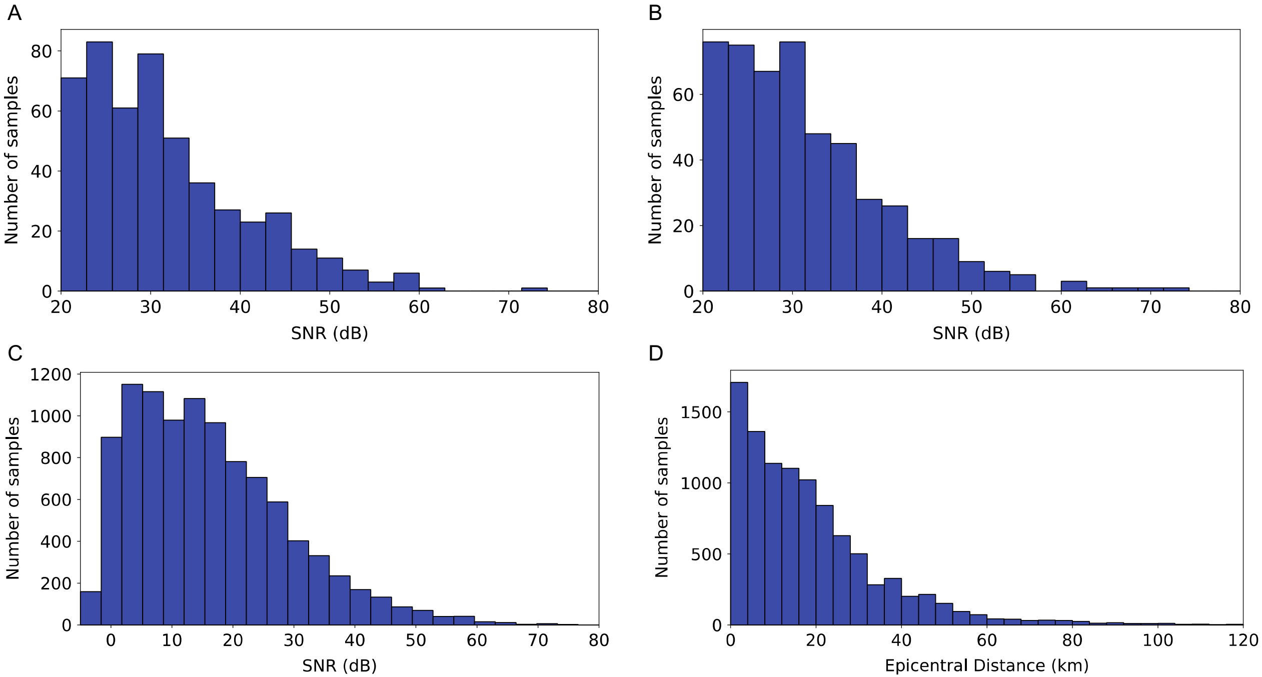

Fig. 2 Statistics of the benchmark datasets: the SNR distributions of (A) 500 high quality training samples (SNR > 20 dB) and (B) 500 high quality validation samples (SNR > 20 dB), both sampled from earthquakes that occurred prior to 2018; (C) the SNR distribution of 10,000 test samples from earthquakes that occurred in 2018; (D) the epicentral distance distribution of the test samples. The training and validation datasets have similar epicentral distance distributions, which are not shown here.

We used the Adam optimization with a learning rate of 0.01, a batch size of 20, and 100 total epochs. Augmentation increases the size of the training dataset by generating a large number of new training samples. Here, we kept the total number of training samples the same in each epoch (500 samples) and apply data augmentation on the fly, so that augmentation increases the variety of training samples in each epoch. The neural network architecture, training procedure, and optimization method will also affect the generalization for different problems and data formats (He et al., 2019). In this paper, we keep the training procedure simple and focus on the effect of data augmentation for seismic data. With suitable modification, our results should apply to related problems like earthquake detection, phase detection, as well as other deep learning applications to seismic data.

3. Augmentations

Many data augmentation techniques have been proposed and applied in image and acoustic signal processing. Here we examine those augmentations of particular relevance to processing seismic data. Note that due to the randomness in initialization and optimization of neural networks and data augmentation, the reported performance will have a certain randomness.

3.1 Random shift

Seismic training data are usually collected based on the phase arrival information from earthquake catalogs. Therefore, it is tempting to define a cutout window based on the time relative to a particular seismic phase such as the first P-wave arrival. Training neural networks using data with a fixed reference time or with a limited time shift of a few seconds around the reference time point can introduce positional bias into the model. Such a neural network model is prone to memorizing the location of the anchor time point rather than learning more general functions from the feature space.

Here, we train three models on data with a 30-s time window using: no random shift, limited random shift from 10 to 15 s, and full random shift from 0 to 30 s. The random shift is applied on the fly so that each training waveform is shifted differently in each epoch. We examine the P-wave arrival predictions by sliding the test waveform from the left side to the right side of the window and record the predicted activation scores at the true P-arrival locations (Fig. 3). For the model trained without random shift, regardless of where the true P-wave arrival is, the neural network continues to predict a high probability score at 10 s, which is the fixed P-wave arrival time during

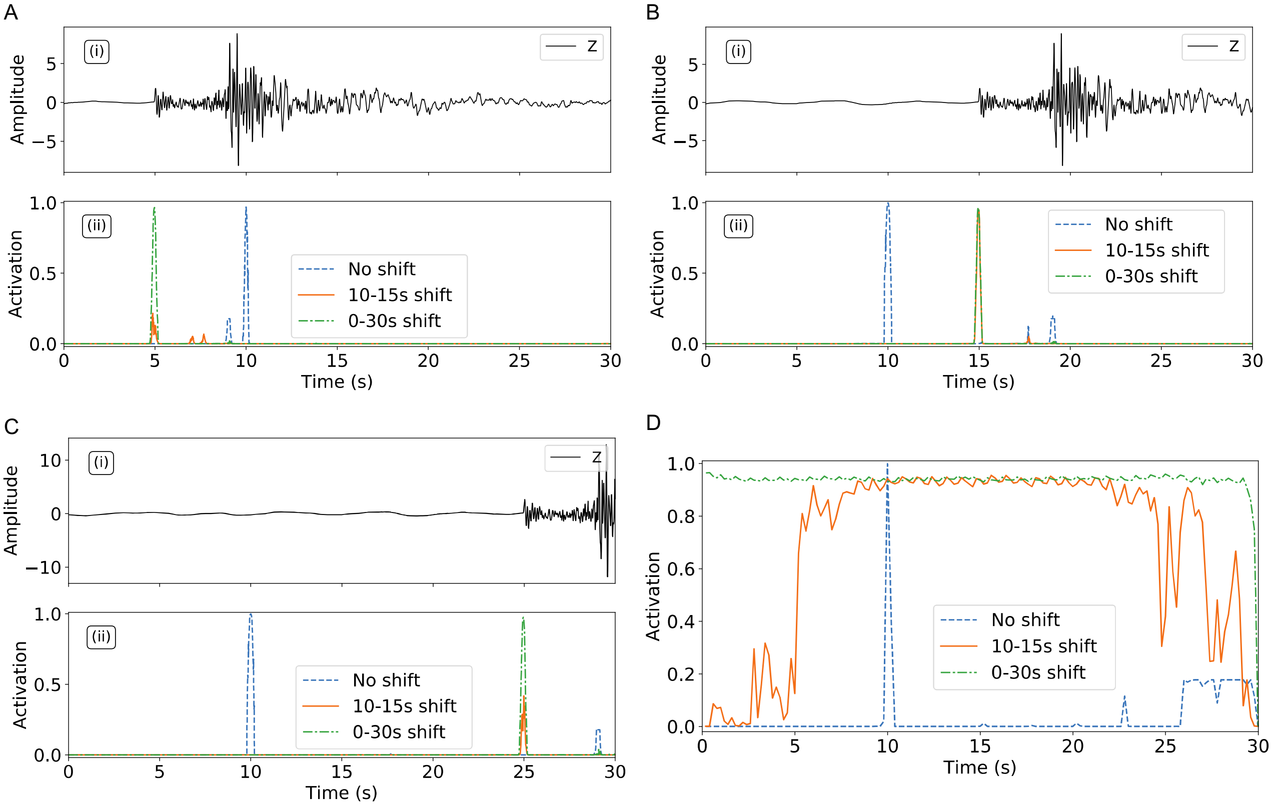

Fig. 3 See figure legend on opposite page.

training (Fig. 3A–C). For the model trained with limited shift within 10–15 s, the result shows a statistical bias that phases are expected to exist within a limited window. The predicted activations fall off at the edges of the time window (Fig. 3D). This bias can be neglected in evaluation if the test dataset is shifted within the same window from 10 to 15 s; however, the bias will result in deteriorated performance when applied to continuous data where the signal arrival time is not known a priori. A very narrow shift range can make a model ineffective for detecting earthquakes outside the shift window used in training. In contrast, the case with random shift from 0 to 30 s has high activation scores across the whole window. This potential bias from random shift could also occur in other deep learning applications, such as earthquake detection, based on a specific window size.

In addition to mitigating the potential bias across the prediction window, random shift can also increase sample variety and improve detection performance. Table 1 compares two implementations of random shift: applying a fixed set of shifting times to the 500 training samples, and applying random shift on the fly so that each sample is shifted differently in each epoch. Random shift on the fly allows a greater variety of time shifting on different training epochs and leads to better performance. In contrast, a fixed set of time shifting for all epochs may only sample limited time points, especially for a small number of training data. In order to effectively apply random shift on the fly within the 30 s window in this case, we cut the training sample to a larger window of 90 s to avoid the need for zero padding. Zero padding may itself introduce a subtle bias by having the neural network learn the transition from zeros to signals and use this artifact as the basis for its predictions. Investigating the distribution of arrival times in a training dataset prior to and after the augmentation is a good practice for deep-learning-based seismic phase detecting/picking models.

Fig. 3 Activation scores from three models trained with different random shift augmentations. (A)–(C) show the predicted probability sequences of neural networks trained with no random shift (blue short-dashed line), random shift within 10–15 s (orange solid line), and random shift within 0–30 s (green alternate short-long dashed line). The test waveform slides from the left to the right edge of the window, and the cases with P-wave arrivals at 5, 15, 25 s are shown in (A)–(C). (D) shows the P-wave arrival predictions at every time point when sliding waveform across the window. Training without random shift leads the neural network to learn a fixed time regardless of the waveform content. Training using incomplete shift between 10 and 15 s leads to activations decay at the edges of the considered window, causing missed detections when applied to continuous waveforms.

Table 1 Comparison between two implementations of random shift: (a) precomputing a fixed random shift for training samples; (b) calculating random shifts on the fly so that each training sample has a different shift in each epoch.

Random shifting type		Precomputed	On the fly
P-wave	Precision	**0.908**	0.902
	Recall	0.550	**0.588**
	F1 score	0.685	**0.712**
S-wave	Precision	0.738	**0.748**
	Recall	0.529	**0.571**
	F1 score	0.616	**0.647**

The higher scores are marked in bold.

3.2 Superimposing events

With the exception of earthquake swarms and aftershock sequences, cataloged earthquakes tend to occur in isolation; thus, training samples contain only one earthquake, commonly referred to as an "event" in the training window. However, training only on single event data can lead neural networks to learn a subtle bias of expecting only one event within the duration of the window and suppressing the detection of smaller events that are also present in the time window. This bias can result in missed events for semantic-segmentation-based methods (Mousavi, Zhu, Sheng, & Beroza, 2019; Zhu & Beroza, 2018), which are designed to detect every event in a time window. We would like a well-trained neural network to perform appropriately on normal earthquakes, but also to generalize to extreme cases, like earthquake swarms and induced earthquakes, when information is dense and earthquakes occur at much more frequent intervals than is usually the case.

An effective augmentation to address the case of multiple events in a short window is to artificially superimpose events in a way that mimics such cases and removes the bias of only one event existing in each window. Superimposition simply means adding two or more time series together, often referred to as "stacking" in seismological parlance. During superimposition, we also apply a random ratio between event amplitudes, which further enhances the neural network's ability to detect smaller earthquakes that occur close in time to larger ones. Fig. 4 shows one example with two earthquakes occurring close to one another in time. The neural network

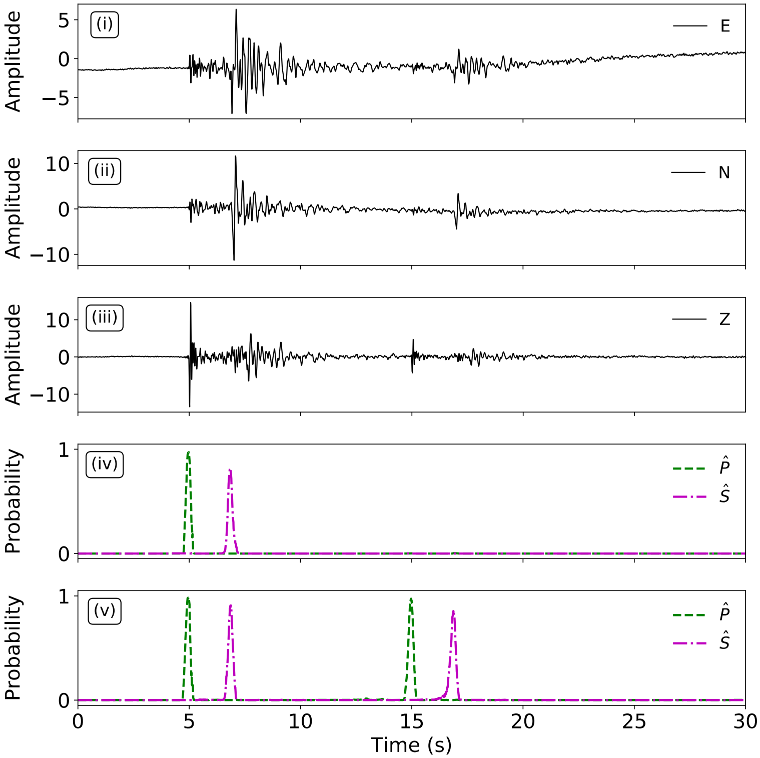

Fig. 4 Comparison of predicted activations for training without and with superimposed events: (i)–(iii) waveforms of ENZ channels; (iv) training without superimposed events; (v) training with superimposed events.

model trained without superimposed events only detects the first large event and neglects the second smaller one (Fig. 4(iv)); however, the model trained with superimposed events predicts the second smaller event with high probability scores for both P and S waves (Fig. 4(v)).

Although event waveforms in real data may completely overlap each other when a station simultaneously records two earthquakes, we avoid synthesizing these cases. These are event waveforms that usually left unused during the manual processing. Because neural networks' performance is highly dependent on training data, these cases have the potential to increase false positives on common seismic waveforms. Thus, we avoid implicitly introducing fully masked events to the training through augmentation. Our observations suggest that a well-trained model should be capable of generalizing to the events with overlapping waveforms. In special applications to realistically replicate an earthquake swarm, stacking much more overlapping events could be useful. Since most of the datasets provide no information regarding the end of waveform (or end of the earthquake coda), we can roughly estimate the end of the earthquake using the time between the P and S arrival times. Another option is to use measurements based on envelope functions similar to those used in coda magnitude estimation.

3.3 Superposing noise

Although most manual labels are selected from high-quality seismic data, a robust neural network should also be able to work on low-quality or otherwise complex data. Superposing noise is a straightforward way to increase the performance of neural networks applied to low signal-to-noise ratio (SNR) data. A distinct advantage of this augmentation is that the high reliability of labels from high SNR data can be retained when the waveforms are superposed with strong noise. Because the augmented weak signals are de-amplified versions of known high SNR signals, their labels are more accurate than those on low SNR signals. By controlling the ratio between the signal and the superposed noise, we can influence the detection limits of the neural network. In particular, by superposing strong noise, we can push the neural network to detect weak signals hidden inside the background noise; however, it should be noted that the potential for false positives may increase as well. Nevertheless, superposing noise is also an effective way to mitigate overfitting on a small training dataset because noise samples are easily obtained from continuous seismic recordings or from synthetically generated random noise.

Fig. 5 compares the neural network's performance with and without superposing noise. It is clear that high scores of precision, recall, and F1 score can be obtained on the high SNR test samples (>20 dB) training with only a small training dataset. However, recall is much lower for the low SNR test samples ($\leq$20 dB) when only high-quality samples are used for training. After training with superposing noise as augmentation, the recall and F1 score are significantly improved for the low SNR data; meanwhile, the performance for high SNR data is maintained. Note that the increase of recall is more significant than precision; this is because the neural network trained with augmented noisy data becomes more sensitive to weak signals buried inside noise and recovers more events, but this may also increase the potential of false positives, thus limits the improvement in precision. Here we have fixed the activation score threshold for comparison. In practice, we can tune a sequence of activation thresholds and generate a

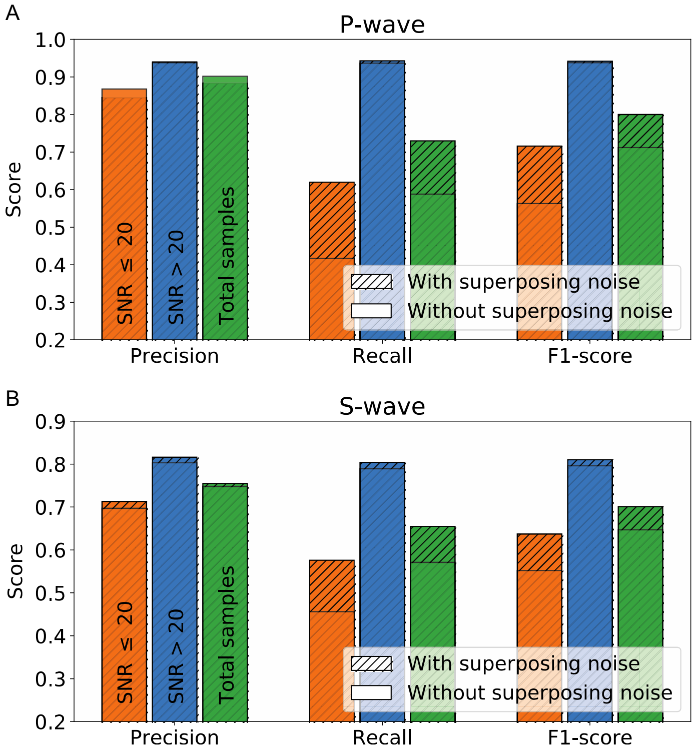

Fig. 5 Comparison of detection performance with and without noise superposition: (A) P-wave arrival; (B) S-wave arrival. For high quality test data ($\text{SNR} > 20\,\text{dB}$), both models achieve a high performance; however, the model with augmentation significantly increases the recall and F1 score for low quality test data ($\text{SNR} \leq 20\,\text{dB}$).

receiver operating characteristic (ROC) curve to balance between precision and recall and to determine a better activation threshold with both improved precision and recall.

The choice of superposed noise depends on the specific environment, such as applications to borehole data, urban data, ocean bottom seismometer (OBS) data. Using real seismic noise recorded by seismic instruments in situ should result in a more realistic augmentation and better performance; however, care must be taken to avoid including undetected events within the noise windows, which would inadvertently increase the mis-labeling rate in the dataset and deteriorate the performance. There is also a risk of biasing the performance for a specific type of instrument if the noise waveforms are not sampled appropriately. Although the use of Gaussian noise would be safe in terms of avoiding mislabeling, the result would be a less realistic representation of real seismic noise, which is usually strongly coherent and non-stationary. Thus, real seismic noise is preferred for augmentation provided that a reliable set of noise samples that have been checked for the existence of non-cataloged earthquakes is available.

3.4 False positive noise

As shown above, superposing noise on earthquake signals helps improve the performance of neural networks for low SNR data. Adding false positive signals (non-earthquake signals) is another way to deal with complex noise effects, such as shaped pulses from urban vibration. This is particularly effective for training neural networks to recognize negative samples (i.e., reducing the false positive rate). Due to the complex noise and non-stationary sources of noise in continuous seismic data, a small training dataset can only cover a limited range of noise. As a result, the neural network trained on limited noise samples may result in many false positives on unseen noise with features similar to seismic signals. To tackle this problem, we can add these false positive noise examples or synthesize similar non-earthquake signals into the training dataset to retrain or fine-tune the neural network to learn features to recognize these false positives and correct their predictions.

Fig. 6 presents a common case in seismic data acquisition when part of the data is missing due to instrumental or telemetry errors. This introduces abrupt changes in continuous data that may result in false positives. These types of false positives are common in traditional methods, like STA/LTA, and they can confuse deep learning methods if this kind of noise is absent from the training dataset. The model trained without appropriate augmentation can produce a false prediction of P and S waves at the abrupt waveform change, as shown in Fig. 6(iv). Adding a small number of similar kinds of noise samples into the training data, however, can effectively suppress false-positive predictions of this type. The same logic applies to other types of common false positive noise, such as impulsive signals from human activities. In practice, identifying different classes of common false positives is challenging due to the need for comprehensive test data and manual examination. To improve the efficiency of identifying false positive samples, active learning (Bergen & Beroza, 2017; Cohn, Atlas, & Ladner, 1994; Kirsch, van Amersfoort, & Gal, 2019) could be one option. Because manually recognizing false positives from a large number of unlabeled samples during application is often difficult, active learning aims to design a strategy to rank these unlabeled samples and annotate the most informative samples

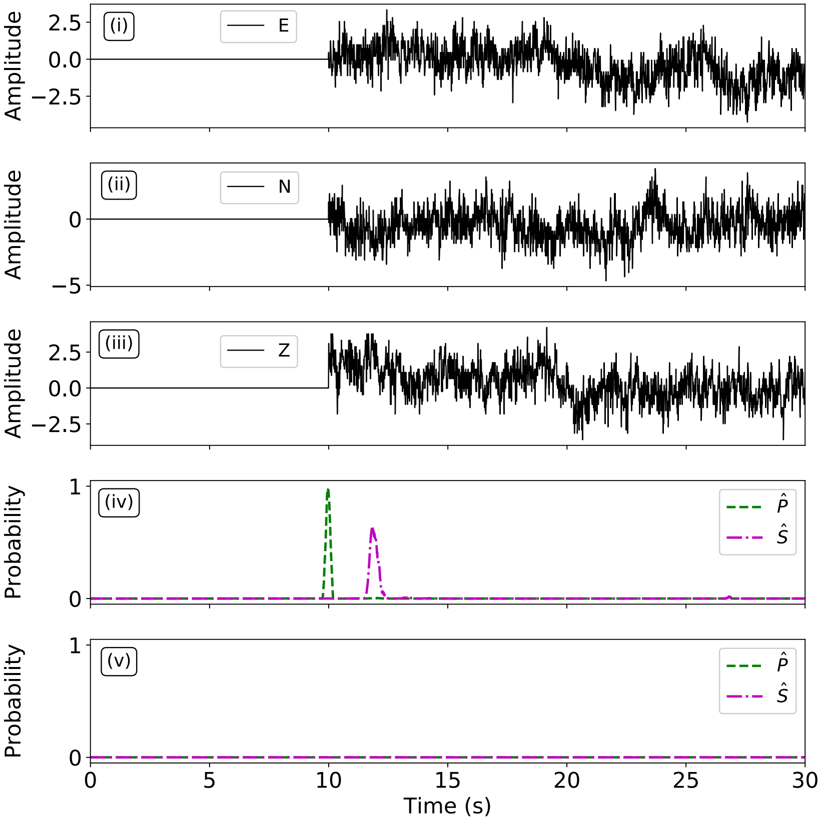

Fig. 6 Comparison of predicted activations before and after adding false positive noise: (i–iii) waveforms of ENZ channels; (iv) training without false positive noise; (v) training with false positive noise.

first, such as samples with the largest uncertainty. These samples are more likely to be recognized as false positives or false negatives, and adding them into training can improve learning efficiency.

3.5 Channel dropout

Three-component seismic data are the most common data form in modern earthquake seismology; however, single channel recordings dominate many historical archives and are still used in some deployments. Moreover, with three-component recordings, it is not uncommon for one of the channels to fail due to either instrument malfunctions or errors in telemetry. Augmentation is an effective strategy to improve the performance of models trained using three-component data on single-channel data. A suitable approach is to use a technique similar to dropout (Gal & Ghahramani, 2016; Srivastava, Hinton, Krizhevsky, Sutskever, & Salakhutdinov, 2014). In the input layer, we randomly drop one or two channels from the EZN input channels. This channel dropout trains the neural network to also predict on data with missing channels. For applications like phase association, where the training is done based on data from multiple stations, we can apply a similar approach to randomly dropping data from part of the stations during training (Zhu, Tai, Mousavi, & Beroza, 2019). This can prevent the network from overfitting on dominant stations and increases the neural network's robustness in the inevitable cases where data from some stations are missing or corrupted.

Table 2 compares the performance between training with and without channel dropout. We apply the trained model on high quality test data (SNR > 20 dB) and examine the performance on different components.

Table 2 Comparison of detection performances on different channels.

		Z	E	N	EN	ENZ
P-wave performance						
With channel dropout	Precision	**0.952**	**0.869**	**0.886**	**0.893**	**0.957**
	Recall	**0.944**	**0.825**	**0.856**	**0.869**	**0.943**
	F1 score	**0.948**	**0.846**	**0.870**	**0.881**	**0.950**
Without channel dropout	Precision	0.944	0.718	0.712	0.795	0.938
	Recall	0.928	0.684	0.705	0.777	0.937
	F1 score	0.936	0.700	0.709	0.786	0.938
S-wave performance						
With channel dropout	Precision	0.523	0.748	0.777	**0.824**	**0.827**
	Recall	**0.436**	**0.732**	**0.767**	**0.808**	**0.810**
	F1 score	**0.476**	**0.740**	**0.772**	**0.816**	**0.818**
Without channel dropout	Precision	**0.630**	**0.771**	**0.793**	0.806	0.803
	Recall	0.019	0.683	0.740	0.787	0.789
	F1 score	0.036	0.724	0.766	0.796	0.796

The higher scores are marked in bold.

Both models show performance similar to that of three-component data; however, the model trained with channel dropout works better on single-component data. The performance on single E-, N-, Z-, and EN-component combinations is instructive and reflects the information learned by the neural network to distinguish P and S waves. For picking the P-wave arrival, performance scores using only the Z-component are similar to those using all ENZ-components, reflecting that the Z-component provides most of the information used for picking P-wave arrivals. In contrast, the horizontal EN-components contain the essential information for picking S-wave arrivals. This is in agreement with the polarization of P and S waves–the P waves appear stronger on the vertical component, while the S waves show up more strongly on the horizontal components.

3.6 Resampling

In deep learning, effective training using imbalanced datasets can be challenging (He & Garcia, 2009; Kotsiantis, Kanellopoulos, Pintelas, et al., 2006). This issue may be especially significant when training a neural network using earthquake signals due to the imbalance of earthquake magnitude distributions. A power law relationship (Gutenberg & Richter, 1944) exists between earthquake magnitude and the number of earthquakes, meaning that the number of large magnitude earthquakes for the training is much more limited compared to the number of small magnitude earthquakes. This imbalance directly impacts applications like magnitude estimation using neural networks (Mousavi & Beroza, 2020). Similar issues can exist for distance, depth, geographic location, tectonic setting, source mechanism, magnitude type, instrument type, and SNR in a specific training set. Station coverage and configuration can also vary significantly among seismic monitoring networks. These imbalances can reduce the generalizability of a model trained on a specific dataset to a broader range of earthquakes. For this reason, it is necessary to investigate data properties during the construction of a training dataset. Based on such preliminary investigations, an appropriate resampling approach can be developed to address possible imbalance problems within a dataset.

Random resampling is a technique to deal with the imbalance issue by oversampling the minority class or undersampling the majority class during training, so that the class distribution does not become biased toward a few specific classes, and better generalization can be achieved by training on a more balanced sample distribution. Resampling can, however, bring about undesirable side effects. Undersampling the majority classes comes with the cost of losing part of the training data and reducing the training size. Extreme oversampling, by repeating a few minority samples of similar magnitudes or from a same region, can also bias the neural network to simply memorizing these samples, which clearly works against generalization. Moreover, oversampling may have limited applications to large earthquakes. Not only are large events rarer, but they are also are more complex compared with small ones. Large earthquakes usually exhibit complex spatial and temporal rupture patterns involving multiple faults. Thus, oversampling may be insufficient to capture the full variety of large magnitude earthquakes. Combining oversampling with the augmentation methods discussed above could be a more effective way to increase both the ratio and variety of the minority samples. Another alternative would be synthesizing training samples from existing instances using more advanced approaches, such as SMOTE (Chawla, Bowyer, Hall, & Kegelmeyer, 2002), ADASYN (He, Bai, Garcia, & Li, 2008), and GAN (Goodfellow et al., 2014).

3.7 Augmentation for synthetic data generation

In some applications, augmentation techniques can be used to generate semi-synthetic data for training. Two such examples are the seismic denoising problem (Zhu, Mousavi, & Beroza, 2019) and earthquake detection on scanned analog-seismograms (Wang, Zhu, Ellsworth, & Beroza, 2019), for which the ground truth (the training target) is unknown and infeasible to obtain from manual labeling. To tackle this problem, we can use augmentation to synthesize input and target pairs from the abundant seismic waveforms. For example, Zhu, Mousavi, and Beroza (2019) generated an accurate denoising mask as the training target for neural networks based on high SNR earthquake signals and a group of noise waveforms. As a result, this augmentation provided a sufficiently large number of training samples by randomly combining signal and noise with a random ratio on the fly during training. In this way, the neural network is trained to learn a challenging inverse process to separate signal and noise in opposite to the forward synthesizing process.

Here we show another example of clipped seismic waveform recovery. Clipped waveforms commonly occur for moderate to large earthquakes recorded on nearby weak motion instruments (Yang & Ben-Zion, 2010; Zhang et al., 2016). Because the true unclipped waveform cannot be observed at the station, we cannot directly get training data from historical waveforms. We can, however, synthesize training data from unclipped waveforms by manually clipping these waveforms. In this way, the input data for the neural network is the synthetically clipped waveforms and the training target the true unclipped waveforms, so that we can easily collect a large number of training data through augmentation. As in denoising, this augmentation has the advantage of being derived from a signal where the (unclipped) ground truth is known and provides an accurate training label. In this case, we use the same network architecture as the other cases but use a mean squared error (MSE) to measure the waveform difference between the recovered and the true unclipped waveforms. Fig. 7 shows the recovered waveforms using the neural network model trained on the synthetic clipped waveforms.

Applying augmentation to synthesizing training data solves the problem of unknown ground truth for some applications. The idea is similar to generating training data using numerical simulations; however, the augmentation method generates the training data based on real seismic waveforms, which is efficient and results in samples that are ipso facto realistic. The trained model can generalize better from the semi-synthetic data to real seismic recordings. If we think of the data generation process as a forward operation, the neural network essentially learns an inverse modeling from the synthesized training data to the true signal of interest that underlies the synthetic data. On the other hand, for cases where not only the label is missing but the real data is also scarce, numerical simulations could become a source for training data, such as finite fault modeling of large complex earthquakes. In this case, we could combine the synthetic earthquake waveforms with real noise to generate training data to improve detection on large magnitude earthquakes. However, the model trained with simulation data may have a generalization issue when applied to real seismic data. Model fine-tuning or transfer learning on a few real seismic waveforms would be needed to narrow the generalization gap. Many other algorithms in computer vision can also be used to bridge the domain gap between simulation and real word, such as adversarial discriminative domain adaptation (Tzeng, Hoffman, Saenko, & Darrell, 2017). The importance of large earthquakes provides strong motivation for future research in this direction.

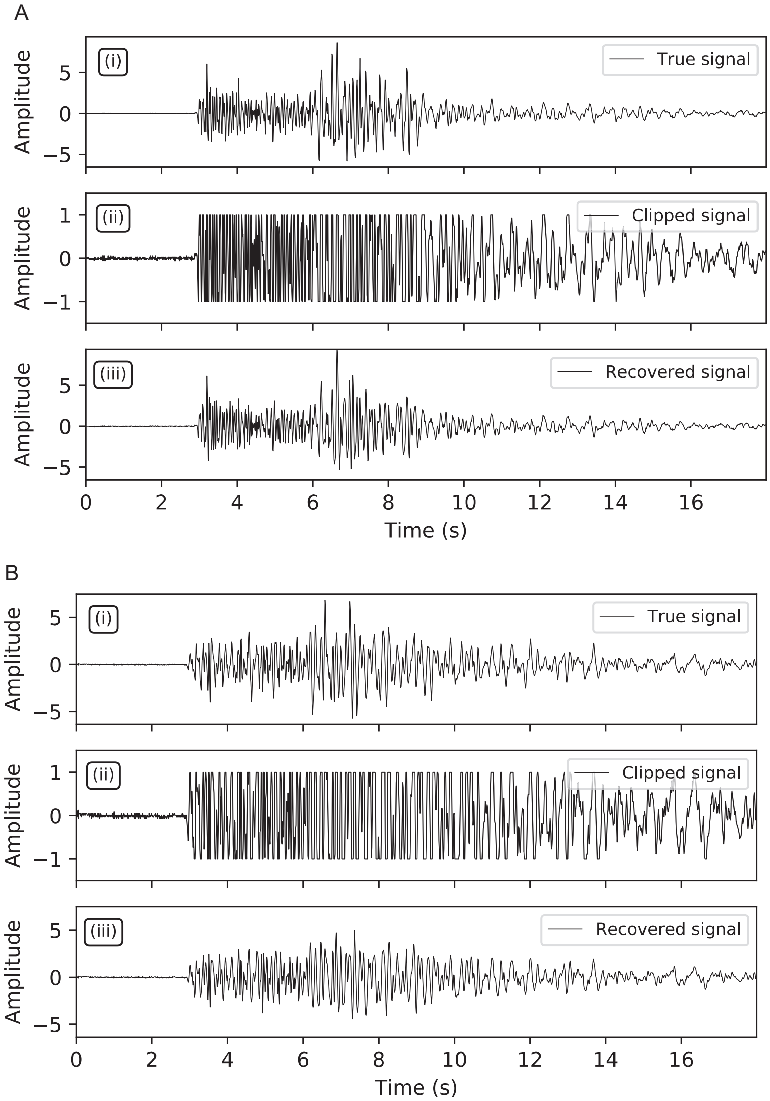

Fig. 7 Two examples (A, B) of clipped waveform recovery: (i) true seismic signals; (ii) manually clipped signals; (iii) recovered signals based on neural networks trained with input-target pairs between (ii) and (i).

4. Discussion

In this paper, we have introduced and discussed several augmentation techniques that can improve the performance of deep learning methods for seismological applications. Combining these augmentations can expediently increase the possible training samples and improve the model generalization even on small training datasets. In addition to the augmentations discussed above, other augmentation methods used in image and speech processing could also be applicable to seismic data, such as (1) filtering seismograms in different narrow frequency bands; (2) time or frequency stretching (Salamon & Bello, 2017); (3) masking part of signal by zeros (DeVries & Taylor, 2017b; Zhong, Zheng, Kang, Li, & Yang, 2017); (4) vertically or possibly horizontally flipping signal; (5) rotating the horizontal components to account station orientation issues and create novel source-station paths; (6) scaling between three components using PCA augmentation (Krizhevsky et al., 2012); and (7) feature space augmentation (DeVries & Taylor, 2017a). Some augmentation, such as time stretching or vertically flipping, can result in potential side effects like changing the phase or polarization information. Hence, some caution is advised in selecting augmentation techniques. Beyond signal-processing-based augmentation, the Generative Adversarial Network (GAN) approach can be used as a synthetic signal generator to make new training samples (Frid-Adar et al., 2018; Li, Meier, Hauksson, Zhan, & Andrews, 2018; Shin et al., 2018; Yi, Walia, & Babyn, 2019). AutoML-based methods, such as AutoAugment, can be used to automatically search for appropriate data augmentations for different problems and datasets (Cubuk, Zoph, Mane, Vasudevan, & Le, 2019). The effectiveness of these methods for seismic data remains as an important area for future research.

The augmentations discussed above were designed during training neural networks, while test time augmentation (He, Zhang, Ren, & Sun, 2016; Krizhevsky et al., 2012) can also help to improve the prediction performance after training. In image classification, several fixed crops and scales are applied to the test image. Similar to ensemble learning, the final prediction score is averaged over these augmentations to improve the overall prediction. The training time and test time augmentations serve different purposes. Training time augmentations, like superposing noise, aim to increase the variety and complexity of training samples, which renders the recognition task much more challenging and pushes the neural networks to learn more accurate decision boundaries between signal and noise (Fig. 1). The test time augmentations aim to make the recognition task easier by sampling certain transforms that properly represent the data features and make the prediction more robust by aggregating different augmentations. For seismic data, data pre-processing methods, such as filtering, can be used as test time augmentation. Filtering can transform the signal into certain high SNR frequency bands covered by the training dataset, thus improving the prediction accuracy on noisy data. Table 3 shows the improved prediction performance by applying 1 Hz high-pass filtering on the test dataset. Another potential method for test time augmentation is to compress the large epicentral distance waveform into a shorter time window in order to mitigate the learning bias due to the imbalanced distribution of epicentral distances (91% samples <40 km) in the training dataset. We applied a 50% down-sampling to compress the waveforms of earthquakes with epicentral distance larger than 40 km (Fig. 2D) into half of the original time window. Table 4 shows the improved prediction performance after time compression. Note that we used prior information that these observations are from 40 km away before applying the compression, which may not be available in applications.

In addition to using data augmentation methods to improve neural network performance for small datasets, generalization of deep learning models is also determined by other factors, such as the neural network architecture, loss functions, optimization methods, and various training techniques, including batch normalization (Ioffe & Szegedy, 2015), layer normalization (Ba, Kiros, & Hinton, 2016), dropout (Srivastava et al., 2014), early stopping (Prechelt, 1998), learning rate decay, weight decay (weight regularization),

Table 3 Detection performance using a 1 Hz high-pass filtering as test time augmentation.

Filtering as test time augmentation		Without filtering	With filtering
P-wave	Precision	0.902	0.902
	Recall	0.588	**0.626**
	F1 score	0.712	**0.739**
S-wave	Precision	**0.748**	0.725
	Recall	0.571	**0.650**
	F1 score	0.647	**0.685**

The higher scores are marked in bold.

Table 4 Detection performance using time compressing as test time augmentation.

Time compressing as test time augmentation		**Without compressing (>40 km)**	**With compressing (>40 km)**
P-wave	Precision	0.785	**0.889**
	Recall	0.374	**0.393**
	F1 score	0.507	**0.545**
S-wave	Precision	0.545	**0.708**
	Recall	0.342	**0.424**
	F1 score	0.421	**0.531**

The higher scores are marked in bold.

label smoothing (Müller, Kornblith, & Hinton, 2019), and model ensemble (Deng & Platt, 2014; He et al., 2016). Most of these training techniques aim to stabilize training and prevent overfitting. Note that overfitting and poor generalization do not necessarily go hand-in-hand. A model without overfitting can still suffer from poor generalization to unseen datasets with significantly different characteristics from the training dataset, while an overfitted model can always have generalization problems. With the development of deep learning, a variety of neural network architectures have been designed, modified and tested on seismic signals, and this trend will continue. The data augmentation techniques discussed here can generally apply to these deep learning models to improve their performance and generalization.

For seismic data, choosing the data processing domains, such as the time domain, time-frequency domain, or wavelet domain, is also important for model performance depending on specific applications. Neural networks can flexibly process multi-dimensional data, like time sequence, image, and videos, which allows presenting seismic signals in either time or frequency domains when designing deep learning models for earthquake detection, denoising, and other problems. For convolutional neural networks, the convolutional kernels have a certain degree of similarity to sine/cosine or wavelet kernels used in Fourier or wavelet transform. The convolutional kernels after training show a variety of frequency and orientation features in image recognition (Krizhevsky et al., 2012). Although a neural network can learn frequency information directly from time domain waveforms, transforming seismic data into time-frequency domain explicitly presents the change of frequency distribution with time, making it easier to capture the frequency information during training. But training and testing in the time-frequency also comes with an expensive computational cost and a loss of high time resolution.

Another approach to addressing the issue of insufficient labeled training data is transfer learning and domain adaptation methods, which are developed to adapt the features and knowledge learned from a large training dataset to improve the training and generalization on a new dataset or task of interest, which often has much less training data (Bengio, 2012; LeCun, Bengio, & Hinton, 2015; Pan & Yang, 2009). For example, pre-trained models on ImageNet dataset (Deng et al., 2009; Oquab, Bottou, Laptev, & Sivic, 2014) has been used for a wide range of problems, such as object detection (Girshick, Donahue, Darrell, & Malik, 2014; Ren, He, Girshick, & Sun, 2015), image segmentation (He, Gkioxari, Dollár, & Girshick, 2017; Long, Shelhamer, & Darrell, 2015), medical image recognition (Tajbakhsh et al., 2016) and remote sensing (Marmanis, Datcu, Esch, & Stilla, 2015). Commonly, transfer learning means transferring low-level features and representations trained on large datasets from the designed task to a different task on a new dataset, while domain adaptation refers to the case of the same task on two different datasets. Because of the similarity among earthquake signals, pre-trained deep neural networks on large datasets like STEAD (Mousavi, Sheng, Zhu, & Beroza, 2019) extract common low-level features for seismic waves, which could be used for applications without enough training data by transfer learning or domain adaptation. Unsupervised pretraining, such as auto-encoding (Vincent, Larochelle, Bengio, & Manzagol, 2008), can also be used to extract good data representation for transfer learning. In contrast to pre-training, self-training is another way to use the extensive unlabeled data (Zoph et al., 2020). Self-training first trains a model on the labeled dataset, then generates pseudo labels on a much large unlabeled dataset. The new pseudo-labeled data is combined with the labeled dataset to train a new model. This same process can be iterated a few times to advance model performance.

5. Conclusions

Data augmentation is an efficient way to improve the performance and generalization of deep neural networks for common cases where labeled data is scarce. We have presented and analyzed data augmentation methods that are well-suited for seismic data. Although we used only a small training dataset, our results show data augmentation can mitigate the bias in training data and improve the performance on a dataset with different statistics. Because data augmentation is independent of the particular neural network model and the computational cost is negligible compared with training, augmentation methods can widely benefit the training of deep learning models on seismic data.

Acknowledgments

We thank Dr. William L. Yeck and Dr. Zachary E. Ross for their constructive comments and suggestions. This work was supported by the Department of Energy Basic Energy Sciences; Award DE-SC0020445.

References

Ba, J. L., Kiros, J. R., & Hinton, G. E. (2016). *Layer normalization*. arXiv preprint arXiv:1607.06450.

Bengio, Y. (2012). Deep learning of representations for unsupervised and transfer learning. In *Proceedings of ICML workshop on unsupervised and transfer learning* (pp. 17–36).

Bergen, K., & Beroza, G. C. (2017). In *Automatic earthquake detection by active learning*. *AGU Fall Meeting 2017*.

Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: Synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*, *16*, 321–357.

Cohn, D., Atlas, L., & Ladner, R. (1994). Improving generalization with active learning. *Machine Learning*, *15*(2), 201–221.

Cubuk, E. D., Zoph, B., Mane, D., Vasudevan, V., & Le, Q. V. (2019). Autoaugment: Learning augmentation strategies from data. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 113–123).

Cui, X., Goel, V., & Kingsbury, B. (2015). Data augmentation for deep neural network acoustic modeling. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, *23*(9), 1469–1477.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., & Fei-Fei, L. (2009). ImageNet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition* (pp. 248–255).

Deng, L., & Platt, J. C. (2014). Ensemble deep learning for speech recognition. In *Fifteenth annual conference of the international speech communication association*.

DeVries, T., & Taylor, G. W. (2017a). *Dataset augmentation in feature space*. arXiv preprint arXiv:1702.05538.

DeVries, T., & Taylor, G. W. (2017b). *Improved regularization of convolutional neural networks with cutout*. arXiv preprint arXiv:1708.04552.

Dokht, R. M., 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.

Fadaee, M., Bisazza, A., & Monz, C. (2017). *Data augmentation for low-resource neural machine translation*. arXiv preprint arXiv:1705.00440.

Frid-Adar, M., Diamant, I., Klang, E., Amitai, M., Goldberger, J., & Greenspan, H. (2018). GAN-based synthetic medical image augmentation for increased CNN performance in liver lesion classification. *Neurocomputing*, *321*, 321–331.

Gal, Y., & Ghahramani, Z. (2016). Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *International conference on machine learning* (pp. 1050–1059).

Girshick, R., Donahue, J., Darrell, T., & Malik, J. (2014). Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 580–587).

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., et al. (2014). Generative adversarial nets. In *Advances in neural information processing systems* (pp. 2672–2680).

Gutenberg, B., & Richter, C. F. (1944). Frequency of earthquakes in California. *Bulletin of the Seismological Society of America*, *34*(4), 185–188.

He, H., Bai, Y., Garcia, E. A., & Li, S. (2008). ADASYN: Adaptive synthetic sampling approach for imbalanced learning. In *2008 IEEE international joint conference on neural networks (IEEE world congress on computational intelligence)* (pp. 1322–1328).

He, H., & Garcia, E. A. (2009). Learning from imbalanced data. *IEEE Transactions on Knowledge and Data Engineering*, *21*(9), 1263–1284.

He, K., Gkioxari, G., Dollár, P., & Girshick, R. (2017). Mask R-CNN. In *Proceedings of the IEEE international conference on computer vision* (pp. 2961–2969).

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 770–778).

He, T., Zhang, Z., Zhang, H., Zhang, Z., Xie, J., & Li, M. (2019). Bag of tricks for image classification with convolutional neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 558–567).

Ioffe, S., & Szegedy, C. (2015). *Batch normalization: Accelerating deep network training by reducing internal covariate shift*. arXiv preprint arXiv:1502.03167.

Kirsch, A., van Amersfoort, J., & Gal, Y. (2019). Batchbald: Efficient and diverse batch acquisition for deep bayesian active learning. In *Advances in neural information processing systems* (pp. 7024–7035).

Kotsiantis, S., Kanellopoulos, D., Pintelas, P., et al. (2006). Handling imbalanced datasets: A review. *GESTS International Transactions on Computer Science and Engineering*, *30*(1), 25–36.

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). ImageNet classification with deep convolutional neural networks. In *Advances in neural information processing systems* (pp. 1097–1105).

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, *521*(7553), 436–444.

Li, Z., Meier, M.-A., Hauksson, E., Zhan, Z., & Andrews, J. (2018). Machine learning seismic wave discrimination: Application to earthquake early warning. *Geophysical Research Letters*, *45*(10), 4773–4779.

Long, J., Shelhamer, E., & Darrell, T. (2015). Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 3431–3440).

Müller, R., Kornblith, S., & Hinton, G. E. (2019). When does label smoothing help? In *Advances in neural information processing systems* (pp. 4696–4705).

Marmanis, D., Datcu, M., Esch, T., & Stilla, U. (2015). Deep learning earth observation classification using ImageNet pretrained networks. *IEEE Geoscience and Remote Sensing Letters*, *13*(1), 105–109.

McBrearty, I. W., Delorey, A. A., & Johnson, P. A. (2019). Pairwise association of seismic arrivals with convolutional neural networks. *Seismological Research Letters*, *90*(2A), 503–509.

Mikołajczyk, A., & Grochowski, M. (2018). Data augmentation for improving deep learning in image classification problem. In *2018 International interdisciplinary PhD workshop (IIPhDW)* (pp. 117–122).

Mousavi, S. M., & Beroza, G. C. (2019). *Bayesian-deep-learning estimation of earthquake location from single-station observations*. arXiv preprint arXiv:1912.01144.

Mousavi, S. M., & Beroza, G. C. (2020). A machine-learning approach for earthquake magnitude estimation. *Geophysical Research Letters*, *47*(1).

Mousavi, S. M., Sheng, Y., Zhu, W., & Beroza, G. C. (2019). STanford EArthquake Dataset (STEAD): A global data set of seismic signals for AI. *IEEE Access*, 7.

Mousavi, S. M., Zhu, W., Ellsworth, W., & Beroza, G. (2019). Unsupervised clustering of seismic signals using deep convolutional autoencoders. *IEEE Geoscience and Remote Sensing Letters*, *16*(11), 1693–1697.

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*(1), 1–14.

NCEDC. (2014). *Northern California earthquake data center*. UC Berkeley Seismological Laboratory.

Oquab, M., Bottou, L., Laptev, I., & Sivic, J. (2014). Learning and transferring mid-level image representations using convolutional neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 1717–1724).

Pan, S. J., & Yang, Q. (2009). A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering*, *22*(10), 1345–1359.

Perez, L., & Wang, J. (2017). *The effectiveness of data augmentation in image classification using deep learning*. arXiv preprint arXiv:1712.04621.

Perol, T., Gharbi, M., & Denolle, M. (2018). Convolutional neural network for earthquake detection and location. *Science Advances*, *4*(2).

Prechelt, L. (1998). Early stopping-but when? In *Neural networks: Tricks of the trade* (pp. 55–69), Springer.

Ren, S., He, K., Girshick, R., & Sun, J. (2015). Faster R-CNN: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems* (pp. 91–99).

Ross, Z. E., Meier, M.-A., & Hauksson, E. (2018). P wave arrival picking and first-motion polarity determination with deep learning. *Journal of Geophysical Research: Solid Earth*, *123*(6), 5120–5129.

Ross, Z. E., Meier, M.-A., Hauksson, E., & Heaton, T. H. (2018). Generalized seismic phase detection with deep learning. *Bulletin of the Seismological Society of America*, *108*(5A), 2894–2901.

Ross, Z. E., Yue, Y., Meier, M.-A., Hauksson, E., & Heaton, T. H. (2019). PhaseLink: A deep learning approach to seismic phase association. *Journal of Geophysical Research: Solid Earth*, *124*(1), 856–869.

Salamon, J., & Bello, J. P. (2017). Deep convolutional neural networks and data augmentation for environmental sound classification. *IEEE Signal Processing Letters*, *24*(3), 279–283.

Shin, H.-C., Tenenholtz, N. A., Rogers, J. K., Schwarz, C. G., Senjem, M. L., Gunter, J. L., et al. (2018). Medical image synthesis for data augmentation and anonymization using generative adversarial networks. In *International workshop on simulation and synthesis in medical imaging* (pp. 1–11).

Si, X., & Yuan, Y. (2018). Random noise attenuation based on residual learning of deep convolutional neural network. In *SEG technical program expanded abstracts 2018* (pp. 1986–1990), Society of Exploration Geophysicists.

Simard, P. Y., Steinkraus, D., Platt, J. C., et al. (2003). Best practices for convolutional neural networks applied to visual document analysis. In *3. Icdar*.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: A simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, *15*(1), 1929–1958.

Tajbakhsh, N., Shin, J. Y., Gurudu, S. R., Hurst, R. T., Kendall, C. B., Gotway, M. B., et al. (2016). Convolutional neural networks for medical image analysis: Full training or fine tuning? *IEEE Transactions on Medical Imaging*, *35*(5), 1299–1312.

Tzeng, E., Hoffman, J., Saenko, K., & Darrell, T. (2017). Adversarial discriminative domain adaptation. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 7167–7176).

Um, T. T., Pfister, F. M., Pichler, D., Endo, S., Lang, M., Hirche, S., et al. (2017). Data augmentation of wearable sensor data for Parkinson's disease monitoring using convolutional neural networks. In *Proceedings of the 19th ACM international conference on multimodal interaction* (pp. 216–220).

Vincent, P., Larochelle, H., Bengio, Y., & Manzagol, P.-A. (2008). Extracting and composing robust features with denoising autoencoders. In *Proceedings of the 25th international conference on machine learning* (pp. 1096–1103).

Wang, J., Xiao, Z., Liu, C., Zhao, D., & Yao, Z. (2019). Deep learning for picking seismic arrival times. *Journal of Geophysical Research: Solid Earth*, *124*(7), 6612–6624.

Wang, K., Zhu, W., Ellsworth, W. L., & Beroza, G. C. (2019). Earthquake detection in develocorder films: An image-based detection neural network for analog seismograms. In *AGU Fall Meeting 2019*.

Wu, Y., Lin, Y., Zhou, Z., Bolton, D. C., Liu, J., & Johnson, P. (2018). DeepDetect: A cascaded regionbased densely connected network for seismic event detection. *IEEE Transactions on Geoscience and Remote Sensing*, *57*(1), 62–75.

Yang, W., & Ben-Zion, Y. (2010). An algorithm for detecting clipped waveforms and suggested correction procedures. *Seismological Research Letters*, *81*(1), 53–62.

Yi, X., Walia, E., & Babyn, P. (2019). Generative adversarial network in medical imaging: A review. *Medical Image Analysis*, 101552.

Zhang, J., Hao, J., Zhao, X., Wang, S., Zhao, L., Wang, W., et al. (2016). Restoration of clipped seismic waveforms using projection onto convex sets method. *Scientific Reports*, *6*, 39056.

Zhang, X., Zhang, J., Yuan, C., Liu, S., Chen, Z., & Li, W. (2020). Locating induced earthquakes with a network of seismic stations in Oklahoma via a deep learning method. *Scientific Reports*, *10*(1), 1–12.

Zheng, J., Lu, J., Peng, S., & Jiang, T. (2018). An automatic microseismic or acoustic emission arrival identification scheme with deep recurrent neural networks. *Geophysical Journal International*, *212*(2), 1389–1397.

Zhong, Z., Zheng, L., Kang, G., Li, S., & Yang, Y. (2017). *Random erasing data augmentation*. arXiv preprint arXiv:1708.04896.

Zhou, Y., Yue, H., Kong, Q., & Zhou, S. (2019). Hybrid event detection and phase-picking algorithm using convolutional and recurrent neural networks. *Seismological Research Letters*, *90*(3), 1079–1087.

Zhu, W., & Beroza, G. C. (2018). *PhaseNet: A deep-neural-network-based seismic arrival time picking method*. arXiv preprint arXiv:1803.03211.

Zhu, W., Mousavi, S. M., & Beroza, G. C. (2019). Seismic signal denoising and decomposition using deep neural networks. *IEEE Transactions on Geoscience and Remote Sensing*, *57*(11), 9476–9488.

Zhu, L., Peng, Z., McClellan, J., Li, C., Yao, D., Li, Z., et al. (2019). Deep learning for seismic phase detection and picking in the aftershock zone of 2008 Mw7. 9 Wenchuan Earthquake. *Physics of the Earth and Planetary Interiors*, *293*.

Zhu, W., Tai, K. S., Mousavi, S. M., & Beroza, G. C. (2019). An end-to-end earthquake monitoring method for joint earthquake detection and association using deep learning. In *AGU Fall Meeting 2019*.

Zoph, B., Ghiasi, G., Lin, T.-Y., Cui, Y., Liu, H., Cubuk, E. D., et al. (2020). *Rethinking pre-training and self-training*. ArXiv:2006.06882 [Cs, Stat].