GEOPHYSICS, VOL. 89, NO. 1 (JANUARY-FEBRUARY 2024); P. WA279–WA294, 12 FIGS., 8 TABLES.
10.1190/GEO2022-0741.1

A deep learning benchmark for first break detection from hardrock seismic reflection data

Pierre-Luc St-Charles[1], Bruno Rousseau[1], Joumana Ghosn[1], Gilles Bellefleur[2], and Ernst Schetselaar[2]

ABSTRACT

Deep learning techniques are used to tackle a variety of tasks related to seismic data processing and interpretation. Although many works have shown the benefits of deep learning, assessing the generalization capabilities of proposed methods for data acquired in different conditions and geologic environments remains challenging. This is especially true for applications in hardrock environments. The primary factors that impede the adoption of machine learning in geosciences include the lack of publicly available and labeled data sets and the use of inadequate evaluation methodologies. Because machine learning models are prone to overfit and underperform when the data used to train them are site specific, the applicability of these models on new survey data that could be considered "out-of-distribution" is rarely addressed. This is unfortunate, as evaluating predictive models in out-of-distribution settings can provide a good insight into their usefulness in real-world use cases. To tackle these issues, we develop a simple benchmarking methodology for first break picking to evaluate the transferability of deep learning models that are trained across different environments and acquisition conditions. For this, we consider a reflection seismic survey data set acquired at five distinct hardrock mining sites combined with annotations for first break picking. We train and evaluate a baseline deep learning solution based on a U-Net for future comparisons and discuss potential improvements to this approach.

INTRODUCTION

The application of machine learning techniques and methodologies in geoscience and geophysics is an active and popular research area (Yu and Ma, 2021). Exploration seismology in particular has great potential for impactful machine learning contributions, as seismic data are often voluminous and difficult to comprehensively analyze. Concretely, the analysis of seismic data relies on many preprocessing steps (see, e.g., Yilmaz, 2001) that could greatly benefit from automation using machine learning. The primary goal of seismic preprocessing is to eliminate or reduce the impact of noise and artifacts related to surface or acquisition conditions and thus improve the quality of the subsurface reflectivity data. For active source surveys, one of the early and fundamental preprocessing steps is to identify the onset of the signal attributed to the seismic waves originating from the sources. This task is commonly referred to as first break picking. Properly identified first breaks can lead, for example, to static corrections mitigating the effects of the near-surface weathered layer in land surveys.

Although automated first break picking solutions were first introduced decades ago (e.g., Allen, 1982; Coppens, 1985), classic approaches are often fragile when faced with the environment and acquisition conditions seen in practice, especially with land seismic data. In particular, land seismic data with low signal-to-noise ratio are challenging to most classic trace-by-trace algorithms. Various machine learning approaches have been proposed in recent years to learn useful seismic data representations to be leveraged for first break picking. Most approaches presented in the

Manuscript received by the Editor 16 December 2022; revised manuscript received 16 August 2023; published ahead of production 10 October 2023; published online 13 December 2023.
[1]Mila — Québec Artificial Intelligence Institute, Applied Machine Learning Research Team, Montréal, Québec, Canada. E-mail: pierreluc.stcharles@mila.quebec (corresponding author); bruno.rousseau@mila.quebec; joumana.ghosn@mila.quebec.
[2]Geological Survey of Canada, Natural Resources Canada, Ottawa, Ontario, Canada. E-mail: gilles.bellefleur@NRCan-RNCan.gc.ca; ernst.schetselaar@NRCan-RNCan.gc.ca.

literature rely on convolutional neural networks (CNNs) and especially on the U-Net architecture of Ronneberger et al. (2015), which is often modified with modern neural network layers and blocks (Cova et al., 2020; Fernhout et al., 2020; Ma et al., 2020; Yuan et al., 2020; Zheng et al., 2020; Zwartjes et al., 2020; Nikita et al., 2021; Han et al., 2022; Zwartjes and Yoo, 2022). Unfortunately, it is difficult to compare these different approaches to each other because they are applied to different (and often private) data sets. These works also report different metrics or mostly show qualitative results. Finally, it can be unclear how well a trained model would generalize to a new survey data set unseen during training. This is because models are either trained and validated on a single survey data set, or, when multiple surveys are available, they are mixed randomly across different sets for training, validation, and testing.

Recent efforts have been made to gather and study seismic data sets using machine learning (Alaudah et al., 2019; Dumont et al., 2020; Magrini et al., 2020), but the unavailability of multisurvey data sets is still a major impediment for researchers, as it makes the transfer of predictive models precarious across surveys. This is especially true for the field of reflection seismology for mineral exploration in hardrock environments, which suffers from the lack of a public multisurvey labeled data set or benchmark.

In this work, we introduce a data set of land seismic surveys captured across multiple mining sites and labeled for first break picks. This data set is used to define a sound evaluation protocol that can be used to assess how well automatic first break picking models generalize to new unseen testing sites that could be considered out-of-distribution with respect to the training data. Following this protocol, we propose a benchmark for the evaluation of future automated first break picking solutions with baseline results obtained using a U-Net-based deep learning model.

To the best of our knowledge, this is the first public contribution of a curated multisurvey seismic data set focused on crystalline hardrock environments. The size of our data set, which contains millions of seismic traces with labeled first break picks, will allow researchers to assess how well their predictive models generalize across varied conditions. Each of the 3D seismic surveys focuses on a massive sulfide deposit hosted in metasedimentary or metaigneous crystalline rocks. Our benchmark's evaluation protocol is established so that the performance of predictive models for first break picking is directly measured on sites unseen during training. This is aligned with first break picking use cases where automatic methods are applied to new data sets for which no annotated first breaks are yet available. The baseline evaluation results that we provide are based on a model that interprets receiver-line gathers as images. We also describe several ideas on how to improve upon this design and how to incorporate prior geophysical knowledge to further improve model performance.

The paper is organized as follows. First, we describe the seismic data used in this work and detail some of the challenges of working with land seismic surveys. Then, we describe the task of automated first break picking in seismic traces and detail popular approaches from the literature based on machine learning. Next, we present our proposed benchmark methodology for first break picking across different surveys as well as how we implement a robust method to provide a performance baseline. Finally, we discuss improvements for this baseline and new ways to interpret the seismic data for future works.

SEISMIC DATA

Our data set is composed of five 3D surveys acquired at unique mining sites, four in Canada and one in Finland. The Canadian sites are referred to as "Lalor," "Brunswick," "Halfmile" (short for "Halfmile Lake"), and "Sudbury"; and the Finnish site is referred to as "Kevitsa." The data were acquired with dynamite sources for all Canadian sites, whereas hydraulic hammers, also known as vibsist seismic sources (Yordkayhun et al., 2009), were used at Kevitsa in combination with dynamite. Geophones were used at all sites except at Lalor where microelectro-mechanical systems (MEMS) accelerometers were used to record the seismic data. The sampling rate of the recordings varies across the surveys: it can be either 1 ms (Lalor) or 2 ms (Brunswick, Halfmile, and Sudbury). Two acquisition systems were used at Kevitsa, each configured with different sampling rates (1 or 2 ms) (see Malehmir et al., 2012). We used a merged version of the Kevista data, which have a uniform sampling rate of 1 ms. The traces at each receiver were sampled to preserve the first 751, 1001, or 1501 samples, depending on the survey. The sample ranges include first arrivals and sufficient data at far offsets. With the exception of the resampling of the Kevitsa data to 1 ms, all seismic data used as input to the CNNs were kept in their original field state without the application of any preprocessing (i.e., no MEMS-to-geophone conversion or other filtering). The rationale for not applying any preprocessing is to keep data sets as they are typically used for first break picking.

For first break picking, we define each seismic data set as a collection of recordings of traces across each receiver line for each shot. A "shot gather" is defined as the collection of all traces for a given shot. We further define a "line gather" as the collection of traces for a given receiver line and shot: a shot gather is thus composed of multiple line gathers. Examples of three line gathers taken from a common shot are shown in Figure 1 for each Canadian site. Additional details on the acquisition of the 3D seismic surveys and key geologic results can be found in Bellefleur et al. (2015) for Lalor, Cheraghi et al. (2012) for Brunswick, Malehmir and Bellefleur (2009) for Halfmile, Milkereit et al. (2000) for Sudbury, and Malehmir et al. (2012) and Valasti et al. (2012) for Kevitsa.

The annotation of the first breaks can be accomplished visually based on domain knowledge and specialized software tools. The exact location of the first break in a line gather can however be ambiguous. The first break can be defined at one of three moments within the first-arrival window of the seismic trace, following the SEG normal polarity convention (see Veeken, 2007): (1) when the background noise starts being disrupted by the seismic event (the onset), (2) when the amplitude reaches its first minimum (the trough), or (3) when the amplitude reaches its first maximum (the peak). Annotating the onset is the most reliable way to avoid issues caused by phase inversions along a receiver line, but the onset is not always easily identifiable due to background noise. Similarly, identification of the first trough on seismic data with a low signal-to-noise ratio can be challenging; this explains why maximum peaks are sometimes used. We highlight that consistent annotations are required to train good supervised machine learning models.

To enable the training of machine learning models on these data, we relied on software-generated and human-validated annotations for a substantial fraction of the recorded traces following the trough convention. Each survey contains bad picks due to a variety of factors such as the choice of autopicking method, its performance on noisy data, and the amount of editing performed on picks. The latter

is often a function of the volume of data and time allocated for manual first break picking by experts. A visual inspection revealed that picks from Halfmile, Brunswick, and Lalor are more consistent (and thus of higher quality) compared with Sudbury and Kevitsa, where stronger noise levels make the annotation process more difficult due to ambiguities. In addition, in contrast to the other sites, picks from Lalor followed the onset annotation convention. Thus, they were shifted to the trough convention by relying on an automatic tool that fine tuned the picks to the deepest trough in a bidirectional window of ±5ms centered on the original picks. A visual inspection of a sample of the transformed picks suggests that this approach works well on traces with high signal-to-noise ratios but is imperfect when the noise level is high. These issues seem to impact only a small percentage of the data. Note that traces across all sites are not systematically annotated because low signal-to-noise ratios sometimes make the process too ambiguous. Consequently, we deem a whole line gather to be invalid if all of its composing traces are missing an annotation. Although such data may be of interest in an unsupervised machine learning setting, for line gather-based supervised learning, these provide no useful signal. Finally, some valid line gathers were ultimately rejected following a visual inspection to only keep gathers whose first break annotations all seem roughly located over samples with any significant seismic activity. We stress that this final sanitization step was conducted at the line gather level: this allowed the removal of line gathers that were largely ambiguous but did not prevent the presence of some noisy picks within line gathers that looked mostly correct. A detailed summary of the studied sites is presented in Tables 1 and 2.

AUTOMATIC FIRST BREAK PICKING

The main challenge preventing us from using simple, trace-wise, automatic first break picking approaches (e.g., Coppens, 1985; Sabbione and Velis, 2010) in practice is that they fail in ambiguous situations where the signal-to-noise ratio of the waveforms is poor. Hence, machine learning, due to its capacity to build features from

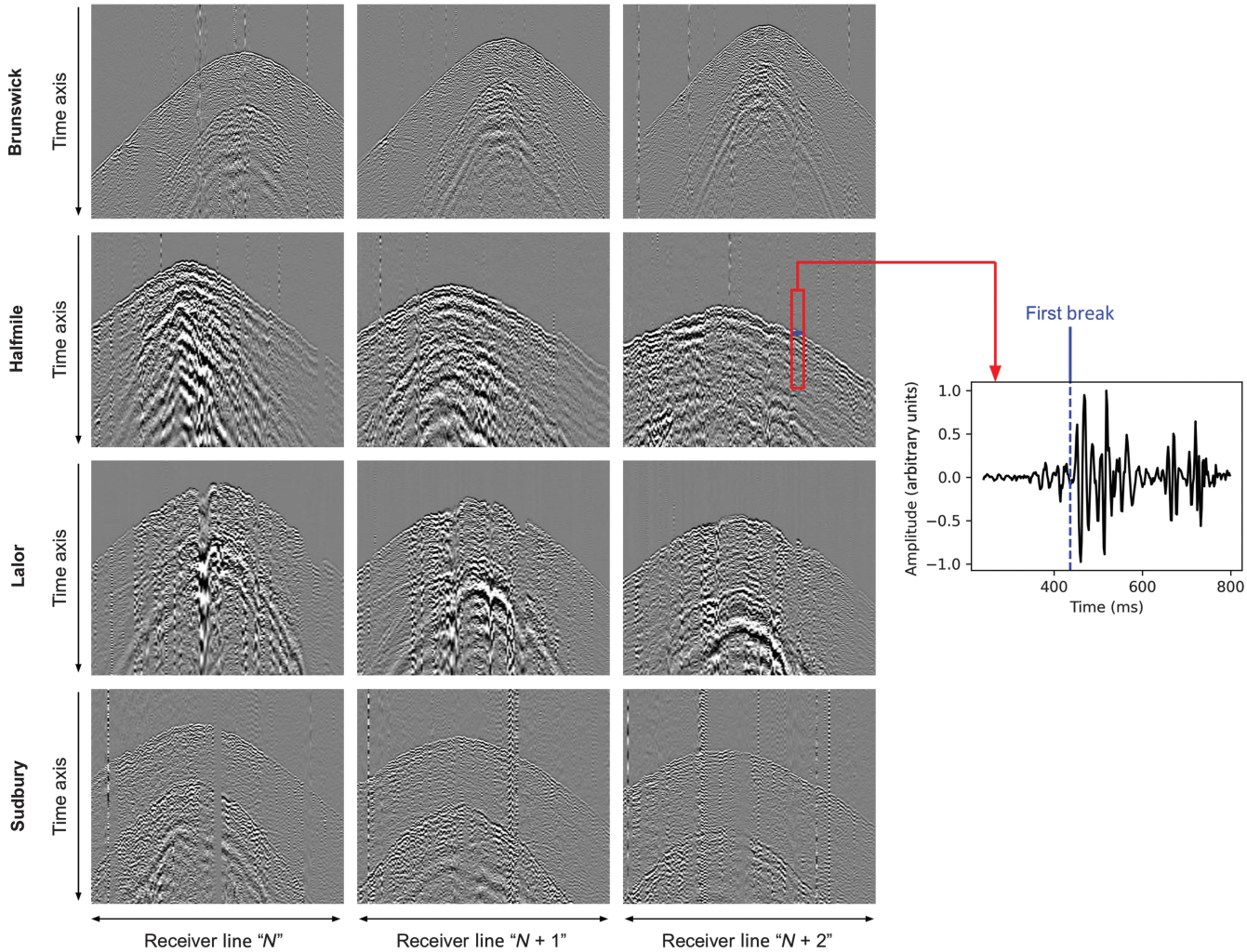

Figure 1. Example of line gathers taken from the Canadian 3D seismic surveys released as part of our benchmark data set. Each image (or line gather) shown corresponds to a receiver line that recorded the same shot from different locations. These images are individually processed in our baseline model. Note that we resized, cropped, and normalized the images to help show seismic patterns for all sites. The raw seismic amplitudes of a single trace (highlighted in red) are plotted on the right with the location of the first arrival that should be picked by predictive models as a blue line.

contextual information, is increasingly applied for first break picking. In particular, deep learning based on neural networks, with minimal needs for adaptation, is suitable to address the first break picking problem. In contrast to "shallow" neural networks that have been used in the past to interpret the combinations of handcrafted features (Dai and MacBeth, 1997; Gentili and Michelini, 2006), deep neural networks are typically applied in a holistic ("end-to-end") fashion. In other words, they can be trained to automatically extract, combine, and interpret task-relevant features with varying levels of complexity. This ability is dependent on the availability of a sufficient amount of training data that can be processed in a structured fashion.

Predictive models can be built from a vast spectrum of neural network architectures that essentially dictate how the input data are encoded into complex features and how the predictions are decoded from these features. A simple encoding approach is to group seismic traces into line gathers and process them as if they were 2D images. With this approach, the gaps or variations in receiver distances would not be reflected in the shape or stride of the array itself, but the seismic amplitude values would still be provided in a 32-bit floating point format. This interpretation of line gathers as images allows for the use of convolutional layers that can exploit the correlations between neighboring traces, a popular strategy in the first break picking literature. For example, CNNs have been specifically shown by Gillfeather-Clark et al. (2021) to be more robust than other encoder architectures for first break picking on the traces of an ore deposit survey. Using larger and deeper CNNs can increase the amount of contextual information available for picking, leading to more accurate results (if sufficient training data are available). The U-Net architecture of Ronneberger et al. (2015) is a good example of an encoder-decoder setup that can predict sample-wise attributes (such as the first break probability) over an entire line gather at once. This particular architecture seems to have been the most popular design choice for first break picking in recent years (Cova et al., 2020; Fernhout et al., 2020; Ma et al., 2020; Yuan et al., 2020; Zheng et al., 2020; Zwartjes et al., 2020; Nikita et al., 2021; Zwartjes and Yoo, 2022). The work of Zwartjes and Yoo (2022) stands out from the rest in terms of protocol quality: they conduct a thorough performance analysis with respect to various hyperparameters, considering fully convolutional, standard U-Net, and U-Net with ResNet block architectures. The neural networks are trained with a two-class mask setup (pre and postfirst break) and the mean absolute error of their first break pick predictions is reported. The authors leverage four land seismic data sets: namely the Teapot Dome 3D, the Stratton 3D, the BP 2D synthetic, and an in-house data set. These data sets do not have human expert-level labels; those labels were instead generated using an automated approach. The authors conclude that the standard U-Net architecture performs best; however, the various site data sets were randomly mixed to train and evaluate, so it is unclear what the generalization performance would be to an unseen new site.

Most published approaches for automatic first break picking rely on image processing architectures directly, and few researchers have worked on adapting these architectures to process geophysical data. The work of Yuan et al. (2020) is one example of the latter: they combined their U-Net architecture with recurrent neural blocks to improve the pick accuracy of the model by integrating features over sequences of concatenated line gathers. This can be seen as a first step toward the design of model architectures made explicitly for

Table 1. Basic information and line gather counts for the sites of interest.

Site name	Trace sampling rate (ms)	Samples per trace	Total line gathers	Valid line gathers (and percentage)	Rejected line gathers	Useful line gathers
Sudbury	2	1001	11,420	5106 (44.7%)	463	4643
Halfmile	2	751	5520	5497 (99.6%)	6	5491
Kevitsa	1	1001	23,111	22,770 (98.5%)	2271	20,499
Brunswick	2	751	18,475	18,457 (99.9%)	14	18,443
Lalor	1	1501	14,455	12,119 (83.8%)	79	12,040

An entire line gather is deemed invalid if none of its traces has a valid first break pick (i.e., within image bounds and at a nonzero sample index). Valid line gathers were manually inspected and a number were flagged for rejection based on the poor quality of their annotations; the useful count is the number of valid line gathers minus the number of rejected line gathers. Although a large fraction of line gathers are valid for most sites, Sudbury stands out with more than half of its line gathers being invalid.

Table 2. Total number of unique shots, receiver lines, and traces for the set of all useful line gathers across our surveys.

	Total count over all useful line gathers			Average count per useful line gather		
Site name	Shots	Receiver lines	Traces	Invalid picks	Dead traces	Traces
Sudbury	777	12	762,506	118.2	0.0	164.2
Halfmile	690	8	1,093,842	18.4	2.7	199.2
Kevitsa	2798	24	1,862,240	13.3	0.2	90.8
Brunswick	1541	28	4,490,714	41.2	0.0	243.5
Lalor	905	16	2,027,587	75.5	0.0	168.4

The average counts of invalid picks (i.e., picks with out-of-bounds sample indices), dead traces (i.e., traces without any observable seismic signal due to sporadic acquisition problems), and traces per useful line gather for the various sites are also provided.

seismic data processing, which is a practice we want to encourage. However, their recurrent blocks had little information to understand the first break time discontinuities between receiver lines, meaning that improvements are still warranted. In addition, more generally, our review and understanding of the recent literature highlights the prominent position of 2D image processing approaches for automated first break picking. In conjunction with the difficulty of developing a generic solution for the processing of 3D seismic data across different sites due to variations in line gather sizes and resolutions, this entails that a 2D CNN-based first break picking model can be considered today as a baseline solution. Future works should aim to improve upon the performance of such a baseline under similar training and evaluation settings; we provide some directions for improvements in our discussion section. However, we must stress that there is no evidence that non-CNN approaches (such as those based on transformers, as proposed by Harsuko and Alkhalifah, 2022) cannot compete with CNNs. More generally, we hope that the introduction of our multisurvey data set will allow new types of models to be designed and trained, and these new models will exploit the full potential of geophysical data.

We now highlight two fundamental issues found in previously proposed evaluation methodologies for automated first break picking solutions. The first issue pertains to the choice of performance evaluation metrics. As noted previously, many researchers now opt for the image processing approach to first break picking where sample-wise (or pixel-wise) first break class probability maps are produced for an entire line gather at a time. This is not problematic per se, but it creates a gap between the predictions of the trained models and the definition of first break picking, which is a regression task, namely the prediction of a continuous quantity (as opposed to a classification task, in which a selection must be made over a limited number of possible choices). Specifically, in first break picking, a single and unique first break should be located in each trace, and temporal location errors can be quantified using, for example, the absolute difference between the annotated and predicted first break locations. Converting predicted sample-wise first break probabilities into trace-wise temporal locations is a postprocessing step that can have a significant impact on model performance. As a consequence, many researchers (e.g., Xie et al., 2019; Cova et al., 2020; Yuan et al., 2020, 2022) have opted to evaluate model performance using classification metrics such as pixel-wise accuracy. Given the imbalanced nature of the pixel-wise first break classification problem, the accuracy is a poor choice as it makes models seem better than they actually are. For example, given a line gather of traces with $N = 1000$ samples each, a model that always predicts "not first break" for all samples would obtain 99.9% accuracy. This problem underlines the need to focus on regression metrics based on temporal location errors instead of pixel-wise classification metrics for the evaluation of predictive models.

The second and more widespread evaluation issue pertains to the potential generalization of predictive models across different sites or surveys. Deep neural networks can perform well when predicting on data that is "in-distribution" (i.e., similar in terms of features and annotations) with respect to their training data set. In contrast, evaluating out-of-distribution data may result in subpar prediction quality, as what these models learn does not always generalize well across different environments or acquisition conditions. Thus, evaluation protocols need to be carefully designed so that performance indicators truly reflect what would happen in realistic cross-site application scenarios. In the case of first break picking, predictive models would likely be used to assist experts in the annotation process of new survey data. This means that an ideal evaluation protocol should always rely on a separate, never-seen-before testing site; this is however rarely the approach used in the literature. As we discuss in the next section, our proposed benchmarking methodology addresses these issues.

METHODOLOGY

In this section, we detail our proposed methodology for the preparation and separation of the survey site data, for the evaluation of first break picking performance, and for the development of a solid baseline model for future comparisons.

Data preparation and separation

The first step in the preparation of the supervised training of a machine learning model is to determine the format of examples

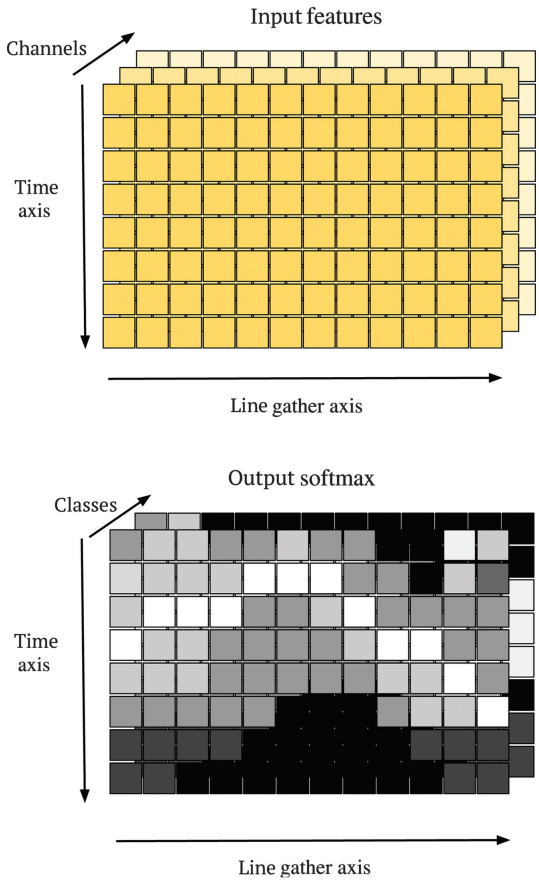

Figure 2. Schematic representation of the proposed baseline model inputs and outputs, which are image-like tensors. The vertical axis corresponds to time and the horizontal axis corresponds to positions along a line gather. The inputs have various channels for the seismic amplitudes and geospatial information and the outputs are pixel-wise probabilities over several predicted classes. Because our proposed model architecture is fully convolutional, it can ingest images of variable sizes. In other words, our baseline approach will work on receiver lines and traces of varying lengths, even after training.

(i.e., pairs of provided inputs and expected outputs or targets) that will be used. A schematic view of our proposed baseline model's inputs and outputs is shown in Figure 2. The input to the model is a line gather, which is treated as an image where samples from receivers on the same line are considered neighboring pixels in the image. Note that using line gathers instead of shot gathers allows us to use well-known 2D architectures that are fully convolutional for our baseline. Note also that we keep the 32-bit precision of trace data intact during processing, as all inputs to our models are provided as arrays of floating point numbers.

Our input images also contain additional channels that are used to provide trace-wise geospatial cues to the model along with the concatenated trace samples. We create three extra image channels in total for each line gather. Concretely, these encode the distance between the shot location and each receiver as well as the two distances between each receiver and its two closest neighbors on the same receiver line. All three distance channels are constant along the time axis. We rescale the shot-to-receiver distances by dividing them by 3000 m; similarly, we rescale the receiver-to-receiver distances by dividing them by 50 m. The 3000 and 50 m normalizing factors were chosen empirically to produce input feature values close to the $[0, 1]$ interval. As for the trace data itself, we normalize the amplitudes to the $[-1, 1]$ range by dividing them by the maximum absolute amplitude found in each trace. Normalizing the input feature values to these ranges is a commonly used strategy to improve training speed and numerical stability.

For the model output, we use pixel-wise class probability maps that have the same temporal and line gather axes as the input. For training and evaluation, we generate the target class label maps that the model should predict by transforming the trace-wise first break annotations into pixel-wise indices. We consider two class setups that seem to work relatively well in practice: a binary and a ternary setup. For each trace, the single pixel that matches the location of the first break is set to the class "first break." In the binary class setup, all other pixels on the trace are assigned the class "not first break." In the ternary class setup, the pixels that correspond to the times before the first break are assigned the class "before," and the pixels corresponding to the times after the first break are assigned the class "after." This latter setup relies on two auxiliary classes: although the before and after classes are of no real use to downstream applications, they can still help the model understand and locate the first break class itself more accurately. This setup might also help mitigate the issue of class imbalance caused by the underrepresentation of the first break class. The predictive model is designed to output a probability distribution over these two or three classes for each pixel using a *softmax* function (see Bridle, 1990). The softmax function σ takes n real numbers $\mathbf{x} = (x_i, .., x_n)$ as an input and returns a probability distribution over n possibilities; explicitly, $\sigma(\mathbf{x}) = (\sigma(\mathbf{x})_i, .., \sigma(\mathbf{x})_n)$, where $\sigma(\mathbf{x})_i = (\exp(x_i)/\sum_{j=1}^{n} \exp(x_j))$. For each trace, the pixel with the largest first break probability is retained as the model's prediction for the first break pick.

As for data splitting, the main driver for our experiments is to determine whether trained models can generalize their knowledge across survey sites. To reach such a conclusion, we manually split the sites at our disposal into different cross-validation folds. As shown in Table 3, we first consider training sets composed of three sites and use one site each for the validation set and the test set. To limit the computational requirements of our experiments, we only kept five combinations with all sites (folds A–E) by requiring that each site appears once in the validation set and once in the test set. Given that the Kevitsa site cannot be released for public use due to licensing concerns, we also define folds without the Kevitsa with only two sites in the training set (folds H–K) for future comparisons by other researchers.

Evaluation metrics

For the performance metrics, we consider the mean average error (MAE), the mean bias error (MBE), and the root mean square error (RMSE). These are defined as

$$\mathrm{MAE} = \frac{1}{N}\sum_{i=1}^{N} |\hat{t}_i - t_i|, \tag{1}$$

$$\mathrm{MBE} = \frac{1}{N}\sum_{i=1}^{N} (\hat{t}_i - t_i), \tag{2}$$

$$\mathrm{RMSE} = \sqrt{\frac{\sum_{i=1}^{N} (\hat{t}_i - t_i)^2}{N}}, \tag{3}$$

where N is the number of annotated traces, t_i is the ground truth value of the first break pick for the ith trace, and $\hat{t}_i$ is the corresponding model prediction. The picks are located in terms of sample indices, i.e., on a pixel scale; correspondingly, all errors are measured on the same scale, namely in terms of pixel distances in the line gather images. The MAE, MBE, and RMSE are standard metrics commonly used to evaluate regression models. The MAE provides an overall measure that is less sensitive to outliers, the MBE provides insight into whether the first break is predicted early or late, and the RMSE provides an overall measure with a larger penalty for outliers.

We also rely on the hit rate at δ pixels (HR@δpx), defined as the fraction of annotated traces where the prediction error is smaller than δ pixels, namely

Table 3. Proposed site folds for experiments with the full data set (A–E) and for future comparisons without Kevitsa (H–K).

Data set	Fold A	Fold B	Fold C	Fold D	Fold E	Fold H	Fold I	Fold J	Fold K
Train	Lalor	Kevitsa	Halfmile	Sudbury	Brunswick	Halfmile	Sudbury	Lalor	Brunswick
	Brunswick	Lalor	Kevitsa	Halfmile	Sudbury	Lalor	Halfmile	Brunswick	Sudbury
	Sudbury	Brunswick	Lalor	Kevitsa	Halfmile				
Validation	Halfmile	Sudbury	Brunswick	Lalor	Kevitsa	Brunswick	Lalor	Sudbury	Halfmile
Test	Kevitsa	Halfmile	Sudbury	Brunswick	Lalor	Sudbury	Brunswick	Halfmile	Lalor

$$\mathrm{HR@}\delta\mathrm{px} = \frac{1}{N}\sum_{i=1}^{N} h(\hat{t}_i, t_i, \delta), \tag{4}$$

where $h(\cdot)$ is a thresholding function defined as

$$h(\hat{t}_i, t_i, \delta) = \begin{cases} 1 & \text{if } |\hat{t}_i - t_i| < \delta \\ 0 & \text{otherwise} \end{cases}. \tag{5}$$

Thus, the parameter δ controls how stringent the corresponding metric is: a choice of $\delta = 1$ penalizes any prediction that is not exactly correct, whereas larger values of δ are more lenient. In this work, we consider values of δ between 1 and 9.

Finally, we define trace coverage (TC) as the fraction of annotated traces where the model makes a prediction. This metric may be useful for applications in which a minimum prediction confidence is required, as the model might sometimes refrain from predicting anything on ambiguous traces. The TC is equivalent to HR@δpx in the limit where δ becomes very large. Note that we do not consider a confidence threshold in our own predictive models, consequently, our models always make a prediction and the TC is essentially 100%. As we will detail subsequently, the automated pick baselines have TC scores that differ from 100%.

For a given data set fold, we conduct a random hyperparameter search over 50 trials. The hyperparameters are configuration variables that have an impact on the model architecture or its training process. The definition of these hyperparameters and their potential values are provided in the next section. For each trial, we train a model from a random initialization on the training sites and evaluate it on the validation site. After 50 trials, the hyperparameter configuration leading to the model with the highest validation HR@1px score is selected, and 10 models are trained once again with this configuration, starting from different random seeds. These 10 models are finally evaluated on the test site to give a range of performance.

Baseline model description

For our baseline, we consider the U-Net architecture where the encoder and decoder are composed of stacks of fully convolutional blocks. This architecture is shown in Figure 3. For the encoder, we evaluate ResNet blocks (He et al., 2016) with two different depths (18 and 34 total layers) and EfficientNet blocks with b0, b2, and b4 configurations (defined by Tan and Le, 2019); these are all CNNs with increasing levels of complexity and learnable parameters. For the decoder, we use blocks composed of stacked 3×3 convolutions, batch normalization (Ioffe and Szegedy, 2015), and rectified linear unit activation layers. Normalization layers (such as batch normalization) are commonly used in CNNs to stabilize intermediate features during training. We explore three different levels of complexity for the decoder blocks by scaling the number of feature maps carried over from the encoder. Specifically, we use [256, 128, 64, 32, 16], [512, 256, 128, 64, 32], or [1024, 512, 256, 128, 64] feature maps, where each number corresponds to the input depth for each of the five decoder blocks. Note that across all potential hyperparameter configurations, our smallest model has approximately 14M trainable parameters, whereas the biggest model has 49M trainable parameters. The entire list of hyperparameters and their ranges are shown in Table 4.

We use four data augmentation operations during training to increase the perceived diversity of the data set and help avoid overfitting. First, line gather images are randomly cropped so that they have between 512 and 1024 samples on their time axis. Second, traces inside each gather are randomly dropped (as if there were a gap in the receiver line) or added with null amplitudes (as if an extra receiver were present but "dead") to reach a new line gather axis length that is randomly selected in [64, 128, 256, 512]. These operations allow us to generate a wide range of image formats even when training on sites with relatively consistent line gather lengths. Note that when needed, distances used in the extra input channels for added receivers are interpolated from the data of the nearest (real) receivers. Third, we nullify the amplitudes (i.e., we replace all sample values with zero) of approximately 8% of all the traces in the gathers. This forces the model to rely on more contextual information. Finally, we flip the gather images along their receiver axis to increase the (perceived) diversity of the data sets. Other augmentation operations were originally considered, but their impact was subsequently found to be marginal. The parameter values for the selected operations were found empirically on a small subset of the available data before conducting the large-scale hyperparameter search.

Given our binary or ternary class setup, we define a loss function to train our models based on one of two possible approaches. The first approach relies on the standard pixel-wise cross-entropy (CE) loss. The CE measures how different two probability distributions are; specifically, for $\{p_1, \ldots, p_n\}$ and $\{q_1, \ldots, q_n\}$, respectively, the target and estimated probability distributions over n discrete events, the CE is given by $\mathrm{CE}(p, q) = -\sum_{i=1}^{n} p_i \ln q_i$, which is often used in the image segmentation literature (Ronneberger

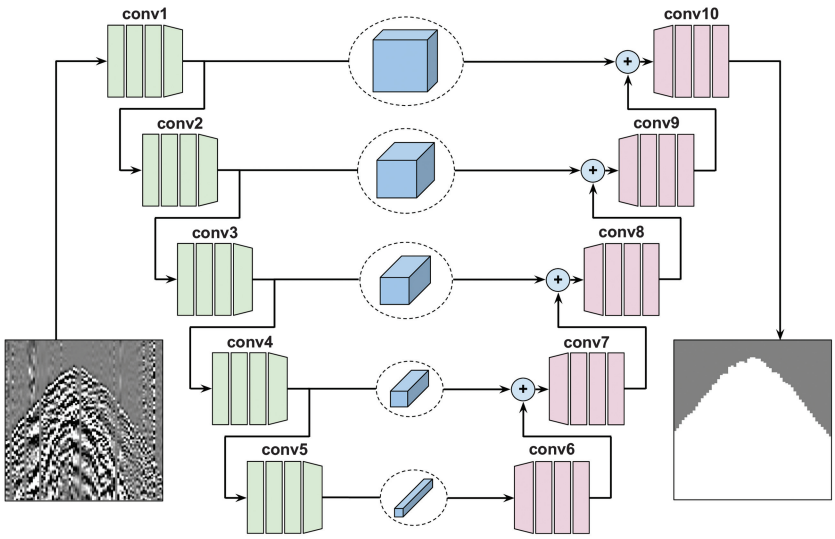

Figure 3. Illustration of the U-Net model architecture we use for first break picking. The conv1–conv5 blocks form the encoder and conv6–conv10 blocks form the decoder. The exact composition of each convolutional block can vary by application. The skip connections are shown in the middle and allow the concatenation of encoder and decoder feature maps.

et al., 2015). The most common criticism of this loss for semantic segmentation is that it does not properly handle imbalanced classes. Because this is the case in our application, we also experiment with a second loss based on the Dice coefficient (the Dice coefficient is a continuous version of the $F1$ score, which can thus be used as a loss). Specifically, for $\{p_1, \ldots, p_n\}$ and $\{q_1, \ldots, q_n\}$, two probability distributions over n discrete events, the Dice coefficient is given by

$$\mathrm{DC}(p,q) = 1 - \left(\left(\sum_{i=1}^{n} p_i q_i + \epsilon\right) \Big/ \sum_{i=1}^{n} p_i + q_i + \epsilon\right) - \left(\left(\sum_{i=1}^{n} (1-p_i)(1-q_i) + \epsilon\right) \Big/ \sum_{i=1}^{n} 2 - p_i - q_i + \epsilon\right), \quad (6)$$

where ε is a small value used to ensure numerical stability (Sudre et al., 2017). An interesting property of the Dice loss is that it results in very sharp boundaries in the predicted class probability maps. Although this might seem beneficial in some cases, it also means that gradients will not be as smooth as when using the CE loss, which may impact the training process.

For the optimization, we rely on the Adam optimizer. The optimizer is the algorithm that updates the weights of a neural network during training based on the gradient of the loss function with respect to these weights. The Adam optimizer is a standard approach that uses a smoothed version of the gradients to accelerate convergence (Kingma and Ba, 2014). We pick the base learning rates uniformly across a logarithmic scale of $[10^{-5}, 5 \times 10^{-3}]$. We train for a maximum of 20 epochs and either never modify the learning rate or reduce it by multiplying it with a factor of 0.1 after either 5 or 10 epochs. These hyperparameters were selected empirically following preliminary experiments: the maximum number of epochs seems to correspond to the typical duration it took for our models to converge with three training sites. The learning rate is commonly reduced during training in most deep learning experiments to help the model converge more effectively toward a minimum in the loss function. This prevents oscillations around the optimal solution and prompts better model generalization.

The batch size is fixed for all experiments at 16 line gathers. We use a custom collate function for batching to ensure that all gathers are padded to a common resolution that is also a power of two for compatibility with the U-Net's default decoder architecture. Because the overall architecture of our models is fully convolutional, the actual dimensions of the input gathers following this padding should not have an impact on the quality of predictions, as long as the padded gathers are sufficiently large. Early stopping is performed if the HR@1px on the validation site does not improve for more than four consecutive epochs.

To provide a sense of scale for the performance of our baseline model, we also apply a classic picking algorithm across all sites. The picks were generated automatically by sequentially combining the three following methods: linear moveout based on a constant velocity, autoregressive Akaike-information-criterion (AR-AIC) picking that focuses only on P waves (Sleeman and Van Eck, 1999), and fine-tuning search. The linear moveout makes a coarse approximation of the first arrivals based on a 6 km/s moveout velocity. Those coarse approximations are then used as input to the AR-AIC method to separate the seismic traces into two intervals, each fitted with an autoregressive function. The position of the first break is determined by finding the time that best separates the seismic trace into noise (nondeterministic) and signal (deterministic) components. This is accomplished by finding the intervals that provide the lowest order of the variance not explained by the autoregressive models and estimated with the Akaike information criterion (Akaike, 1974). Finally, the final and most precise picks are fine tuned by finding the nearest trough within a 10 ms window of the AR-AIC pick. The same parameters were used for all data sets. Trace amplitude balancing was applied to all data sets prior to autopicking. The picks were then subjected to a noise-based filtering process. The root mean square (RMS) amplitude of the trace was computed in a 30 ms window before (RMS_1) and after (RMS_2) the pick, and the ratio $\mathrm{RMS}_1/\mathrm{RMS}_2$ was then compared with a user-specified threshold value. If the ratio was larger than the threshold,

Table 4. Main hyperparameters and their possible values or ranges.

Hyperparameter	Possible values	Description
Maximum epoch	20	Maximum number of epochs a training session can run for.
Patience	4	Maximum number of consecutive epochs without improvement before stopping a training session.
Encoder type	ResNet18, ResNet34, EffNetB0, EffNetB2, and EffNetB4	Configuration of the encoder blocks.
Decoder blocks	[256, 128, 64, 32, 16], [512, 256, 128, 64, 32], and [1024, 512, 256, 128, 64]	Number of feature channels for all decoder blocks.
Class setup	Binary and ternary	Number of classes to predict; either first break or not, or before/after/first break.
Optimizer	Adam (Kingma and Ba, 2014)	Optimizer used to update the model's weights.
Learning rate	LogUniform $[10^{-5}, 5 \times 10^{-3}]$	Controls the magnitude of the weight updates.
Weight decay	10^{-6}	Magnitude of the penalty added to the loss to keep weights as small as possible.
Loss type	CE and Dice	Objective function used for backpropagation.
Scheduler step	5, 10, and ∞	Number of epochs before the learning rate is multiplied by a factor of 0.1.

When a single value is present, the hyperparameter is fixed to that value.

then the pick was rejected; thus, the autopicker did not necessarily generate picks for all traces. A small threshold is more conservative and refrains from producing a pick when the noise level is high; a higher threshold value is more permissive with respect to the noise level. Each site is evaluated for two values of the noise level threshold parameter: 0.3 and 1.0. We note that, in practice, such automatic picks are considered an intermediate product that needs editing and validation and would not be used directly: correspondingly, we do not consider the comparison of our baseline models with the autopicker to be a robust performance indicator.

EXPERIMENTAL RESULTS

The performance of the automated picking algorithm (i.e., the sequential approach described previously) is presented in Table 5. For every site, the corresponding hit rates (HRs) are higher when the threshold is set to 1.0 but are still somewhat poor, reaching the highest HR@1px of 77.2% on the Halfmile site.

An overview of the results of the random searches of the hyperparameters related to the U-Net model is shown in Figure 4. As Figure 4 makes it clear, many trained models have HR@1px values in a narrow range of the nominal "best." Each trial can be considered as a point estimate of expected performance for its corresponding set of hyperparameters, and as such we cannot make statements about the statistical significance of choosing one model over another at the highest levels of performance. Making such robust statistical statements would require a large computational budget and would provide diminishing value at this stage; thus, we adopt the pragmatic approach of selecting a good model without claiming it is necessarily the best in a statistical sense.

We present the hyperparameter combinations of the top-performing models on the validation set of each fold in Table 6 and the corresponding metrics in Table 7. We can first observe that the HR@1px performance on folds A, B, C and D is fairly high, while it is low for fold E. This latter fold is validated on the Kevitsa site, which suffers from less consistent annotations and uses a different source type (i.e., hydraulic hammers) (for more information, see Valasti et al., 2012). Furthermore, fold E is the only full data set fold where we train on sites with 2 ms sampling rates but validate on a site that had mixed sampling rates subsequently resampled to 1 ms; all the other folds train on combinations of sites that exhibit both sampling rates as can be deduced from Tables 1 and 3. Folds A and C have the best performance (mid to high 80s percentage), whereas folds B and D lag behind with performance in the mid 70s percentage. This correlates with the putative quality of the ground truth picks, where Halfmile and Brunswick were annotated by careful experts, whereas the annotations at Sudbury are of lower quality, and those at Lalor followed the incorrect convention (onset instead of trough) and were corrected using an automated tool (a pragmatic but imperfect solution). Other factors that might explain the performance variations are detailed in the next paragraph. We further note that HR performance increases sharply as

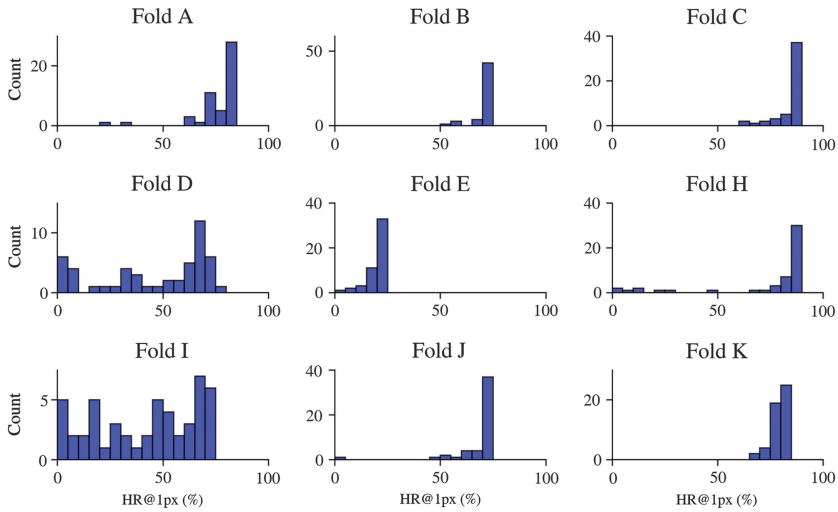

Figure 4. Distribution of HRs at one pixel (HR@1px) over the validation set for all folds, based on 50 random hyperparameter choices per fold.

Table 5. Various metrics for the *baseline autopicks* method.

		HR@								
Site	Th.	1px	3px	5px	7px	9px	TC	RMSE	MAE	MBE
Sudbury	0.3	65.1	82.7	84.3	85.1	85.5	86.1	42.9	15.5	−15.0
	1.0	68.9	87.8	90.4	92.4	93.5	96.2	22.8	5.1	−3.8
Lalor	0.3	47.6	48.2	51.4	58.1	63.1	66.3	152.5	82.3	−80.0
	1.0	51.7	52.4	58.3	75.4	89.2	95.0	61.1	15.4	−9.8
Brunswick	0.3	29.6	38.1	46.6	62.6	70.6	83.0	132.2	53.7	−50.3
	1.0	31.0	40.8	50.4	68.0	77.3	94.7	74.3	20.3	−15.6
Kevitsa	0.3	16.5	38.4	53.1	59.7	62.0	69.0	75.9	42.1	−38.5
	1.0	20.6	46.8	64.7	73.3	76.5	90.7	41.8	16.2	−9.4
Halfmile	0.3	71.4	75.6	77.6	79.3	80.5	82.3	97.0	39.4	−39.1
	1.0	77.2	82.4	85.1	87.6	89.6	96.0	47.7	10.7	−9.7

The noise threshold is represented by "Th." The HR and TC are in percentage and the errors (RMSE, MAE, and MBE) are in the number of samples.

the threshold varies from 1 to 9 pixels. Already at three pixels, the HR is above 90% on folds A, B, and C. Approximately a 20% improvement from HR@1px to HR@3px on Sudbury is especially notable: a brief inspection of the predictions suggests that this may be caused by some ground truth annotations being slightly off trough, such that the predictions made on the trough are penalized in terms of HR@1px but quickly contribute to HRs at larger thresholds. The performance eventually reaches approximately 99% for folds A, B, and C, 93% for fold D, and approximately 90% for fold E at nine pixels. This indicates that trained models can place first break picks in the correct region most of the time, but either struggle to locate the pick precisely in ambiguous situations or are evaluated against incorrect/inconsistent annotations. Finally, we highlight that the trained models perform much better than the autopick baseline, from approximately 3% HR@1px improvement on the Kevitsa site to approximately 58% on the Brunswick site. However, we stress once again that the autopick baseline is not strong: it is meant to reflect how well a traditional approach fares when applied to different sites without hyperparameter adjustments.

Apart from variations in the quality of annotations, other factors can explain the lagging validation performance on some sites. For Lalor, one factor is the frequency content of seismic traces below 5 Hz and above 200 Hz due to the use of MEMS and 1 ms sampling rate, which makes this site quite different compared with the other sites; this is shown in Figure 5. As shown in Table 7, the RMSE, MAE, and MBE metrics are much larger when a model is evaluated on Lalor; the error distribution for the model corresponding to fold D is shown in Figure 6. We can see that there is a substantial number of predictions resulting in large errors. We investigated and found that these bad predictions often correspond to noisy traces that are common on this site, to annotation errors, or to the model being confused by reflected waves that do not correspond to the first break; examples are shown in Figure 7. Some of these bad predictions also exceed the number of samples per trace (1501 for Lalor), as the model sometimes predicts the first breaks inside the image's zero-padding zone (because our architecture requires power-of-two dimensions, we use 2048 time samples per seismic trace for Lalor, with samples beyond 1501 set to a value of zero). This misbehavior is likely due to the out-of-distribution nature of Lalor with respect to other sites. If we discard the predictions made beyond the real sample range, the TC for fold D falls from 100% to 98.6% and the regression errors become 10.4 pixels for the RMSE, 1.6 pixels for the MAE, and 0.7 pixels for the MBE; these values are much closer to the range of values from the other sites. As shown in Figure 8, the error distribution for the model of fold C applied to the Brunswick site also

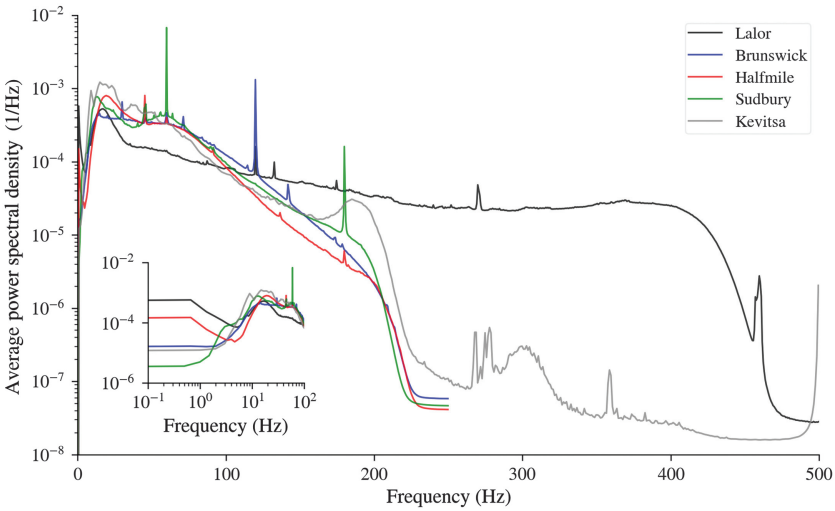

Figure 5. Averaged power spectral densities of raw field seismic signals for all sites (the inset plot shows the low-frequency part of the average power spectral densities). This is obtained for each site by averaging the power spectral densities of all the normalized traces of the useful line gathers, as defined in Table 1. As discussed in the "Data preparation and separation" section, the seismic amplitudes are normalized by dividing them by the maximum absolute amplitude found in each trace so that the resulting normalized amplitudes lie in the $[-1, 1]$ range. As evidenced in the inset, Lalor has a large average power density peak below 5 Hz, probably due to the use of MEMS. In addition, although the average power density drops after 200 Hz for most sites, we can see that Lalor still has significant spectral weight at higher frequencies due to the 1 ms sampling rate combined with the broad spectral signature of explosives. The Kevitsa data, although acquired with mixed sampling rates, have a power spectral density similar to the other 2 ms data sets (Brunswick, Halfmile, and Sudbury).

Table 6. Best hyperparameter configurations found for all folds, after 50 trials, in terms of HR@1px on each fold's validation site.

Fold	Encoder type	Decoder blocks	Loss type	Learning rate	Scheduler step	Class setup
A	ResNet18	[256, 128, 64, 32, 16]	CE	0.002136	10	Binary
B	EffNetB4	[512, 256, 128, 64, 32]	CE	0.002708	10	Binary
C	EffNetB0	[512, 256, 128, 64, 32]	CE	0.003417	5	Binary
D	EffNetB4	[512, 256, 128, 64, 32]	CE	0.002580	5	Binary
E	EffNetB0	[1024, 512, 256, 128, 64]	CE	0.000295	10	Binary
H	EffNetB4	[512, 256, 128, 64, 32]	CE	0.003480	5	Binary
I	EffNetB0	[512, 256, 128, 64, 32]	CE	0.002453	10	Binary
J	EffNetB4	[1024, 512, 256, 128, 64]	CE	0.002762	5	Binary
K	EffNetB4	[512, 256, 128, 64, 32]	CE	0.001053	10	Binary

shows predictions made inside the zero-padding zone with low probabilities, leading to larger RMSE, MAE, and MBE values. If again we discard these predictions beyond the real sample range, the TC for fold C falls from 100% to 99% and the regression errors become 5.1 pixels for the RMSE, 0.4 pixels for the MAE, and 0.1 pixels for the MBE. Furthermore, some high-probability predictions that lead to large errors that are not in the padding are shown in Figure 8. These occur when the model is confused by late, high-intensity samples recorded by the receivers closest to the source stemming from explosion-induced air waves; examples are shown in Figure 9. For other sites, predictions in the padding affect less than approximately 0.2% of the traces, not warranting further discussion.

The impact of Kevitsa in training can be gauged by comparing the performance of the folds provided for future comparisons (H–K) with their parent full-data set folds A–E. Fold H (parent fold C), with an HR@1px of 89.6% (89.4%), and fold J (parent fold B), with an HR@1px of 74.3% (74.2%), see their performance marginally increase by the removal of Kevitsa, suggesting that this latter site may not always provide a useful signal during training. In contrast, fold I (parent fold D), with an HR@1px of 72.4% (75.4%), sees its performance reduced by a nonnegligible 3% with respect to its parent. One possible explanation for this drop in performance is the fact that in this case, similar to fold E, we are training on sites with 2 ms sampling rates, whereas we are validating on a site with a 1 ms sampling rate and that used MEMS as receivers.

Next, we report the performance of our best hyperparameter configurations on the withheld test set of each fold in Table 8. As a reminder, we highlight that these final scores were computed 10 times with different random seeds for each fold, and only after all hyperparameters were fixed using the validation set. We can observe that the results seem to be mostly similar to those of Table 7 when considering only the evaluation site instead of the actual fold. This indicates that our best models are relatively robust to the different training set configurations we used and that the reported performance of a model is tied closely to how different the test site is from the training (and, at least partially, from the validation) sites. The largest spread of performance for a particular site happens with Lalor, with validation scores of 75.4% (fold D) and 72.4% (fold I) and test scores of 71.6% (fold E) and 76.3% (fold K). Many factors may affect this spread in performance: the size of the training set, the quality of the validation site used for model selection, the match or mismatch of sampling rates between training sites and Lalor, and the finiteness of the hyperparameter searches. Although there does not appear to be a simple relationship between these factors and the observed HR@1px, it could be that the underperformance on fold E is most impacted by the strategy that was used to select the best model for this fold: we relied on the Kevitsa site for model selection, a site with a different source type and less consistent annotations. Those elements can lead to models that might be unfit for other data distributions.

Finally, we ran a series of experiments to quantitatively assess the difficulty of generating predictions for all sites which may better

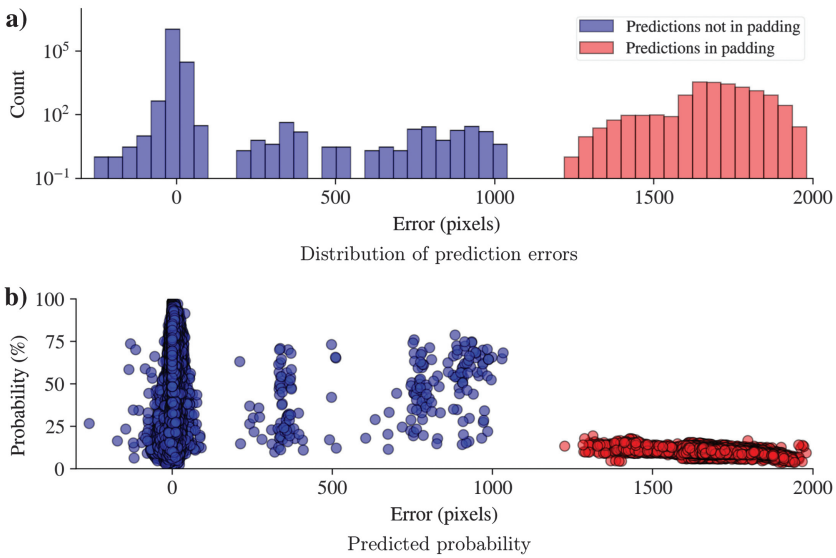

Figure 6. (a) Distribution of prediction errors, i.e., the distance between predicted and annotated first breaks in sample or pixel units and (b) scatter plot of predicted first break probability with respect to prediction errors, using the model reported for fold D (validated on Lalor) in Table 7. (a) Several predictions are in the zero-padding zone and result in large errors. (b) These predictions (drawn in red) have relatively low probability.

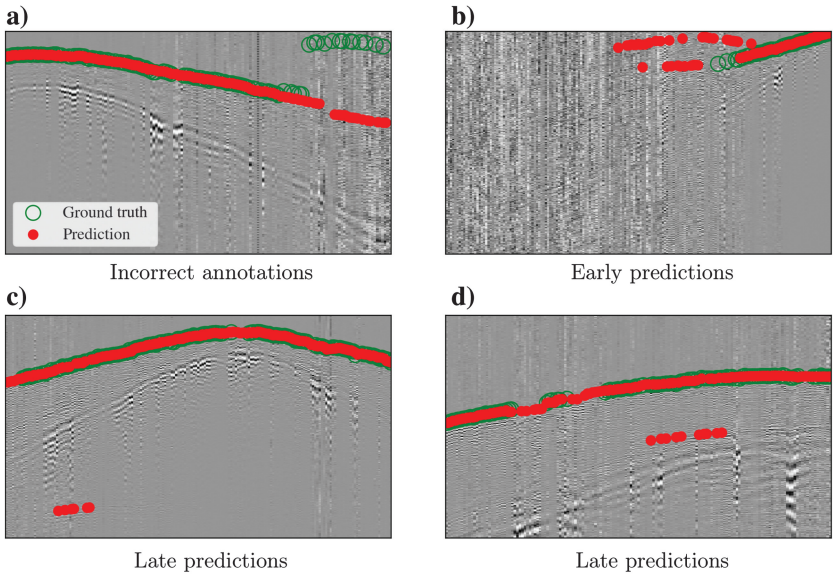

Figure 7. Examples of predicted first break picks with large errors for Lalor using the model reported for fold D in Table 7. (a) An example in which some annotations are manifestly incorrect. (b) An example in which a large amount of noise induces the model to predict a first break pick too early. (c and d) Examples where the model is confused by reflected waves at subsequent times.

explain the differences in performance across folds. In Figure 10, we show the quality of the seismic recordings by computing a ratio that quantifies the amount of noise around the annotated picks in each trace. The ratios are computed using the same approach used for automatic picking, i.e., by comparing RMS amplitudes computed over fixed-size (30 ms) windows on each side of the first break. In short, small RMS ratios indicate that there is a strong contrast between prefirst and postfirst break amplitudes. This should lead to more easily predictable picks, as the exact location of the first break is less ambiguous. Here, we can observe that these ratios are significantly larger in Kevitsa than in the other sites. This supports our original intuition regarding the difficulty of this site based on a visual analysis of its gathers. Next, we run the best validation model of each fold on its corresponding test site and analyze the quality of its predictions from two other angles. In Figure 11, we first show the HR@1px scores computed for all traces at different RMS amplitude ratios. As expected, higher RMS ratios lead to a degradation in prediction performance, but it is interesting to note that low ratios on Kevitsa still lead to subpar performance. This indicates that noise and less consistent picks may account for this performance, although the effect of each is difficult to distinguish as the quality of picks generally degrades with increasing noise for any land seismic data set. Finally, the relation between HR@1px scores and the receiver-shot (offset) distances is shown in Figure 12. Most of these curves confirm the intuition that bigger offsets lead to weaker signals in the recorded traces, and thus harder-to-predict pick locations; however, some sites (Kevitsa, Sudbury) do not support this conclusion. This may be due to the smaller spatial dimensions of their surveys, which translate into smaller maximum offsets.

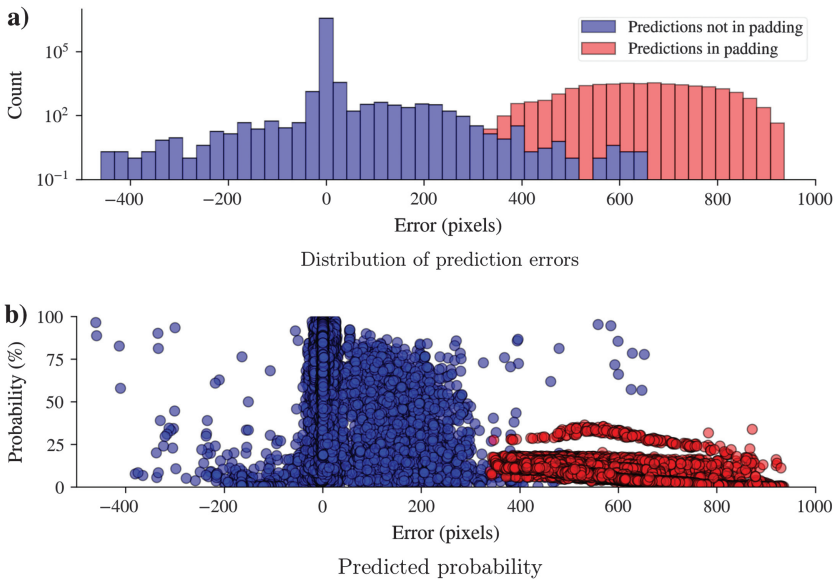

Figure 8. (a) Distribution of prediction errors, i.e., the distance between predicted and annotated first breaks in sample or pixel units and (b) scatter plot of predicted first break probability with respect to prediction errors, using the model reported for fold C (validated on Brunswick) in Table 7. (a) Several predictions are in the zero-padding zone and result in large errors. (b) These predictions (drawn in red) have relatively low probability.

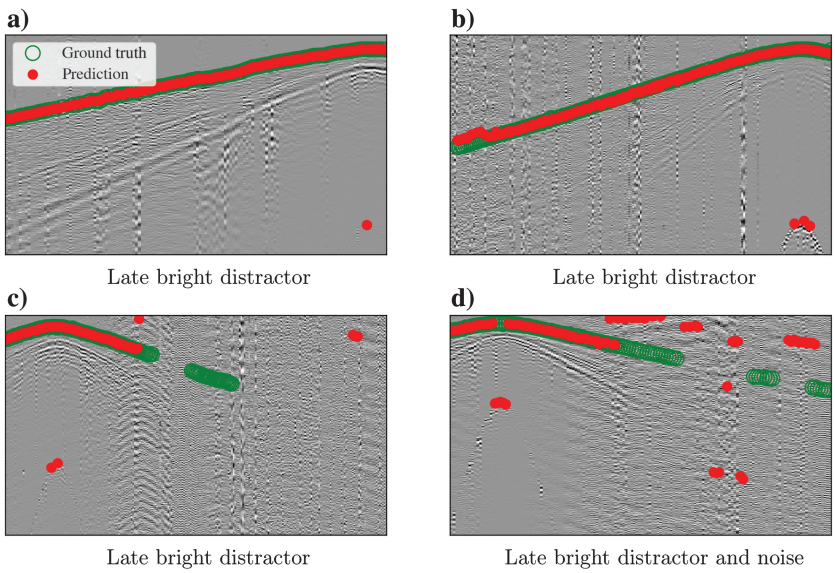

Figure 9. Examples of line gathers with ground truth and predicted first break picks for Brunswick using the model reported for fold C in Table 7. (a), (b) and (c): some predicted picks for receivers closest to the source have large errors because the model is confused by a bright signal at subsequent times. This irrelevant signal is probably caused by an air wave induced by the source explosion. (d) Predictions are also impacted by the presence of coherent noise before the first arrivals.

Remarks for future works

Ultimately, with this study, our hope is to promote the design and development of new models specifically tailored to assist in the picking of first arrivals. In particular, it is unclear whether the interpretation of line gathers as images is ideal. Our proposed baseline ingests such images with extra channels specifically designed to express the geospatial setup of seismic surveys, but this is an indirect way to provide this information that can be ignored by the model during training. There are model architectures designed for graph or point cloud data processing (e.g., Zhang et al., 2019) that could be better suited to leverage spatial information for first break picking. Using model architectures that are not translation invariant (e.g., vision transformers; Dosovitskiy et al., 2020) or removing this property from CNN architectures using coordinate convolutions (Liu et al., 2018) could help models rely more on the geospatial data. However, these approaches may increase the risk of overfitting on smaller surveys, and they may reduce the robustness to variations in line gather sizes or resolution. Exploiting the correlation across all traces recorded for a single shot (i.e., using shot gathers

as examples instead of line gathers) should provide another source of improvement for models. This could be achieved by modifying our current baseline approach to use 3D CNN blocks that are also fully convolutional. These new blocks would allow entire shot gathers to be processed at once without having to worry about variations in terms of the number of line gathers per shot, traces per line gather, or samples per trace. However, such a modification would require a significant amount of padding to be used to process smaller examples of shot gathers or data from 2D surveys, which may be detrimental to the model's performance.

Regarding the computational costs of deep learning approaches, note that an ordinary desktop computer equipped with a modern CPU and GPU can be used to train the models proposed in this study. When considering the largest backbone architecture studied here (with 49M learn-

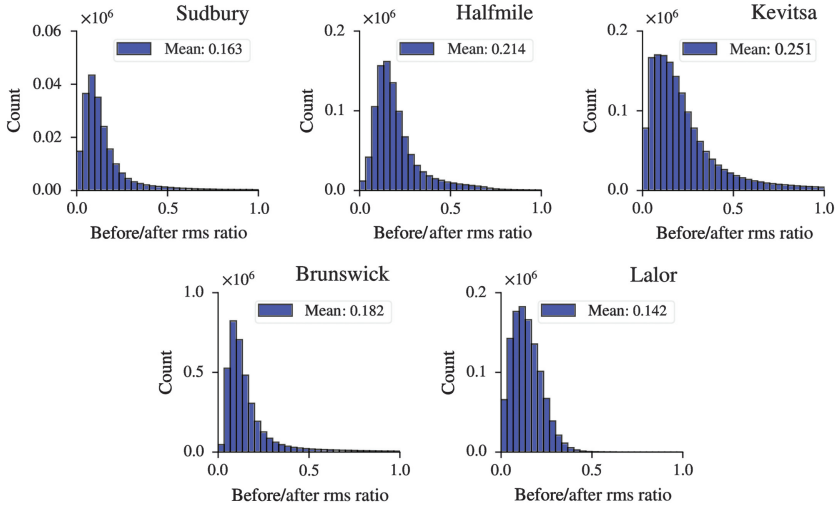

Figure 10. Distribution of the ratios of RMS amplitude values before and after the annotated first break picks for all sites. Smaller ratios mean a more pronounced transition from background noise to relevant seismic events and potentially easier-to-detect first breaks.

Table 7. Performance metrics computed on the *validation site* of each fold using a U-Net model whose hyperparameter configuration resulted in the highest HR@1px score found after 50 trials.

		HR@							
Fold	Site	1px	3px	5px	7px	9px	RMSE	MAE	MBE
A	Halfmile	84.0	92.7	96.1	98.1	99.1	28.0	1.7	1.1
B	Sudbury	74.2	95.4	97.6	98.7	99.2	10.3	0.7	0.5
C	Brunswick	89.4	96.3	97.7	98.2	98.5	65.8	6.8	6.5
D	Lalor	75.4	80.8	84.5	89.5	93.1	200.1	24.8	24.0
E	Kevitsa	23.5	54.7	75.8	85.8	89.6	46.9	6.3	4.8
H	Brunswick	89.6	96.8	98.3	98.8	99.1	43.5	3.1	2.4
I	Lalor	72.4	76.0	79.5	84.3	87.8	453.1	117.3	116.0
J	Sudbury	74.3	95.4	97.7	98.7	99.2	19.2	1.0	0.7
K	Halfmile	83.7	92.8	96.2	98.2	99.1	27.5	1.6	0.9

The validation site over which the metrics are evaluated is restated for convenience in the column "Site." The HRs are in percentage and the errors (RMSE, MAE, and MBE) are in the number of samples.

Table 8. Performance metrics computed on the *test site* of each fold averaged using 10 U-Net models.

		HR@							
Fold	Site	1px	3px	5px	7px	9px	RMSE	MAE	MBE
A	Kevita	22.4 ± 0.6	53.3 ± 1.0	74.3 ± 1.3	84.3 ± 1.4	87.9 ± 1.4	61.7 ± 39.5	10.6 ± 8.6	7.6 ± 9.1
B	Halfmile	82.5 ± 0.9	92.6 ± 0.2	96.0 ± 0.3	98.1 ± 0.3	99.0 ± 0.3	12.9 ± 4.6	1.1 ± 0.6	0.3 ± 0.5
C	Sudbury	73.4 ± 0.6	94.0 ± 0.6	96.2 ± 0.6	97.6 ± 0.5	98.4 ± 0.5	18.8 ± 7.7	1.5 ± 0.6	0.9 ± 0.6
D	Brunswick	87.7 ± 0.8	96.2 ± 0.4	98.0 ± 0.1	98.5 ± 0.1	98.8 ± 0.1	50.9 ± 7.6	4.2 ± 0.9	3.9 ± 1.0
E	Lalor	71.6 ± 3.2	76.3 ± 3.5	79.5 ± 3.8	83.8 ± 4.3	86.8 ± 4.7	415.6 ± 226.0	127.3 ± 109.7	125.4 ± 110.3
H	Sudbury	73.1 ± 0.5	93.9 ± 0.6	96.2 ± 0.6	97.5 ± 0.5	98.2 ± 0.5	35.1 ± 12.3	2.8 ± 1.2	1.2 ± 1.1
I	Brunswick	87.6 ± 1.4	96.4 ± 0.6	97.8 ± 0.6	98.3 ± 0.6	98.6 ± 0.6	50.2 ± 13.8	4.5 ± 2.3	3.8 ± 2.5
J	Halfmile	83.8 ± 0.5	92.6 ± 0.5	95.9 ± 0.6	97.9 ± 0.6	98.8 ± 0.6	35.2 ± 33.3	3.8 ± 4.3	2.9 ± 4.1
K	Lalor	76.3 ± 1.8	80.0 ± 1.7	82.7 ± 1.7	86.4 ± 1.9	89.0 ± 2.0	460.0 ± 73.7	123.9 ± 40.3	123.1 ± 40.4

The models were trained on each fold's training sites while using the hyperparameter configuration found based on the best HR@1px performance on that fold's validation site. For each of the 10 models, a new random seed that affects initialization weights and augmentation operations was randomly picked. The table reports the mean plus-or-minus standard deviation of all metrics obtained using these different models. The test site over which the metrics are evaluated is restated for convenience in the column "Site". The HRs are in percentage and the various errors are in the number of samples.

able parameters), a maximum VRAM size of 20 GB is required when using a batch size of 16 gathers with mixed precision. This is compatible with most high-end GPUs from the past few years; lower-end GPUs could be accommodated by reducing the batch size, which would result not only in smaller VRAM usage but also in longer training times. To give an example with practical numbers, using an NVIDIA GeForce RTX 3090 GPU, one training epoch (with three survey sites) for the largest proposed model can be completed in approximately 10 min. This means that a 49M parameter model can be trained entirely in a few hours. To predict first breaks on new data using an already-trained model, for a 512 trace gather and using only an Intel Core i7-8700K CPU, the typical inference time is between 1 and 2 s. In the end, these timings show that the training and use of pre-trained models are dwarfed by the time potentially required to manually analyze and correct picks generated by a not-fully-automated approach. The potential for transfer learning using pre-trained models is also compelling, as significant improvements in the quality of predictions on new data would likely be attainable with very few human annotations on the new data.

We also acknowledge a preprint by Wang et al. (2022) that appeared as we were writing this paper. They adopt our proposed benchmark data set and evaluation methodology and propose a multistaged approach with ingenious improvements to the basic U-Net architecture. They report excellent HR@1px performance (from 92% to 98%, depending on the fold). These numbers cannot be directly compared with our own, however, as they modified the ground truth labels by moving them to the nearest trough (aside from Lalor, we used the ground truth labels as provided). In addition, they apply a postprocessing step where predictions are rejected if they deviate by more than five pixels from a velocity-based estimation: this rejection procedure affects the TC, which is not reported, and it inflates the HR as they define it. They define HR as the ratio of the number of accurate predictions to the total number of nonrejected predictions. We define the HR with the total number of labeled traces as the denominator. By rejecting poor predictions, they reduce the value of their denominator and thus inflate the value of their HR. In short, this is precisely the kind of work we hope to foster, but it highlights the need for great care in comparing the reported performance between studies. To quantify whether new ideas bring tangible benefits, it is necessary for the community to use open-source data sets, standardized annotations, and standardized metrics.

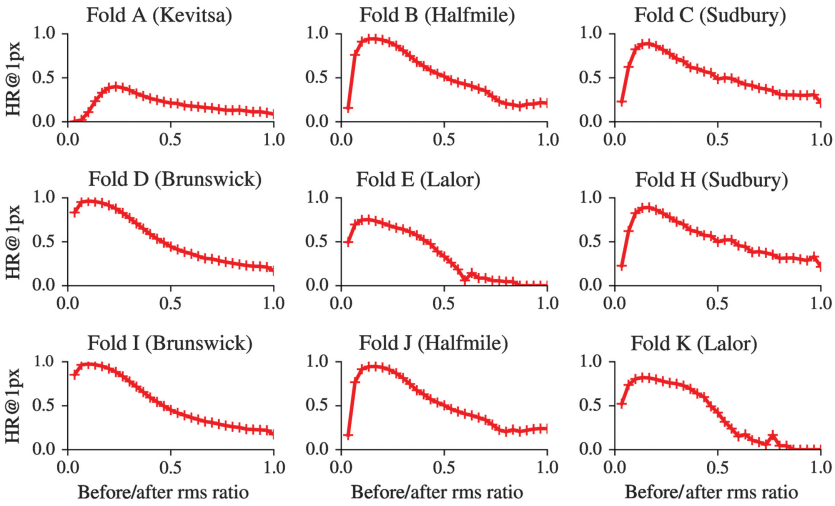

Figure 11. Relationship between the noise in the recorded seismic traces and the accuracy of model predictions on test sites in terms of HR@1px. The traces across the test set for each fold were binned into 30 intervals of before/after RMS ratios.

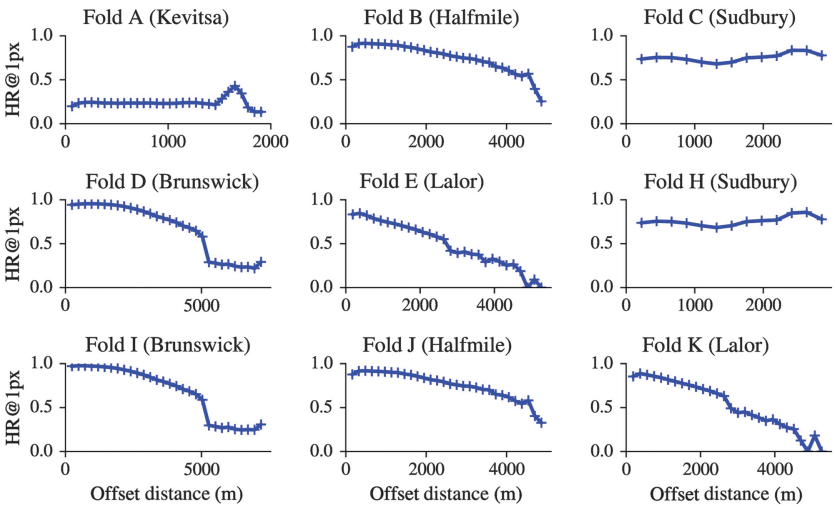

Figure 12. Relationship between the receiver-shot (offset) distances and the accuracy of model predictions in terms of HR@1px. The traces across the test set for each fold were binned into 30 intervals of offset distances. Note that the horizontal axis of each plot is scaled differently.

CONCLUSION

We have presented a multisite data set of annotated seismic traces from hardrock mining environments and benchmark results for first break picking based on the U-Net architecture following a sound methodology. Our results show that the approach of analyzing line gathers as images using fully convolutional architectures with a proper experimental protocol leads to satisfactory performance across multiple survey sites. Visual inspection of our model predictions also confirms that many line gathers can be automatically assigned first break picks of expert-level quality. In contrast to related works found in the literature, these results are meaningful due to the use of separate surveys for the training of models, for the validation of their hyperparameters, and their final evaluation. However, we have observed that our models can misbehave when there are important differences in the appearance of seismic patterns and the quality and consistency of annotations used across surveys. This is not entirely unexpected, as training models using noisy labels and applying them to

so-called out-of-distribution data sets are two active research topics in machine learning.

We ran experiments with a wide array of model architectures and hyperparameters. Interestingly, the most drastic improvements that were obtained in terms of performance over the course of our study were linked to the rectification or elimination of bad annotations. In particular, high-quality annotations in validation data sets helped us distinguish real improvements from spurious correlations. The development and promotion of an annotation standard for future data acquisition and annotation efforts would help accelerate the development of state-of-the-art methods for the task of first break picking. In parallel, we hope this work will promote the use of a standard evaluation methodology for trained models. Many works in the literature have tackled first break picking as an image segmentation problem and thus evaluated their models with image segmentation metrics; we discuss in this paper why this approach is not ideal and why all first break picking methods should always be evaluated from the perspective of a regression task.

ACKNOWLEDGMENTS

We thank E. Adam, S. Cheraghi, and A. Malehmir for providing first break picks for the Brunswick, Halfmile, Kevitsa, and Sudbury 3D seismic surveys. We thank First Quantum Minerals, Glencore, and Trevali Mining for providing access to the Brunswick, Halfmile, and Kevitsa field seismic data. This project, Geological Survey of Canada contribution 20220575, has been funded by Natural Resources Canada and the Ministère de l'Économie, de l'Innovation et de l'Énergie du Québec.

DATA AND MATERIALS AVAILABILITY

Data associated with this research are available online along with the code used to run our experiments; visit https://github.com/mila-iqia/hardpicks for more information. Note that we cannot share data from the Finland site (Kevitsa) due to licensing concerns, but report experimental results with and without this site for future comparisons. For more information on this site, see Valasti et al. (2012).

REFERENCES

Akaike, H., 1974, Markovian representation of stochastic processes and its application to the analysis of autoregressive moving average processes: Annals of the Institute of Statistical Mathematics, **26**, 363–387, doi: 10.1007/BF02479833.

Alaudah, Y., P. Michałowicz, M. Alfarraj, and G. AlRegib, 2019, A machine-learning benchmark for facies classification: Interpretation, **7**, no. 3, SE175–SE187, doi: 10.1190/INT-2018-0249.1.

Allen, R., 1982, Automatic phase pickers: Their present use and future prospects: Bulletin of the Seismological Society of America, **72**, S225–S242, doi: 10.1785/BSSA07206B0225.

Bellefleur, G., E. Schetselaar, D. White, K. Miah, and P. Dueck, 2015, 3D seismic imaging of the Lalor volcanogenic massive sulphide deposit, Manitoba, Canada: Geophysical Prospecting, **63**, 813–832, doi: 10.1111/1365-2478.12236.

Bridle, J. S., 1990, Probabilistic interpretation of feedforward classification network outputs, with relationships to statistical pattern recognition: Neurocomputing, 227–236.

Cheraghi, S., A. Malehmir, and G. Bellefleur, 2012, 3D imaging challenges in steeply dipping mining structures: New lights on acquisition geometry and processing from the Brunswick no. 6 seismic data, Canada: Geophysics, **77**, no. 5, WC109–WC122, doi: 10.1190/geo2011-0475.1.

Coppens, F., 1985, First arrival picking on common-offset trace collections for automatic estimation of static corrections: Geophysical Prospecting, **33**, 1212–1231, doi: 10.1111/j.1365-2478.1985.tb01360.x.

Cova, D., P. Xie, and P.-T. Trinh, 2020, Automated first break picking with constrained pooling networks: 90th Annual International Meeting, SEG, Expanded Abstracts, 1481–1485, doi: 10.1190/segam2020-3427812.1.

Dai, H., and C. MacBeth, 1997, The application of back-propagation neural network to automatic picking seismic arrivals from single-component recordings: Journal of Geophysical Research: Solid Earth, **102**, 15105–15113, doi: 10.1029/97JB00625.

Dosovitskiy, A., L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, 2020, An image is worth 16x16 words: Transformers for image recognition at scale: arXiv preprint, doi: 10.48550/arXiv.2010.11929.

Dumont, V., V. R. Tribaldos, J. Ajo-Franklin, and K. Wu, 2020, Deep learning on real geophysical data: A case study for distributed acoustic sensing research: arXiv preprint, doi: 10.48550/arXiv.2010.07842.

Fernhout, C., P. Zwartjes, and J. Yoo, 2020, Automatic first break picking with deep learning: IOSR Journal of Applied Geology and Geophysics, **8**, 24–36.

Gentili, S., and A. Michelini, 2006, Automatic picking of P and S phases using a neural tree: Journal of Seismology, **10**, 39–63, doi: 10.1007/s10950-006-2296-6.

Gillfeather-Clark, T., T. Horrocks, E.-J. Holden, and D. Wedge, 2021, A comparative study of neural network methods for first break detection using seismic refraction data over a detrital iron ore deposit: Ore Geology Reviews, **137**, 104201, doi: 10.1016/j.oregeorev.2021.104201.

Han, S., Y. Liu, Y. Li, and Y. Luo, 2022, First arrival traveltime picking through 3-D U-Net: IEEE Geoscience and Remote Sensing Letters, **19**, 1–5, doi: 10.1109/LGRS.2021.3096572.

Harsuko, R., and T. Alkhalifah, 2022, StorSeismic: A new paradigm in deep learning for seismic processing: arXiv preprint, doi: 10.48550/arXiv.2205.00222.

He, K., X. Zhang, S. Ren, and J. Sun, 2016, Deep residual learning for image recognition: IEEE Conference on Computer Vision and Pattern Recognition, 770–778.

Ioffe, S., and C. Szegedy, 2015, Batch normalization: Accelerating deep network training by reducing internal covariate shift: International Conference on Machine Learning, 448–456.

Kingma, D. P., and J. Ba, 2014, Adam: A method for stochastic optimization: arXiv preprint, doi: 10.48550/arXiv.1412.6980.

Liu, R., J. Lehman, P. Molino, F. Petroski Such, E. Frank, A. Sergeev, and J. Yosinski, 2018, An intriguing failing of convolutional neural networks and the CoordConv solution: Advances in Neural Information Processing Systems.

Ma, Y., S. Cao, J. W. Rector, and Z. Zhang, 2020, Automated arrival-time picking using a pixel-level network: Geophysics, **85**, no. 5, V415–V423, doi: 10.1190/geo2019-0792.1.

Magrini, F., D. Jozinović, F. Cammarano, A. Michelini, and L. Boschi, 2020, Local earthquakes detection: A benchmark dataset of 3-component seismograms built on a global scale: Artificial Intelligence in Geosciences, **1**, 1–10, doi: 10.1016/j.aiig.2020.04.001.

Malehmir, A., and G. Bellefleur, 2009, 3D seismic reflection imaging of volcanic-hosted massive sulfide deposits: Insights from reprocessing Halfmile Lake data, New Brunswick, Canada: Geophysics, **74**, no. 6, B209–B219, doi: 10.1190/1.3230495.

Malehmir, A., C. Juhlin, C. Wijns, M. Urosevic, P. Valasti, and E. Koivisto, 2012, 3D reflection seismic imaging for open-pit mine planning and deep exploration in the Kevitsa Ni-Cu-PGE deposit, northern Finland: Geophysics, **77**, no. 5, WC95–WC108, doi: 10.1190/geo2011-0468.1.

Milkereit, B., E. Berrer, A. R. King, A. H. Watts, B. Roberts, E. Adam, D. W. Eaton, J. Wu, and M. H. Salisbury, 2000, Development of 3-D seismic exploration technology for deep nickel-copper deposits — A case history from the Sudbury basin, Canada: Geophysics, **65**, 1890–1899, doi: 10.1190/1.1444873.

Nikita, K., P. Dmitry, K. Alexander, and S. Daniil, 2021, An automated pipeline for first break picking and identifying geometry errors: First Break, **39**, 67–71, doi: 10.3997/1365-2397.fb2021093.

Ronneberger, O., P. Fischer, and T. Brox, 2015, U-Net: Convolutional networks for biomedical image segmentation: International Conference on Medical Image Computing and Computer-Assisted Intervention, 234–241.

Sabbione, J. I., and D. Velis, 2010, Automatic first-breaks picking: New strategies and algorithms: Geophysics, **75**, no. 4, V67–V76, doi: 10.1190/1.3463703.

Sleeman, R., and T. Van Eck, 1999, Robust automatic P-phase picking: An on-line implementation in the analysis of broadband seismogram recordings: Physics of the Earth and Planetary Interiors, **113**, 265–275, doi: 10.1016/S0031-9201(99)00007-2.

Sudre, C. H., W. Li, T. Vercauteren, S. Ourselin, and M. J. Cardoso, 2017, Generalised dice overlap as a deep learning loss function for highly unbalanced segmentations, in Deep learning in medical image analysis and multimodal learning for clinical decision support: Springer, 240–248.

Tan, M., and Q. Le, 2019, EfficientNet: Rethinking model scaling for convolutional neural networks: International Conference on Machine Learning, 6105–6114.
Valasti, P., A. Malehmir, and C. Wijns, 2012, 3D seismic surveying in Kevitsa open pit mine: Australian Society of Exploration Geophysicists (ASEG) Conference and Exhibition, Extended Abstracts.
Veeken, P. C. H., 2007, Chapter 2 — The seismic reflection method and some of its constraints, in P. C. H. Veeken, ed., Seismic stratigraphy, basin analysis and reservoir characterisation: Pergamon, Handbook of Geophysical Exploration: Seismic Exploration 37, 7–109.
Wang, H., J. Zhang, X. Wei, C. Zhang, Z. Guo, L. Long, and Y. Wang, 2022, MSSPN: Automatic first arrival picking using multi-stage segmentation picking network: arXiv preprint, doi: 10.48550/arXiv.2209.03132.
Xie, T., Y. Zhao, X. Jiao, W. Sang, and S. Yuan, 2019, First-break automatic picking with fully convolutional networks and transfer learning: 89th Annual International Meeting, SEG, Expanded Abstracts, 4972–4976, doi: 10.1190/segam2019-3215277.1.
Yilmaz, Ö., 2001, Seismic data analysis: Processing, inversion, and interpretation of seismic data: SEG.
Yordkayhun, S., A. Ivanova, R. Giese, C. Juhlin, and C. Cosma, 2009, Comparison of surface seismic sources at the CO_2 Sink Site, Ketzin, Germany: Geophysical Prospecting, **57**, 125–139, doi: 10.1111/j.1365-2478.2008.00737.x.
Yu, S., and J. Ma, 2021, Deep learning for geophysics: Current and future trends: Reviews of Geophysics, **59**, e2021RG000742, doi: 10.1029/2021RG000742.
Yuan, P., S. Wang, W. Hu, X. Wu, J. Chen, and H. Van Nguyen, 2020, A robust first-arrival picking workflow using convolutional and recurrent neural networks: Geophysics, **85**, no. 5, U109–U119, doi: 10.1190/geo2019-0437.1.
Yuan, S.-Y., Y. Zhao, T. Xie, J. Qi, and S.-X. Wang, 2022, SegNet-based first-break picking via seismic waveform classification directly from shot gathers with sparsely distributed traces: Petroleum Science, **19**, 162–179, doi: 10.1016/j.petsci.2021.10.010.
Zhang, C., M. Fiore, I. Murray, and P. Patras, 2019, CloudLSTM: A recurrent neural model for spatiotemporal point-cloud stream forecasting: arXiv preprint, doi: 10.48550/arXiv.1907.12410.
Zheng, J., J. M. Harris, D. Li, and B. Al-Rumaih, 2020, SC-PSNET: A deep neural network for automatic P- and S-phase detection and arrival-time picker using 1C recordings: Geophysics, **85**, no. 4, U87–U98, doi: 10.1190/geo2019-0597.1.
Zwartjes, P., M. Fernhout, and J. Yoo, 2020, Evaluation of neural network architectures for first break picking: 82nd Annual International Conference and Exhibition, EAGE, Extended Abstracts, doi: 10.3997/2214-4609.202010331.
Zwartjes, P., and J. Yoo, 2022, First break picking with deep learning — Evaluation of network architectures: Geophysical Prospecting, **70**, 318–342, doi: 10.1111/1365-2478.13162.

Biographies and photographs of the authors are not available.