Article

A Deep Temporal Convolutional Neural Network for Regional and Teleseismic Detection

Joshua Dickey [1,*], Brett Borghetti [1] and William Junek [2]

1 Air Force Institute of Technology; joshuadickey@gmail.com and brett.borghetti@gmail.com
2 Air Force Technical Applications Center; william.junek@us.af.mil
* Author to whom correspondence should be addressed.

Version November 29, 2018 submitted to Preprints

Abstract: The detection of seismic events at regional and teleseismic distances is critical to Nuclear Treaty Monitoring. Traditionally, detecting regional and teleseismic events has required the use of an expensive multi-instrument seismic array; however in this work, we present DeepPick, a novel seismic detection algorithm capable of array-like performance from a single trace. We achieve this directly, by training our single-trace detector against labeled events from an array catalog, and by utilizing a deep temporal convolutional neural network. The training data consists of all arrivals in the International Seismological Centre Catalog for seven seismic arrays over a five year window from 1 Jan 2010 to 1 Jan 2015, yielding a total training set of 608,362 detections. The test set consists of the same seven arrays over a one year window from 1 Jan 2015 to 1 Jan 2016. We report our results by training the algorithm on six of the arrays and testing it on the seventh, so as to demonstrate the transportability and generalization of the technique to new stations. Detection performance against this test set is outstanding. Fixing a type-I error rate of 1%, the algorithm achieves an overall recall rate of 73% on the 141,095 array beam picks in the test set, yielding 102,394 correct detections. This is more than 4 times the 23,259 detections found in the analyst-reviewed single-trace catalogs over the same period, and represents an 8dB improvement in detector sensitivity over current methods. These results demonstrate the potential of our algorithm to significantly enhance the effectiveness of the global treaty monitoring network.

Keywords: Geophysical signal processing; pattern recognition; temporal convolutional neural networks; seismology; deep learning; nuclear treaty monitoring

---

1. Introduction

Adherence to the comprehensive nuclear test ban treaty is currently verified by the detection, location and identification of seismic events, often at regional (>500km) and teleseismic distances (>1000km). Seismic detection is the critical first step in this process, and it is imperative that the events be detected by multiple stations, as this increases the overall accuracy of the final location estimate. As such, maintaining a large network of highly-sensitive seismic detectors is key to the treaty monitoring community [1] [2].

Traditionally, sensitive teleseismic detection has required the use of a multi-instrument seismic array, a strategy which dates back to the Geneva Conference of Experts in 1958 [3]. The sensitivity is achieved through beamforming [4], a spacial filtering technique that relies on a tuned network of interconnected seismometers which form a single station. This technique is extremely effective, however it is quite expensive to implement due to the additional sensors and processing required, and unfortunately, beamforming is inapplicable to single-instrument stations. As such, the vast majority of seismic stations around the globe are simply unable to detect weak regional and teleseismic events.

In this work, we seek to remedy this situation, by creating a detector with array-like performance from a single trace. Building on several recent efforts which apply the power of deep neural networks to the detection of *local events* [5] [6] [7], we seek to apply similar techniques to the detection of *regional and teleseismic events*, traditionally only detectable from a seismic array. Specifically, we seek to answer

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 29 November 2018 doi:10.20944/preprints201811.0612.v1
Peer-reviewed version available at Sensors 2019, 19, 597; doi:10.3390/s19030597

the following research question: Using the analyst reviewed catalog of events from an array beam as ground truth, what is the maximum recall we can achieve from a single-trace detector with an alpha of 0.01?

To answer this question, we present DeepPick, a single-trace detection algorithm capable of detecting 73% of the events in an array beam catalog. The algorithm is based on a deep Temporal Convolutional Neural Network (TCN), and it is trained against more than five billion raw seismic samples and 608,362 labeled seismic arrivals from seven array beam catalogs in the International Monitoring System (IMS) network: TXAR, PDAR, ILAR, BURAR, ABKAR, MKAR and ASAR located in Lajitas Texas, Pinedale Wyoming, Eielson Alaska, Bucovina Romania, Akbulak Kazakhstan, Makanchi Kazakhstan and Alice Springs Australia, respectively. Performance is reported by training the algorithm against five years of data from six of the arrays, and testing it against a full year of data from the seventh, remaining array. All seven arrays are tested in this manner, resulting in a overall recall of 72.6% at an alpha of 0.01. This represents a marked improvement over the 16.5% detection rate found in the traditional single-trace catalogs over the same time period.

Within this work, we present three major contributions to the literature:

- We present our unique high-fidelity dataset, which combines single-trace waveforms with array catalog labels to create a seismic detection training set suitable for deep learning
- We present *exponential sequence tagging*, the novel labeling schema we use to offset the extreme class imbalance inherent in the teleseismic detection task
- We present DeepPick, a single-trace detection algorithm capable of achieving array-level performance from a single sensor

In the remainder of this work, we explore these contributions in detail by first reviewing the related literature, then outlining our methodology, and finally detailing and discussing our results.

2. Related Work

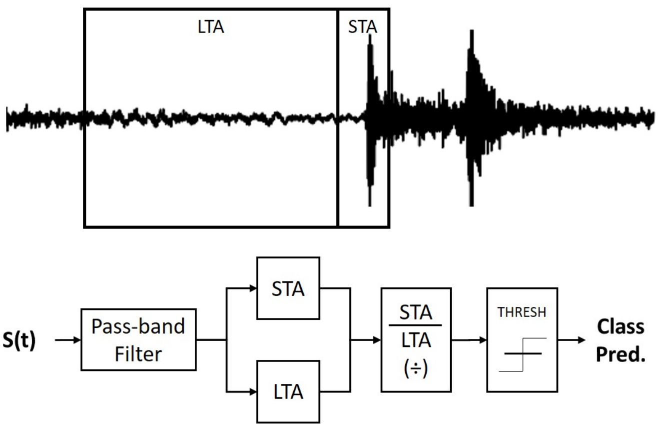

Figure 1. Top: Example seismic waveform, annotated to show the STA and LTA windows. **Bottom:** Diagram detailing the operation of the STA\LTA algorithm.

The most common seismic signal detector is the short-term average, long-term average (STA/LTA) detector [8], first described by Allen in [9]. This detector is a binary classifier, best suited for local events. The basic operation of this detector is detailed in Figure 1. This simple technique enjoys widespread use due to its extreme computational advantage, however its performance is reduced for weaker regional and teleseismic events [10].

To date, one of the most successful techniques for regional and teleseismic signal detection is Beamforming [1] [11], introduced in 1988 [4]. Beamforming gains its effectiveness by linearly combining signals from multiple sensors according to the estimated arrival direction, also known as the

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 29 November 2018 doi:10.20944/preprints201811.0612.v1

Peer-reviewed version available at Sensors 2019, 19, 597; doi:10.3390/s19030597

back-azimuth, allowing it to pick out signals beneath the noise floor of a single sensor. Unfortunately, beamforming is also quite expensive, requiring an interconnected array of seismometers, spread out across a large geographical area. An example array layout is detailed in Figure 2, along with a demonstration of the beamforming technique.

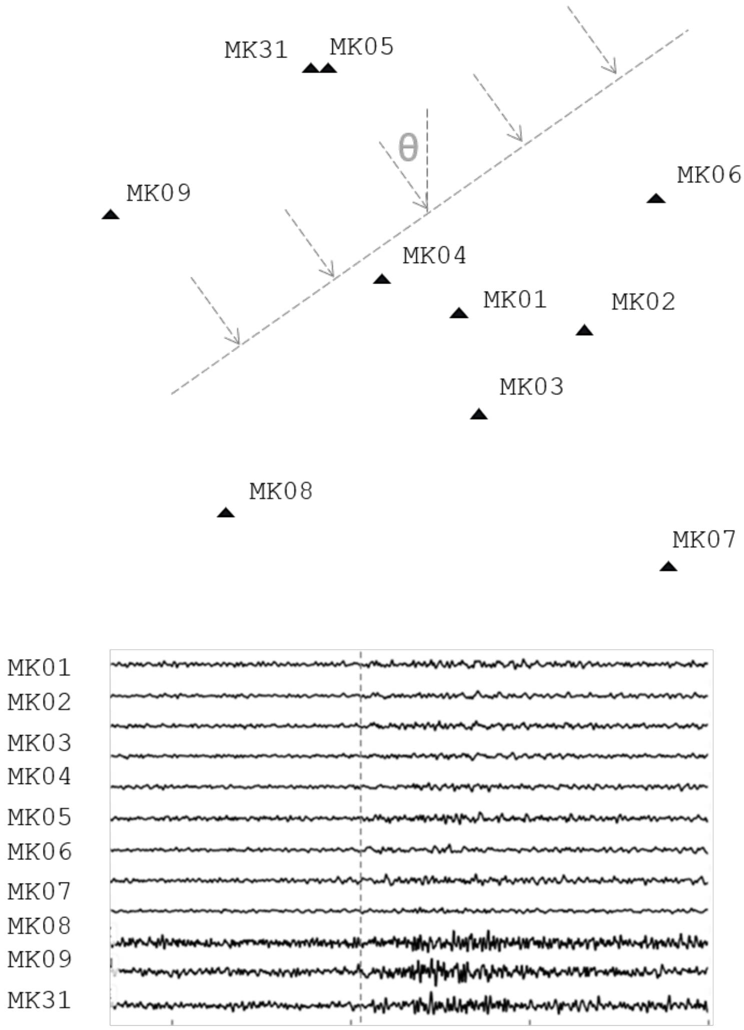

Figure 2. Top: Layout of the 10 element Makanchi Seismic Array, MKAR, in eastern Kazakhstan. The dashed lines illustrate an incoming teleseismic wave with calculated back-azimuth, θ. **Bottom:** Seismic waveforms from an arriving teleseismic event. Beamforming aligns these waveforms via the back-azimuth and wavefront velocity, and then linearly combines them to yield a higher SNR, improving the detection threshold significantly.

Another outstanding technique for the detection of weak teleseismic events is the correlation detector first introduced by [12] and [13] in the early 1990s. Correlation Detectors are a type of Empirical Signal Detector, that work by comparing incoming seismic waveforms to canonical examples in the extant seismic record [14] [15]. This technique is particularly effective for the detection of highly correlated repeating events, even for very weak magnitudes [16]. Unfortunately, to date, this technique is not generally applicable, as only 18% of all global events possess sufficient similarity to be detected with this technique [17].

In [18], the authors demonstrate the power of a richly-featured machine learning based detector. Training a Support Vector Machine against a series of 30 features in the time-frequency plane, they achieved a recall of 97.7% at a type-I error rate of less than 1.3%, for an overall accuracy of 98.2%.

These results compare favorably with STA/LTA. Their work is quite promising, with excellent results, however, the signals investigated were once again limited to strong, local signals; the furthest signals detected had epicenters no more than 5 degrees (∼550km) from the recording sensor.

Recently, several efforts have been made to apply Deep Neural Networks to seismic signal detection. In [5], the researchers utilize a convolutional neural network architecture to perform detection on local seismic signals, formulating the task as a binary classification problem. Their dataset was obtained from two seismic stations in the Oklahoma Geological Survey, consisting of 10s windows with binary class labels: Positive windows were centered around seismic arrival times obtained from an analyst-reviewed arrival catalog, and negative windows were carefully selected to contain no arrival. Against their hold-out test set, they report 100% recall with a high type-I error rate of 1.4%, but by applying a correlation detector to their reported false positives, they determined that a substantial portion of these were actually real detections of very weak events. This work highlights the danger of using conventional catalogs to train such a sensitive detector. Additionally, two major limitations exist in this work. First, because of the extreme care taken to produce 'clean' noise windows in the test set, their reported type-I error rate is not realistic for operational use. Second, their algorithm is applicable only to local events. The short time windows used (10 seconds) prevent the algorithm from being extended to longer-period teleseismic signals.

In [6], the researchers also utilize a deep CNN to perform seismic signal detection on local events. Their dataset consisted of 4.5 million 4 second windows of waveform data recorded and classified by the Southern California Seismic Network. Their task was formulated as a classification problem, assigning one of three classes to each window, P-wave, S-wave and noise. This resulted in 1.5 million windows containing a P-wave arrival, 1.5 million windows containing an S-wave arrival and 1.5 million windows including no arrival. Their validation set consisted of a randomly sampled 25% of the overall data, resulting in 1.1 million seismograms evenly split between the three classes. On the validation set, they report a recall of 96% at a type-I error rate of less than 1%. These results are very impressive, and show that the convolutional neural network is capable of achieving state-of-the-art performance on the seismic signal detection task. Once again, a limitation of this work is that it is applicable only to local signals, and the researchers limited their scope to signals originating within 100km of the recording station. Additionally, due to the fact that only a quarter of a million events were considered, while 1.5 million records were used, it is unclear whether or not there was some leakage from the training set into the validation set.

In [19], the same research team as above considers arrival time estimation. Here they formulate the task as a regression problem, and consider only 4 second windows of data, centered around an arrival, with up to half a second of variance in the arrival time from the center of the window. For this task, they report a mean average error of less than .02 seconds from the analyst recorded picks. Once again, these signals are limited to local events.

Seismic signal detection is an active area of research, with new, improved algorithms being developed capable of achieving near-perfect accuracy for local events. Despite this, little effort has been made to extend detection to regional and teleseismic events without the use of a seismic array. This is exactly the research objective our work shall address.

3. Materials and Methods

Our stated objective is to build a single-trace detection algorithm capable of detecting weak regional and teleseismic signals with array-like performance. We know that such detections are possible using a full seismic array and we have seen the potential for achieving such detections using a deep neural network. With this knowledge as our guide, our approach is to employ a deep TCN model, feed it a single-trace input sequence, and train it to produce an output sequence based on an array beam catalog. In this section, we explore this approach in detail, first defining our dataset, and then describing our modeling strategy.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 29 November 2018 doi:10.20944/preprints201811.0612.v1

Peer-reviewed version available at *Sensors* **2019**, *19*, 597; doi:10.3390/s19030597

3.1. Data Collection

The success of any deep neural network algorithm lies largely in the careful collection and construction of the training data. In this subsection, we present a dataset suitable for training a deep seismic detection algorithm. In particular, we detail two of our major contributions: First, we describe a novel method for obtaining a high-fidelity dataset of single-trace waveforms with labeled arrival times below the noise floor. Second, we present exponential sequence tagging, the unique sequence-to-sequence modeling schema we used to offset the extreme class imbalance inherent in the teleseismic detection task. We conclude this subsection with the details of our finalized training, test and validation datasets.

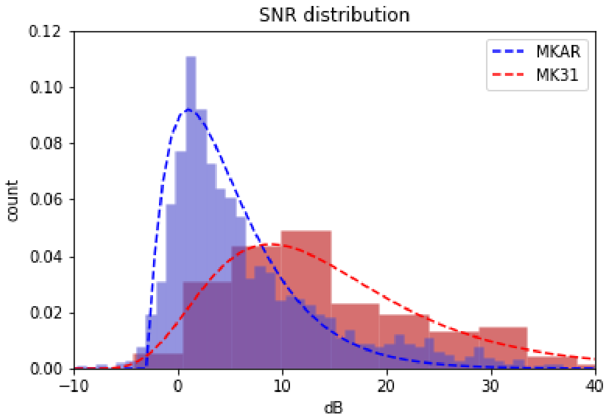

Figure 3. Normalized histograms showing the SNR distributions of detected signals from two seismic arrival catalogs. Both catalogs contain detections for the exact same location, MK31, which is the nominal element of the MKAR seismic array. The MK31 catalog is based on a single-trace detection algorithm applied to the MK31 instrument alone, while the MKAR catalog is based on beam-formed picks from the entire 10-instrument array. The mean SNR detected by the array beam is 8 dB lower than that of the single-trace. This lower detection threshold results in nearly an order of magnitude more detections in the MKAR catalog compared to the MK31 catalog.

3.1.1. High Fidelity Arrival Catalog

At first glance, obtaining a dataset for training a seismic detector would appear to be trivial, as analyst-reviewed arrival catalogs are freely available for millions of seismic events. Unfortunately, despite the rigorous review process and the extensive cross-referencing, each single-trace arrival catalog only contains picks for signals with sufficient strength to be conventionally detectable from within that trace. This is a significant limitation when the goal is to train a detector *more sensitive* than the conventional one. Fortunately, there are certain sensors for which we do have accurate cataloged arrival times for regional and teleseismic signals below the noise floor; namely, the nominal element (usually a broadband 3-channel instrument) of any regional seismic array. Using conventional methods, the nominal element alone is unable to make accurate detections for sub-noise floor events, however the array beam as a whole can make these detections very accurately [11], and the beam arrivals are conveniently aligned to the nominal sensor element of the array. Thus, by obtaining our singe-trace input data from the nominal sensor, and by obtaining our labeled arrivals times from the array beam, we can create a labeled single-trace dataset with signals buried below the noise floor. As an example, Figure 3 demonstrates the significant improvement in detector threshold provided by the Makanchi Array beam in eastern Kazakstan.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 29 November 2018 doi:10.20944/preprints201811.0612.v1

Peer-reviewed version available at Sensors 2019, 19, 597; doi:10.3390/s19030597

3.1.2. Exponential Sequence Tagging

Now that we have established high-fidelity sources for both our waveforms and arrival times, we must formulate them into input/output pairs for training our seismic detector. Typically, seismic detection is formulated as a binary classification task; the input data is partitioned into fixed length windows, each paired with a single Boolean class label: positive class labels are assigned to windows where a signal is present and negative class labels are assigned to windows where signal is absent. This traditional formulation is convenient, as the classes can easily be balanced at training time and it is the common method employed in most recent works in the literature [6], [5], and [18]. However, this method is not well adapted for the detection of regional and teleseismic signals. Teleseismic signals are characterized by long-period features with frequency components as low as 0.01 Hz [20], and the detection of these features necessitates windows that are several minutes in length; unfortunately, this resolution is far too coarse for classification, and often covers multiple arrivals in a single window. As such, there are two conflicting requirements for creating binary classification windows in a teleseismic detection dataset:

- Input windows must contain many samples to capture long-period teleseismic features
- Output labels must cover few samples to allow meaningful temporal resolution for the detection windows

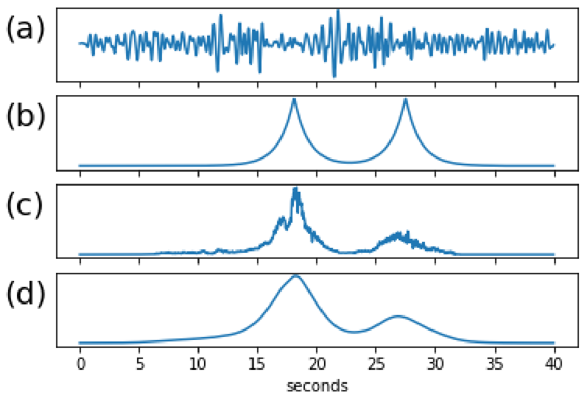

Figure 4. (a): Input Sequence containing two arrivals **(b):** Labeled output sequence using the exponential function. **(c):** Predicted output sequence from the model. **(d):** Cross-correlation of the predicted output sequence with the exponential function.

To resolve this conflict, we reformulate the task. Instead of performing binary classification on each window, we perform regression on each sample, which is known as sequence-to-sequence modeling [21]. Quite simply, the training windows are no longer labeled with a single output Boolean, but instead with an entire output sequence of real-valued numbers; each sample in the input sequence is assigned a corresponding label in the output sequence. But what labels should we assign? A naive formulation is to simply assign a 'one' at each cataloged arrival time and assign a 'zero' everywhere else. This formulation is called sequence tagging [22], and it works well for relatively balanced classes [23]. Unfortunately, binary sequence tagging does not work well for teleseismic detection, as it results in an extreme class imbalance of several orders of magnitude, which hinders learning. For this work, we instead present a novel formulation which we call exponential sequence tagging. This formulation simply builds output sequences that consist of an exponential function applied at each cataloged arrival time, as shown in Figure 4 (b). To be precise, the labels in the output sequence are nominally zero up until a cataloged arrival time, at which point they increase and decrease exponentially, according to the mirrored exponential decay function given in Eq. (1), where λ is the decay rate.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 29 November 2018 doi:10.20944/preprints201811.0612.v1

Peer-reviewed version available at Sensors 2019, 19, 597; doi:10.3390/s19030597

$$y(t) = e^{-\lambda|t|} \tag{1}$$

Because each leg of the mirrored exponential decay function is both monotonic and deterministic, the value at each non-zero label can be used to directly infer the precise arrival time. And because the algorithm learns to match these labels with its output, every non-zero sample in the output is effectively an arrival time estimation. With this in mind, we assign one additional computation to our algorithm at run-time: a cross-correlation of the predicted output sequence with the original exponential decay function. This filters the output and effectively aggregates the arrival time estimates for an even more precise arrival time pick. Figure 4 (c) and (d) shows an example of the predicted output, both before and after this cross-correlation is applied.

3.1.3. Training, Validation and Test Sets

Using this approach to build our training dataset, we obtained a catalog of all local, regional and near-teleseismic arrivals for the seven array beams during a five year period from 1 Jan 2010 to 1 Jan 2015. We generated this catalog through a web query of the International Seismological Centre (ISC) Bulletin for seismic arrivals which can be accessed here: http://www.isc.ac.uk/iscbulletin/search/arrivals/. The corresponding waveforms were then windowed around each arrival (the windows were 6 minutes in total length, sampled at 40Hz for a total of 14400 samples per window), and the raw traces were pulled from the Incorporated Research Institutions for Seismology (IRIS) Database, for the vertical channel of the nominal seismometer for each array (PD31_BHZ, TX31_BHZ, IL31_BHZ, MK31_BHZ, ABK31_BHZ, BUR31_BHZ and AS31_BHZ). This was accomplished via a custom Python script based on ObsPy-1.1.0, and yielded a dataset of 608,362 picks, and a total training size of more than five billion samples. The only pre-processing applied to the raw data was a normalization, detrending and bandpass filtering between 0.02Hz and 10Hz.

From this training dataset, we selected one month of data from each array (1 Jan 2010 to 1 Feb 2010), as a validation set. This validation set was used to tune the models, with final model selection based on validation set performance.

To build our testing dataset, we also obtained a catalog of all local, regional and near-teleseismic arrivals for the seven array beams, in this case during a one year period from 1 Jan 2015 to 1 Jan 2016. This test set is inclusive of 141,095 arrivals in the seven array beam catalogs and 23,259 arrivals in the seven single-trace catalogs. This test set data was not used to train or tune the models, only to report performance against each array. Additionally, to ensure that our reported performance figures are indicative of the expected performance against novel stations, we actually trained seven separate models, each on a different partition of six arrays and tested against the seventh, such that performance for all seven arrays is reported using a model that did not have access to any training data from that array, demonstrating the transportability of our algorithm.

3.2. Modeling

Now that we have defined our dataset, we turn to a precise description of our modeling methodology, detailing the model architecture, hyper-parameter search vectors, and evaluation metrics.

3.2.1. Model Architecture

Our model architecture is based on the Temporal Convolutional Network. TCNs are deep convolutional architectures characterized by layered stacks of dilated causal convolutions and residual connections [24]. These characteristics offer several distinct advantages for a seismic detection algorithm, which we briefly summarize:

- Residual connections allow the model to have high-capacity and stable training
- Causal convolutions allow the model to make predictions on continuous streaming trace data

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 29 November 2018 doi:10.20944/preprints201811.0612.v1

Peer-reviewed version available at Sensors 2019, 19, 597; doi:10.3390/s19030597

- Dilated convolutions allow precise control over the receptive field

The receptive field is of primary importance for time-series modeling, as it explicitly limits the learn-able feature periodicity at a given layer. As such, one of our key design parameters was to ensure adequate receptive field for our algorithm. The equation for calculating the receptive field for a given convolutional layer, l, and dilation rate, d is given in (2):

$$rField(l) = rField(l-1) + [kernelSize - 1] * d \tag{2}$$

Table 1. Layer Parameters for our TCN architecture.

l	k	d	Pad	Input	Output	Receptive Field
1	16	2	30	14400	14400	31
2	16	4	60	14400	14400	91
3	16	16	240	14400	14400	331
4	16	256	3840	14400	14400	4171

Using this equation, we designed our network to have a receptive field of roughly 100 seconds, allowing it to learn long-period features down to 0.01 Hz. We achieved this in just 4 layers, as shown in Table 1. Another key design parameter was to ensure that the dilation rate in each layer remained less than the receptive field in the previous layer, thereby avoiding any gaps in coverage. Notice that this constraint is maintained even for our final layer with a dilation rate of 256, as the previous layer had a receptive field of 331. Our final model architecture is shown in Figure 5.

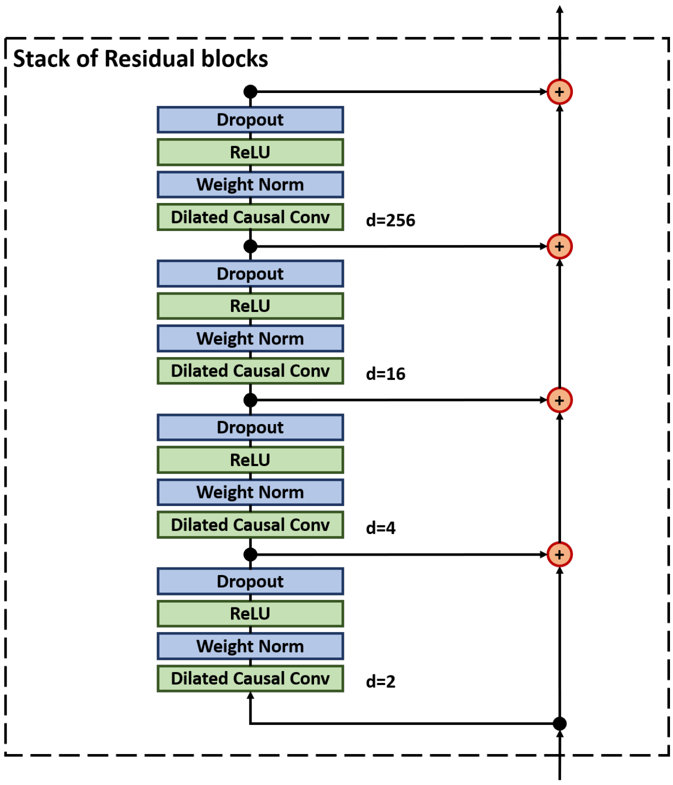

Figure 5. One stack of our chosen TCN architecture.

This basic structure was presented with good results in [24] and proved a good fit for the picking task as well. As such, this basic structure was maintained throughout our formal hyper-parameter search.

Peer-reviewed version available at *Sensors* **2019**, *19*, 597; doi:10.3390/s19030597

3.2.2. Hyper-parameter Search Vectors

Fixing this basic architecture, we engage in a limited hyper-parameter search over two general vectors: the optimal shape for the exponential function, and the optimal capacity for the neural network.

Optimization over the decay rate of the exponential was varied across 3 choices, {0.015, 0.02, 0.04}, selected based on visual inspection. Optimization over model capacity was conducted across two parameters, number of stacks and number of filters. Each parameter was varied across 4 choices, {2, 5, 9, 12} and {5, 10, 15, 20} respectively, ranging from a minimal capacity network (2 stacks with 5 filters and only 3,517 parameters) to a high capacity network (12 stacks with 20 filters and 328,681 parameters). Because these two parameters are highly interrelated, the search was conducted exhaustively, for a total of 16 models. The final hyper-parameter selections were based on validation loss curves.

3.3. Evaluation Criteria

Our stated research objective is to determine the maximum achievable recall of our single-trace detection algorithm against the array beam catalogs. Because recall is a classification metric, and because we have formulated our task as a regression problem, we now carefully proceed to define our methodology for calculating recall:

First, we define our detection window to be 4 seconds, which is identical to the window length used in [19]. Using this, we define the number of Total Positives to be the number of labeled arrivals in the dataset, and we define the number of Total Negatives to be the length of the dataset divided by 4 minus the number of Total Positives, which is a conservative estimate. We next define a predicted arrival to be any peak in the output sequence above a certain threshold, and using this definition, we further define a True Positive to be any predicted arrival within 2 seconds (plus or minus) of a labeled arrival, and a False Positive to be any predicted arrival not within 2 seconds of a labeled arrival. Likewise we define a False Negative to be any labeled arrival not within 2 seconds of a predicted arrival, and a True Negative to be the Total Negatives minus False Negatives. From these definitions, standard equations are used (3) to calculate recall and alpha:

$$
\begin{aligned}
Recall &= \frac{\text{True Positives}}{\text{Total Positives}} \\
alpha &= \frac{\text{False Positives}}{\text{Total Negatives}}
\end{aligned}
\tag{3}
$$

Using these definitions, and treating the analyst-reviewed array beam catalogs as ground truth, we report performance in terms of both receiver operating characteristic (ROC) curves and recall. When reporting recall, we use an alpha of 1%, as this is consistent with the results reported in [5], [6] and [18]. Because our primary interest is toward weak-signal detections, we also report recall as a function of signal to noise ratio (SNR). To do so, we define SNR to be the log ratio between the short-term and long-term average power, as given in Eq. (4), with a short-term window consisting of 5 seconds after the arrival, a long-term window consisting of 40 seconds before the arrival, and a bandpass filter applied from 1.8 to 4.2 Hz.

$$
SNR = 10 * \log_{10}\left(\frac{PWR_{STA}}{PWR_{LTA}}\right) \tag{4}
$$

Additionally, in order to asses the value of our algorithm over existing single-trace methods, we compare our performance directly against the analyst-reviewed single-trace catalogs, noting particularly the increase in detector sensitivity in terms of SNR. And finally, we report our performance for the arrival time estimation task, detailing our mean absolute error across all detected arrivals.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 29 November 2018 doi:10.20944/preprints201811.0612.v1

Peer-reviewed version available at Sensors 2019, 19, 597; doi:10.3390/s19030597

4. Results

In order to define a final model, we explored two hyper-parameter search vectors: exponential decay and model capacity. We varied the decay rate between 0.015 and 0.040, and the results are given in Table 2, which shows 0.020 to be the optimal rate, with optimal recall on the validation set.

Table 2. Decay Rate Optimization.

λ	Recall	MAE
0.015	0.622	0.640
0.020	0.721	0.560
0.040	0.713	0.476

Fixing the decay rate at 0.020, we next varied the overall capacity of the model by increasing both the number of residual stacks, s, and the number of 1D convolutional filters, f. The resultant training curves are given in Figure 6 which shows that model capacity is optimized with 12 stacks and 15 filters, as increasing capacity beyond this point appears to have marginal value. This yields a final model with 12 residual stacks as shown in Figure 5, with 15 filters on each 1D convolution, for a total of 185,311 fully convolutional parameters.

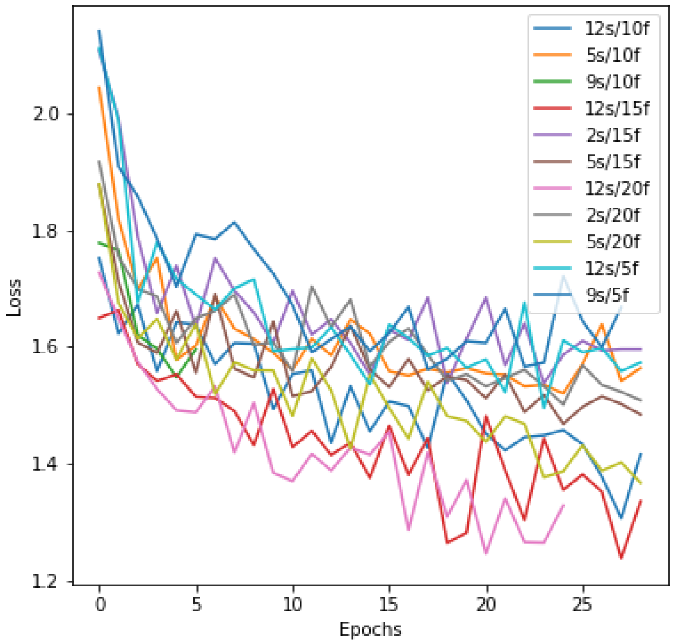

Figure 6. Validation Loss Curves during training. Each curve is labeled according to two hyper-parameters, s: number of residual stacks, and f: number of filters. The number of training epochs for each model were based on early stopping with a patience of 10. Total training time was approximately 200 hrs on an Nvidia GTX 1080 Ti.

Evaluating our final model against the hold out test set, we report our results in Table 3. The results of our algorithm here are ground-breaking. Across the seven arrays, the detector is able to correctly classify 72.57% of the 141,095 array beam picks, yielding 102,394 correct detections. This is more than 4 times the 23,259 detections found in the analyst-reviewed single-trace catalogs for the same period. The ROC curves shown in Figure 7 further illustrate the success of the algorithm. The elbow of

Peer-reviewed version available at Sensors 2019, 19, 597; doi:10.3390/s19030597

the curves are quite tight, with most curves flattening out at an alpha of only 0.3%. In Appendix A, we further explore the performance of our algorithm by plotting several example waveforms for both correct detections and missed detections.

Table 3. Algorithm Performance by Station.

STA	Array Catalog	Single-Trace Catalog		DeepPick Catalog	
	Cataloged Events	Cataloged Events	Detect Rate	Detected Events	Recall (α=1%)
BURAR	4,645	0	0.00%	4,274	92.01%
ABKAR	8,072	0	0.00%	7,136	88.40%
TXAR	16,451	2,228	13.54%	12,884	78.32%
MKAR	40,583	10,493	25.86%	31,253	77.01%
PDAR	12,980	1,657	12.77%	9,512	73.28%
ILAR	20,769	2,563	12.34%	13,386	64.45%
ASAR	37,595	6,318	16.81%	23,948	63.70%
TOTAL	141,095	23,259	16.48%	102,393	72.57%

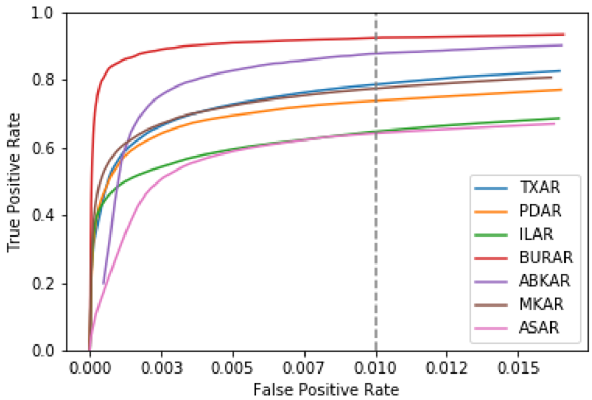

Figure 7. Receiver Operating Characteristic Curves for each of the seven arrays in the hold-out test set. A dashed line is shown in grey, indicating an alpha of 1%.

The primary purpose of our algorithm is to enable detections of weak, distant events. This requires a detector with enough sensitivity to pick out signals near the noise floor. In order to explore our alogorithms performance at this task, we next proceed to examine the ability of our algorithm to detect signals with very low signal to noise ratio. Using the array beam catalog as a baseline, we plot recall as a function of SNR in Figure 8. This demonstrates that DeepPick maintains a more than 95% recall for signals with an SNR of at least 10dB for each of the seven arrays in the test set. The real test, however occurs for signals with an SNR of 10dB or below. These signals are quite difficult to detect from a single trace, as evidenced by the dashed lines in the plot, which represent the detections in the analyst-reviewed single-trace catalogs. Impressively, DeepPick maintains at least an 8dB advantage in sensitivity over the single trace catalogs across all seven test sets arrays.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 29 November 2018 doi:10.20944/preprints201811.0612.v1

Peer-reviewed version available at Sensors 2019, 19, 597; doi:10.3390/s19030597

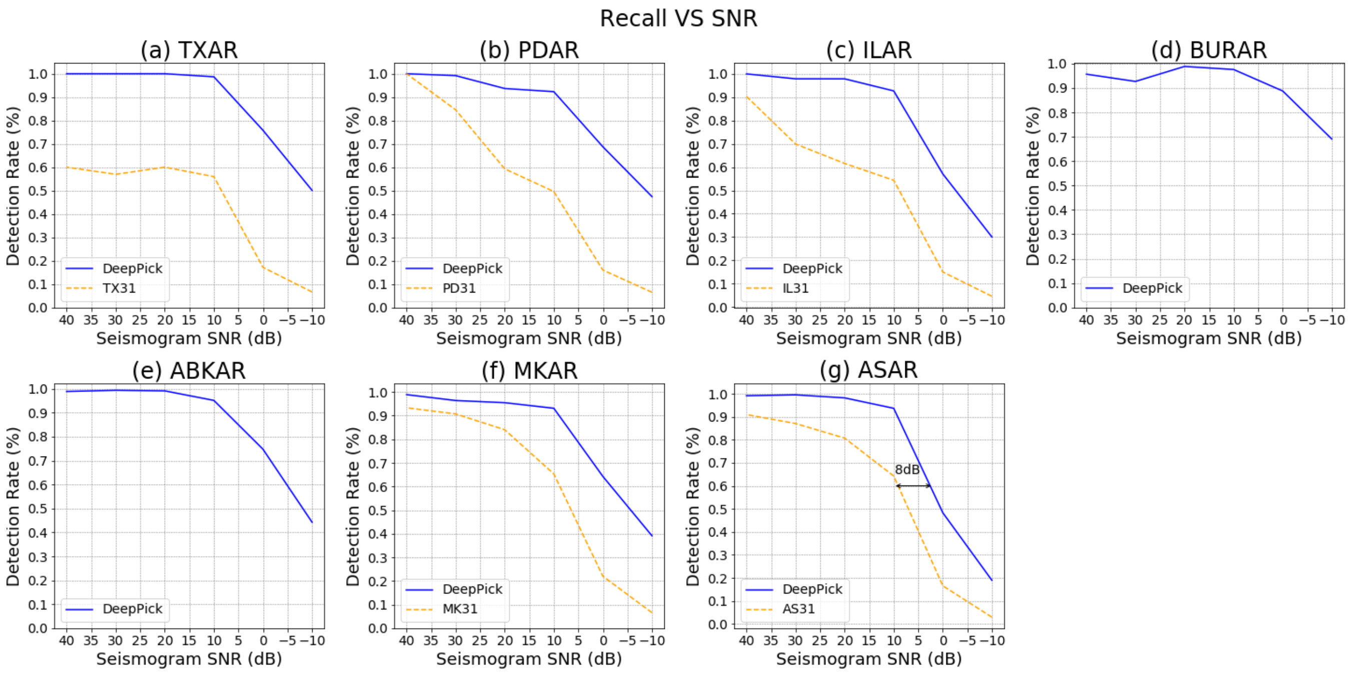

Figure 8. Test-set Recall, reported as a function of SNR, at a fixed alpha of 0.01. Results are compared directly to the detections in the corresponding single-trace catalogs. ***Note:*** *The ISC database does not contain* ***single-trace*** *catalogs for BURAR or ABKAR, however we expect that results would be similar to those depicted in the other five plots.*

Finally, we report the algorithm's performance for the arrival time estimation task[1]. Here, the algorithm achieves a mean average error of 0.61 seconds from the analyst picked arrival times, with a distribution detailed in Figure 9. This plot shows that while the most common histogram bin corresponds to an absolute error of less than 0.025 seconds, the weakest signals are frequently missed by more than a second. This error is high when compared to the accuracy of a dedicated arrival time estimation algorithm, however it should be noted that these estimates are obtained directly from the output of our *detection algorithm*. As such, the 0.61 seconds is excellent when compared to the multi-second classification windows employed by most detectors [5] [6], and is well within the tolerance of a dedicated arrival time algorithm such as that given in [19].

1 We report arrival time error only against true positives, as arrival estimation is distinct from detection for most seismic picking algorithms.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 29 November 2018 doi:10.20944/preprints201811.0612.v1
Peer-reviewed version available at Sensors 2019, 19, 597; doi:10.3390/s19030597

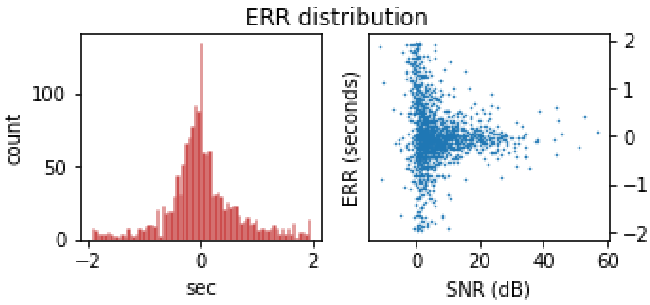

Figure 9. Residual analysis on the errors for the arrival time estimation task. **left:** Histogram showing the distribution of arrival time errors made by the algorithm against the test set, with a bin width of 0.025 seconds. **right:** Scatter-plot showing the distribution of errors with respect to SNR.

5. Discussion

The results in Table 3 demonstrate that the Deep Pick algorithm is capable of achieving a recall of between 64 and 92% against the analyst-reviewed picks from seven array-beam catalogs. The low end of this range, at 64%, still represents a significant improvement over the performance of existing single trace algorithms. However, the spread in our results is quite large, and we now attempt to examine the underlying cause of this performance variance.

The two stations with the worst performance are ILAR and ASAR. Interestingly, these two stations also utilize a different sensor, the Guralp CMG-3TB, from the other five stations, which all use the Geotech KS54000. This shows the importance of training the algorithm on stations with the same instrument type as the stations for which the algorithm is intended to be deployed against operationally. The two stations with the best results are ABKAR and BURAR. Interestingly, due to higher noise levels at these sites, the array catalogs for these two stations contain relatively fewer events with relatively larger magnitudes. This makes the detection of these events a simpler proposition, and the recall rates of 90% and 88% reflect this fact. The final three stations are PDAR, TXAR, and MKAR. These stations utilize a common instrumentation, share similar geology and have similar noise levels; as expected, they also share similar recall rates of 73%, 78% and 77% respectively.

These results show that the primary determinant of algorithm success lies in the degree of similarity between the training stations and the testing station. As such, when deploying this algorithm for operational use it is important to find suitable arrays to train on in order to maximize performance. In any case, the algorithm shows decent performance even when trained across different geographical areas and sensor types.

6. Conclusion

Weak teleseismic event detection is normally only possible using an array of seismic instruments and sophisticated processing techniques. Even recent works in the literature make little attempt to extend single-trace detection algorithms beyond local events. This is primarily due to the lack of available training data, an issue which we address by mining the seismic catalogs in a unique way, building our catalog for an array beam while taking our event waveforms from a single array element. With this training data at our disposal, we find that the combination of temporal convolutions and our unique exponential sequence tagging function forms a powerful tool for weak signal teleseismic detection. In fact, the Deep Pick algorithm is able to accurately detect four times the number of events in the single-trace catalogs in our hold-out test set with an alpha of just 1%.

The findings in this work represent an important step forward in the field of teleseismic detection, and demonstrate that accurate teleseismic event detection is possible from a single seismic instrument. As such, the Deep Pick algorithm has the potential to open up thousands of additional automatic

Peer-reviewed version available at Sensors 2019, 19, 597; doi:10.3390/s19030597

detections to single-instrument seismic stations each year, without the need for additional sensors and equipment.

There is still potential for much improvement on our results. In this work, we develop a single-trace detector, applied only to a single channel of data from a three channel instrument; future work could extend our results to include data from all three channels of the instrument. Furthermore, an application of the same technique to an entire array of channels could also prove interesting, and the potential exists to improve our results significantly by simply incorporating more channels of data. Additionally, the focus of this work has been primarily centered on producing a detector with increased sensitivity and recall, whereas future work could focus on using similar techniques to produce a detector with an even lower false positive rate.

There are also several obvious limitations in our work. Particularly, while we have made a comparison between the detection performance of our algorithm and the analyst-reviewed single-trace catalogs, these catalogs are only an approximation of the performance of the underlying algorithms on which those catalogs were based. It would be quite interesting to directly compare the performance of our algorithm to at least the STA/LTA algorithm across this same year of data in our hold-out test set.

Acknowledgments: The authors would like to thank Dr. Gene Ichniose of Lawrence Livermore National Labs for his generous assistance navigating the online seismic catalogs, as well as Jorge Ramon-Nieves, Chris Barbour, and Vince Gillan of the Air Force Technical Applications Center for their patience and guidance during the analysis process. Additionally, we would like to thank the ISC for providing online access to their arrival catalogs as well as IRIS for providing online access to the raw waveforms. The results presented in this paper are solely the opinion of the authors; they do not represent the official position or policy of the United States Government.

Author Contributions: Conceptualization, Joshua Dickey and William Junek; Formal analysis, Joshua Dickey; Investigation, Joshua Dickey; Methodology, Joshua Dickey; Resources, Brett Borghetti; Supervision, Brett Borghetti; Validation, Brett Borghetti and William Junek; Writing – original draft, Joshua Dickey; Writing – review & editing, Brett Borghetti and William Junek.

Conflicts of Interest: The authors declare no conflict of interest.

Appendix Waveform Examples

In order to more fully represent the capabilities of DeepPick, we now proceed to detail several waveform examples for events that were both missed and detected by the algorithm. To this end we present three groups of signals:

- **Missed Detections:** These waveforms represent cataloged events that were not detected by DeepPick, but were included in the single-trace catalog. As such, they demonstrate some of the limitations of our algorithm.
- **Added Detections:** These waveforms represent cataloged events that were detected by DeepPick but were missed by the single-trace catalog. These Added Detections are verified by the fact that they are included in the Array-Beam catalog, and thus demonstrate the considerable sensitivity of our algorithm to detect weak signals, previously only detectable with an array beam.
- **Unknown Detections:** These waveforms represent detections made by DeepPick that do not correspond to any published events in either the single-trace or array-beam catalogs. More work by a human analyst is required to determine if they are real events or spurious detections; however in this work, we have treated them all as False Positives.

For each waveform in this Appendix, we first plot the raw data (shown on the left) annotated by its cataloged arrival time and instrument channel, along with the ISC eventid, phase, magnitude and distance. We then plot a filtered version of the same waveform (shown on the right), so as to be more easily readable by a human.

We hope that the inclusion of this waveform Appendix will help the reader to better understand the potential limitations of the algorithm, as well as its considerable ability to detect very faint signals from a single trace.

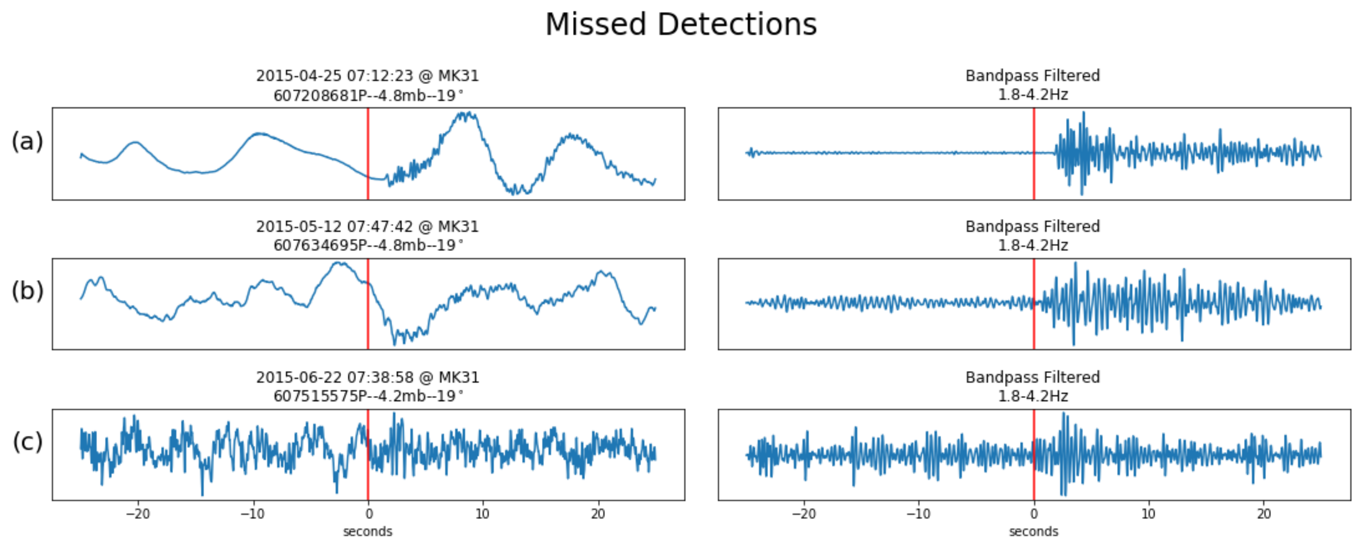

Figure A.1. Three events missed by the DeepPick algorithm; all three events were included in both the single-trace and array-beam catalogs. **(a)** For this event, DeepPick did make a detection, however DeepPick's estimated arrival time was just outside the 2 second margin used by our classifier. **(b) and (c)** DeepPick's output was just below the detection threshold for an alpha of 1%.

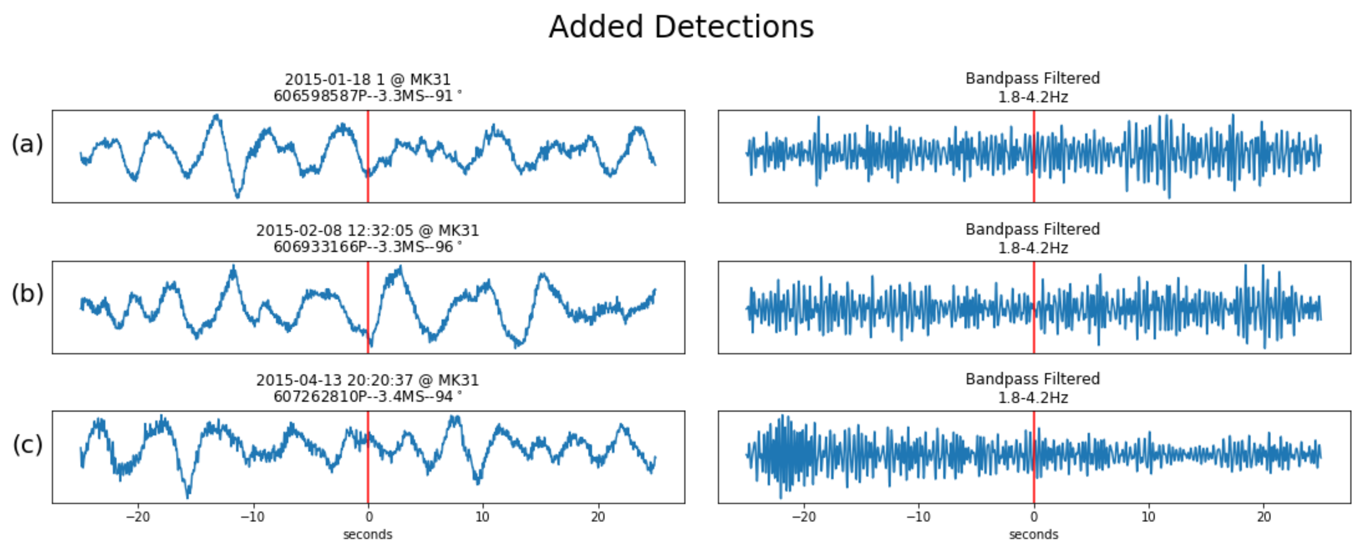

Figure A.2. Three events detected by the DeepPick algorithm; all three events were missing from the single-trace catalog, but included in the array-beam catalog. These examples represent the type of detections previously achievable only with a seismic array, but now possible using a deep single-trace algorithm.

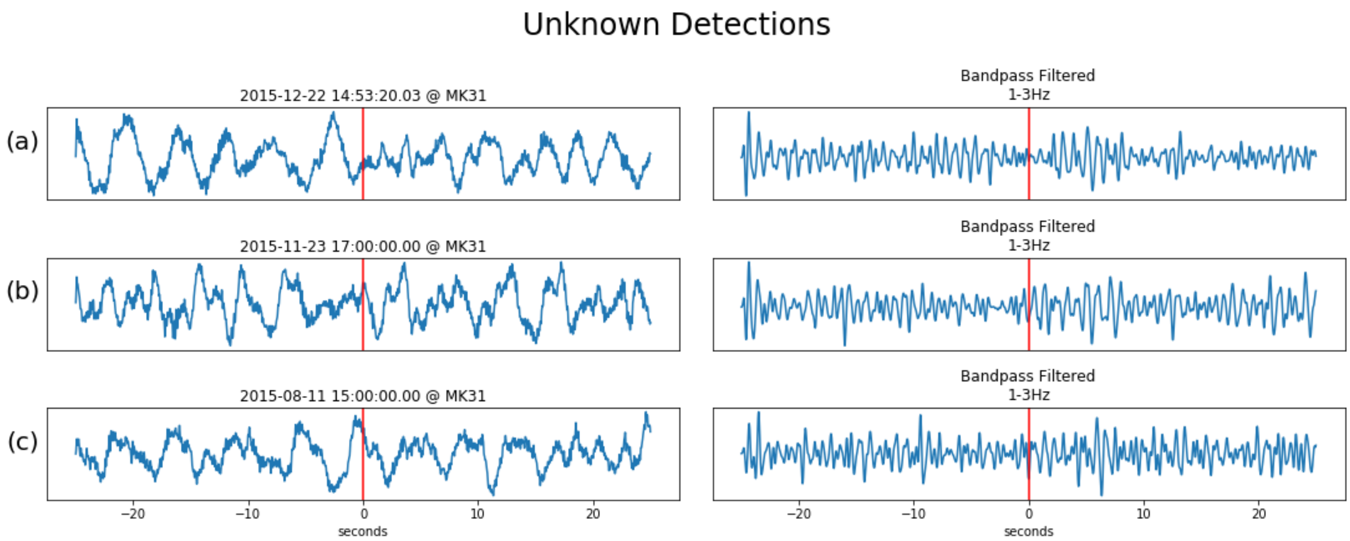

Figure A.3. Three events detected by the DeepPick algorithm; all three events were missing from both the single-trace and array-beam catalogs. These detections require additional work to be either rejected, or added to the seismic record.

1. Ringdal, F.; Husebye, E.S. Application of arrays in the detection, location, and identification of seismic events. *Bulletin of the Seismological Society of America* **1982**, *72*, S201–S224.
2. Anderson, D.N.; Fagan, D.K.; Tinker, M.A.; Kraft, G.D.; Hutchenson, K.D. A Mathematical Statistics Formulation of the Teleseismic Explosion Identification Problem with Multiple Discriminants. *Bulletin of the Seismological Society of America* **2007**, *97*, 1730–1741.
3. Strickland, D.A. Scientists as Negotiators: The 1958 Geneva Conference of Experts. *Midwest Journal of Political Science* **1964**, *8*, 372.
4. Van Veen, B.D.; Buckley, K.M. Beamforming: a versatile approach to spatial filtering. *IEEE ASSP Magazine* **1988**, *5*, 4–24.
5. Perol, T.; Gharbi, M.; Denolle, M. Convolutional Neural Network for Earthquake Detection and Location. *Science Advances* **2018**, *4*, e1700578.
6. Ross, Z.E.; Meier, M.; Hauksson, E.; Heaton, T.H. Generalized Seismic Phase Detection with Deep Learning. *Bulletin of the Seismological Society of America* **2018**, *108*, 2894–2901.
7. Li, Z.; Meier, M.A.; Hauksson, E.; Zhan, Z.; Andrews, J. Machine Learning Seismic Wave Discrimination: Application to Earthquake Early Warning. *Geophysical Research Letters* **2018**, *45*, 4773–4779.
8. FREIBERGER, W.F. AN APPROXIMATE METHOD IN SIGNAL DETECTION. *Quarterly of Applied Mathematics* **1963**, *20*, 373–378.
9. Seismological Society of America., R.V. Automatic earthquake recognition and timing from single traces. *Bulletin of the Seismological Society of America* **1978**, *68*, 1521–1532.
10. Sharma, B.K.; Kumar, A.; Murthy, V.M. Evaluation of seismic events detection algorithms. *Journal of the Geological Society of India* **2010**, *75*, 533–538.
11. Rost, S.; Thomas, C. ARRAY SEISMOLOGY: METHODS AND APPLICATIONS. *Reviews of Geophysics* **2002**, *40*, 2–27.
12. Israelsson, H. Correlation of waveforms from closely spaced regional events. *Bulletin of the Seismological Society of America* **1990**, *80*, 2177–2193.
13. VanDecar, J.C.; Crosson, R.S. Determination of teleseismic relative phase arrival times using multi-channel cross-correlation and least squares. *Bulletin of the Seismological Society of America* **1990**, *80*, 150–169.
14. Junek, W.N.; Vandemark, T.F.; Saults, T.R.; Harris, D.B.; Dodge, D.A.; Matlagh, S.; Ichinose, G.A.; Poffenberger, A.; Kemerait, R.C.; Processing, D.S. Integration of Empirical Signal Detectors into the Detection and Feature Extraction Application at the United States National Data Center. CTBTO Science and Technology Meeting; , 2013.

15. Junek, W.N.; Kværna, T.; Pirli, M.; Schweitzer, J.; Harris, D.B.; Dodge, D.A.; Woods, M.T. Inferring Aftershock Sequence Properties and Tectonic Structure Using Empirical Signal Detectors. *Pure and Applied Geophysics* **2014**, *172*, 359–373.
16. Gibbons, S.J.; Ringdal, F. The detection of low magnitude seismic events using array-based waveform correlation. *Geophysical Journal International* **2006**, *165*, 149–166.
17. Dodge, D.A.; Walter, W.R. Initial Global Seismic Cross-Correlation Results: Implications for Empirical Signal Detectors. *Bulletin of the Seismological Society of America* **2015**, *105*, 240–256.
18. Ruano, A.E.; Madureira, G.; Barros, O.; Khosravani, H.R.; Ruano, M.G.; Ferreira, P.M. Seismic detection using support vector machines. *Neurocomputing* **2014**, *135*, 273–283.
19. Ross, Z.E.; Meier, M.A.; Hauksson, E. P Wave Arrival Picking and First-Motion Polarity Determination With Deep Learning. *Journal of Geophysical Research: Solid Earth* **2018**, *123*, 5120–5129.
20. Ringler, A.T.; Wilson, D.C.; Storm, T.; Marshall, B.; Hutt, C.R.; Holland, A.A. Noise Reduction in Long-Period Seismograms by Way of Array Summing. *Bulletin of the Seismological Society of America* **2016**, *106*, 1991–1997.
21. Sutskever, I.; Vinyals, O.; Le, Q.V. Sequence to Sequence Learning with Neural Networks. In *Advances in Neural Information Processing Systems 27*; Ghahramani, Z.; Welling, M.; Cortes, C.; Lawrence, N.D.; Weinberger, K.Q., Eds.; Curran Associates, Inc., 2014; pp. 3104–3112.
22. Reimers, N.; Gurevych, I. Reporting Score Distributions Makes a Difference: Performance Study of LSTM-networks for Sequence Tagging. *CoRR* **2017**.
23. Yang, Z.; Salakhutdinov, R.; Cohen, W.W. Transfer Learning for Sequence Tagging with Hierarchical Recurrent Networks. *CoRR* **2017**.
24. Bai, S.; Kolter, J.Z.; Koltun, V. An Empirical Evaluation of Generic Convolutional and Recurrent Networks for Sequence Modeling. *CoRR* **2018**, *abs/1803.0*.