Geophysical Journal International

Geophys. J. Int. (2024) **239,** 862–881
Advance Access publication 2024 August 23
GJI Seismology

https://doi.org/10.1093/gji/ggae298

Deep learning models for regional phase detection on seismic stations in Northern Europe and the European Arctic

Erik B. Myklebust [1] and Andreas Köhler [1,2]

[1] *NORSAR, Test Ban Treaty Verification,* 2007 *Kjeller, Norway. Email:* andreas.kohler@norsar.no
[2] *UiT - The Arctic University of Norway, Department of Geosciences,* 9037 *Tromsø, Norway*

Accepted 2024 August 21. Received 2024 August 8; in original form 2024 March 11

SUMMARY

Seismic phase detection and classification using deep learning is so far poorly investigated for regional events since most studies focus on local events and short time windows as the input to the detection models. To evaluate deep learning on regional seismic records, we create a data set of events in Northern Europe and the European Arctic. This data set consists of about 151 000 three component event waveforms and corresponding phase arrival picks at stations in mainland Norway, Finland and Svalbard. We train several state-of-the-art and one newly developed deep learning model on this data set to pick *P*- and *S*-wave arrivals. The new method modifies the popular PhaseNet model with new convolutional blocks including transformers. This yields more accurate predictions on the long input time windows associated with regional events. Evaluated on event records not used for training, our new method improves the performance of the current state-of-the-art methods when it comes to recall, precision and pick time residuals. Finally, we test our new model for continuous mode processing on 4 d of single-station data from the ARCES array. Results show that our new method outperforms the existing array detector at ARCES. This opens up new opportunities to improve automatic array processing with deep learning detectors.

Key words: Machine learning; Seismology; Earthquake monitoring and test-ban treaty verification.

1 INTRODUCTION

Seismic event detection and phase arrival picking are crucial first steps in automatic monitoring pipelines. Reliable picking is required for associating phase detections to events and subsequently locating them. Traditional phase picking methods are often based on power detectors, such as the well-known STA/LTA (Short-Term/Long-Term average) trigger. Furthermore, more advanced approaches, making use of other characteristic functions of the seismic waveform based on various statistical moments and polarization attributes, as well as autoregressive models have been used (Withers *et al.* 1998; Bai & Kennett 2000). Most of these methods do not provide a phase classification and/or do not consider the temporal context of an arrival, for example that it is more likely that a *P* wave is observed if it is followed by an *S* wave. In recent years, the integration of machine learning methods into seismic processing pipelines has made huge progress, taking advantage of manually reviewed seismic event catalogues established over decades (Bergen *et al.* 2019; Kong *et al.* 2019; Mousavi & Beroza 2023). Many deep learning methods have been developed and tested for the detection of phase arrivals of local events (Zhu & Beroza 2018; Mousavi *et al.* 2020; Li *et al.* 2022; García *et al.* 2022; Park *et al.* 2024). Large data sets for training these types of models have been compiled, for example the STanford EArthquake Data set (STEAD, Mousavi *et al.* 2019), the Italian Seismic Data set For Machine Learning (INSTANCE, Michelini *et al.* 2021), a region-specific data set for the Pacific Northwest (Ni *et al.* 2023), and most recently the Curated Regional Earthquake Waveforms (CREW) data set (Aguilar Suarez & Beroza 2024). STEAD consists of 60-s-long waveforms recorded at seismic stations within epicentral distances of 100–350 km. The INSTANCE data set consists of 120-s-long waveforms with *P*- and *S*-arrival picks from events at up to 600 km distance. Both data sets have been used extensively in the validation of phase arrival picker methods. Zhu & Beroza (2018) adapted the UNet (Ronneberger *et al.* 2015) architecture commonly used in image segmentation to the task of phase picking. This method is known as PhaseNet and has been the *de facto* baseline for many studies. Recently, fuelled by advances in language processing, the addition of attention (Bahdanau *et al.* 2016) and transformers (Vaswani *et al.* 2017) has been added to picker models, for example EPick (Li *et al.* 2022) or EQTransformer (Mousavi *et al.* 2020).

So far, few deep learning models have been adapted and tested for events at regional and teleseismic distances, that is further than a few hundred kilometres from a sensor. Wang *et al.* (2019) tested

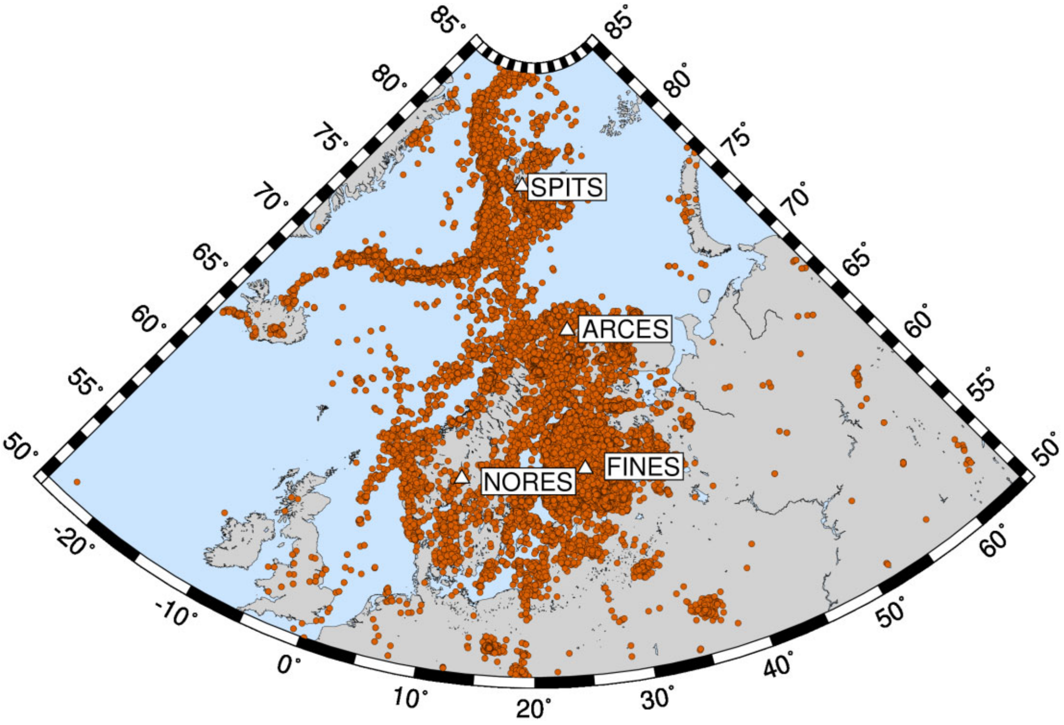

Figure 1. Map of seismic stations (triangles) and seismic events (circles) in Northern Europe and the Arctic used in this study.

a model trained on local data on regional events. Münchmeyer *et al.* (2024) developed a model specialized for picking teleseismic depth phases. Münchmeyer *et al.* (2022) provided a thorough evaluation of different deep learning phase picking models, including regional and teleseismic distance ranges, and concluded that it is most important to use a training data set from the appropriate distance range. Furthermore, Mai *et al.* (2023) developed a framework to customize deep-learning phase pickers by transfer learning and fine-tuning, which allows including regional events.

Seismic recordings from regional distances are crucial for regional and global seismic event monitoring as required for verification of the Comprehensive Nuclear Test Ban Treaty (CTBT, Kalinowski & Mialle 2021). CTBT verification relies on a global seismic network which is part of the IMS (International Monitoring System of the CTBTO). The treaty restricts verification to IMS stations, hence, many events are not observed at local distances on this sparse network. Automatic detection algorithms for regional arrivals are therefore crucial, especially when the event size prevents observation at teleseismic distances. Observations only at regional distances is also a common limitation with other sparse (national) seismic networks, in particular when monitoring off-shore regions.

In this study, we create a data set of seismic waveforms from regional events recorded at stations in Northern Europe and the European Arctic (Section 2) to train and evaluate existing, state-of-the-art deep learning phase picking methods (Section 3). We further develop those methods to improve the performance on regional events. The data set includes regional events from reviewed bulletins produced in Norway and Finland as well as arrivals from the International Data Centre Late Event Bulletin (IDC LEB), a reviewed bulletin produced by the CTBTO. The phase detection methods are evaluated on the test data sets, and the best-performing model is applied to continuous data from selected seismic stations to assess the performance of automatic single-station processing (Section 4).

2 DATA SET

Seismic events in the Nordic countries and the surrounding regions, including the European Arctic, are included in the bulletins manually reviewed by analysts at NORSAR, Norway, (NORSAR Reviewed Bulletin, NRB hereafter, NORSAR 1971a), at the Institute of Seismology, University of Helsinki, Finland (Helsinki Reviewed Bulletin, HRB hereafter, Veikkolainen *et al.* 2021) and at the IDC (LEB). These bulletins have overlapping content and use arrivals at common stations. Details about spatial coverage of the NRB and HRB can be found in Köhler & Myklebust (2023). The spatial distribution of epicentre location is shown in Fig. 1. We use about 151 000 three-component seismic waveforms including arrivals from these event catalogues in the period 2000–2022. This corresponds to about 100 000 individual seismic events. We restrict ourselves to mostly regional events (up to about 2000 km distance to epicentre) and some events approaching near-teleseismic distances (2000–5000 km). Local events are also included (less than 200 km distance), however, due to our station selection, the regional events dominate. The selected events include earthquakes off-shore Norway, along the Mid-Atlantic ridge and in the Northern Atlantic region around Svalbard. In Northern and Eastern Scandinavia, Finland and the Kola Peninsula, frequent seismic signals from mining operations are included. The magnitude distribution of the events in the data set is shown in Fig. A1, and we come back to the epicentre distance distribution later. We use waveform data recorded on four IMS primary and auxiliary seismic stations (Fig. 1): ARA0 (central station of ARCES array, northern Norway), SPA0 (central station of SPITS array, Svalbard), FIA0 (central station of FINES array, southern Finland) and NRA0 (central station of NORES array, southern Norway). NORES is not an IMS station but NRA0 is colocated with the IMS station

Table 1. Number of picked seismic arrivals in the data set compiled for this study.

Bulletin	Station	P picks	S picks	Events
NRB	ARA0	13 401	13 191	12 356
NRB	NRA0	838	648	768
NRB	SPA0	2622	1710	2208
HRB	ARA0	57 200	63 315	64 416
HRB	FIA0	28 855	28 389	28 768
LEB	ARA0	28 223	5848	26 400
LEB	FIA0	14 610	2886	14 816
LEB	SPA0	1450	1	1440

NC602. The instruments for most stations are CMG-3T broad-band sensors.

We extract 9-min-long waveforms around each event in the catalogues, starting 4.5 min before the first picked arrival if there is only a single pick. For multiple picks (majority of the waveforms) the noise time window before the first pick is shortened by half the time difference between the last and first pick, that is if the pick time difference is 1 min, the first pick is at 4 min. Later in the process of preparing the input data for the deep learning methods, the time windows are cut randomly to 5 min to help to generalize the prediction capability of the models. Table 1 includes the number of events and the number of *P* and *S* picks at the different stations for the different bulletins. Note that some event windows may include multiple *P* and *S* picks, for example from Pn and Pg arrivals or overlapping events. We drop the regional phase labels, and label all arrivals either *P* or *S* for model training. Lg wave picks are labelled as *S* waves. The reasoning behind restricting the phase labels to *P* and *S* waves is related to incomplete phase picking by the analyst, and we come back to this in the discussion section. Due to instrument upgrades at FINES (2007), SPITS (2014) and NORES (2015), we use only events after those years for stations FIA0, SPA0 and NRA0 to ensure identical sensor responses. This is also partly why FIA0 and ARA0 arrivals dominate the record. The impact this has on model training is discussed below. The waveforms are filtered between 2 and 8 Hz. Based on experience at the observatory at NORSAR, this frequency band is the most optimal for regional event monitoring in this study region. All stations except SPA0 and NRA0 are sampled at 40 Hz. We downsample SPA0 and NRA0 data to 40 Hz to obtain the same input data dimension.

We decided to not sort out arrivals that are included in more than one bulletin, although the random cutting of the 5-min window means that it is unlikely there will be exact duplicates. Note that the approach to define test and validation data explained below ensures that an arrival included in the test or validation data is not part of the training data. The reason for not removing repeated arrivals is partly related to potential issues with the quality and the completeness of the used bulletins. For example, we noticed that picks may be missing or may have slightly different times in the different catalogues for the same event. Furthermore, some picks are included that show almost no visible signals, where most likely the analyst used theoretical arrival times, array beams, or the moveout observed on a station network to steer picking of arrivals with low signal-to-noise ratio (SNR). Hence, we want to emphasize that our data is not a flawless, re-analysed curated data set for benchmarking machine-learning models such as STEAD. In general, we trust the high quality bulletins generated by trained analysts, but must be aware of some shortcomings. Keeping repeating events may partly compensate for these issues.

3 METHOD

This section describes the deep learning methods used for phase picking. These are published baseline models and our modification of PhaseNet. Vectors are denoted as $\mathbf{x}$, and matrices (and multidimensional tensors) as X. We denote $\hat{y}$ as model prediction while y is the ground truth label.

3.1 Training data set preparation

A three-component waveform, which is the input of each deep learning model, is defined as $W \in \mathbb{R}^{T\times 3}$, where T is the number of time samples of the waveform. The true phase labels are created from two zero arrays, one array for P and one for S, and a point impulse of value one is added at the index closest to the P and S times from the catalogues. For EQTransformer, we follow the definition of the detection label from the original paper (Mousavi *et al.* 2020).

Input data augmentation is a powerful strategy to improve the generalization of the models. We follow the augmentation strategies of Mousavi *et al.* (2020). This includes:

(i) Addition of noise: We add white noise drawn from a Normal distribution $\mathcal{N}(0, [\max(W) \times U(0.01, 0.15)]^2)$ to the waveform, where variance depends on a factor drawn from the uniform distributions U. Noise is only added if $u < p_n$, with u drawn form a uniform distribution $U(0, 1)$ and p_n a preset threshold.

(ii) Addition of secondary event: An event is drawn from the data set, is shifted randomly, and added to the target event with a random amplitude scale ($U(0.1, 1)$). An event is only added if $u < p_e, u \sim U(0, 1)$.

(iii) Channel dropping: A channel of the three-component waveform is randomly dropped by setting input values to zero. A channel is only dropped if $u < p_d, u \sim U(0, 1)$.

(iv) Addition of data gap: A data gap is randomly added to a channel of the waveform. This gap is a maximum 10 per cent of the total length. A gap is only created if $u < p_g, u \sim U(0, 1)$.

(v) Random cropping: Each waveform is cropped randomly to 5 min length. This is to ensure that the arrivals are at different points every time such that the model does not learn picking arrivals based on where in the window they will appear.

The parameters p_n, p_e, p_d, p_g are provided in Table A1. Note that augmentation is only used on the training set. The augmentations are applied to each sample and are different for each iteration during training (epoch). In addition, all waveforms are normalized by the standard deviation of the waveform amplitude across all channels and a taper is used to avoid boundary effects.

The label arrays have the same time dimension size as the input waveforms. We use three classes: P-arrival, S-arrival and noise. We define noise as $N = 1 - P - S$ where P and S are the probabilities of P- and S-arrivals for each time sample. As mentioned above, the distributions for P and S are originally point impulses (probability is one at the picks time and zero otherwise). We apply smoothing using a Gaussian filter with a standard deviation of $0.275s$ to avoid harsh punishment of the model, that is to avoid that the model is equally penalized when predicting an arrival one or one-hundred time-steps from the correct pick. Note that we add labels of all picked arrivals in overlapping event time windows, that is a P wave of a subsequent event may be present after the S wave in a given event time window.

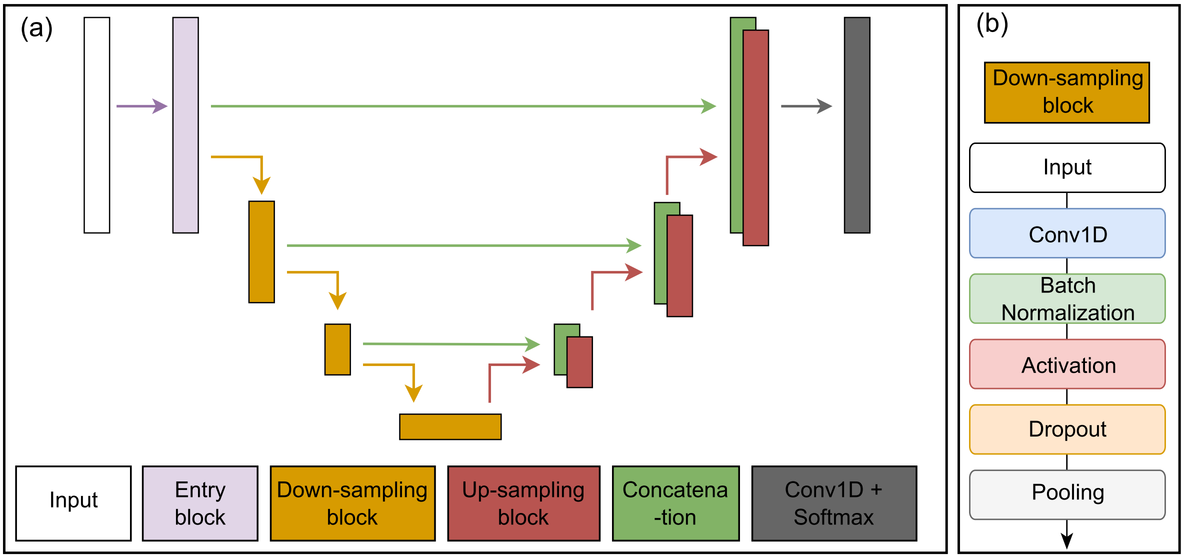

Figure 2. (a) Generic PhaseNet architecture. The input is normalized waveforms, $W \in \mathbb{R}^{T\times 3}$; the entry block increases the feature dimension by convolution (i.e. block shown in b without pooling). Each down-sampling block shown in (b) uses convolution and reduces the time dimension by half to extract higher level features, while the up-sampling increases the time-dimensionality by two and applies a convolutional layer. The skip-connections are concatenated to the up-sampled output. Finally, a 1 × 1 convolutional layer with three filters is used to create the class probabilities. (b) PhaseNet down-sampling block consisting of convolution layer, batch normalization, non-linear activation, dropout, and finally pooling.

3.2 Phase detector models

The models used in this work are formulated as a prediction of noise, *P*-arrival and *S*-arrival per time step given three-component waveforms as input, that is

$$\hat{Y} = M(W), \;\; W \in \mathbb{R}^{T\times 3}, \;\; Y \in (0, 1)^{T\times 3}, \quad (1)$$

where M is the prediction model. The prediction $\hat{Y}$ contains a probability distribution for each time step, for example $\hat{\mathbf{y}}_t = (0.1, 0.85, 0.05)$ means that there is an 85 per cent probability for a *P*-arrival at time t.

Except for EQTransformer (Mousavi *et al.* 2020), all other models in this work are based on PhaseNet (Zhu & Beroza 2018), which is a one-dimensional version of UNet Ronneberger *et al.* (2015). The PhaseNet architecture is shown in Fig. 2. PhaseNet uses a series of blocks that extract features and down-sample the input. Each block consists of a convolutional layer, batch normalization, activation and dropout layer. Finally, a pooling operation (maximum or average) is performed to reduce the time dimension size (Fig. 2b). The output of each block is kept for use in a skip-connection during up-sampling. A skip-connection is an element in the neural network where output features of a layer/block are copied and concatenated with the output of a later layer/block. The up-sampling blocks contain an up-sampling layer (duplication of points in the time dimension), concatenation with skip-connections from the down-sampling blocks, inverse convolutional layer, batch normalization, activation and dropout.

The EPick model (Li *et al.* 2022) follows the same architecture as PhaseNet, albeit attention is applied as residual connections, that is, from input and output of a convolutional block an attention vector is created (Li *et al.* 2022). Attention is used in Neural Networks to introduce weights for feature vectors (originally developed for the embedding of a word in natural language processing models) in the context window (originally a sentence), here the time window being processed. These weights are ‘soft’, meaning they are dependent on the input features, in contrast to ‘hard’ weights, which are found once during training. As the name suggests, attention helps the model to focus its attention on what it finds important; in this case samples in the time window including potential *P* and *S* arrivals. The attention mechanism is repeated several times in the network, each time using the previous layers as input and adding the output to the up-sampling block (Fig. 3). More details about the attention mechanism are given in Appendix B.

EQTransformer (Mousavi *et al.* 2020) is a transformer-based architecture for the task of phase picking and event detection. This method uses similar down-sampling blocks as PhaseNet to encode the input, followed by residual convolutional blocks, Long Short-Term Memory (LSTM) blocks, and transformer blocks, the latter being a multihead attention mechanism implemented in a particular way (Fig. 4b): The residual connections are added to the output of the attention (i.e. a skip-connection as introduced for PhaseNet), and the result are normalized. This is then passed to a feed forward network (dense layers with trainable weights), and the residual is added and normalized again. The unique feature of EQTransformer is that each phase (*P* and *S*), and in addition the event detection probability are predicted using different decoders (Fig. 4). The decoders consist of a transformer block, and up-sampling blocks similar to PhaseNet. The final layers predict the probability of a phase or event detection per time step. Since the event detection probability is trained for as well, the training data also needs to include examples of pure noise waveforms which we provide as time windows starting 36 min before each event in the training set, that is about 151 000 time windows.

We will compare these three state-of-the-art baseline models with our a method which we call TPhaseNet. Inspired by Oktay *et al.* (2018) and Chen *et al.* (2021), we add transformers to PhaseNet. This results in a model which is similar to PhaseNet, but we use 7 down-sampling blocks rather than 4 down-sampling blocks and we add transformers to the last 4 down-sampling blocks (Fig. 5). The number of up-sampling blocks was also increased accordingly. The reason for not including transformers in the first 3 down-sampling blocks is memory limitations of the hardware. This modified architecture thus includes different innovations of previous phase

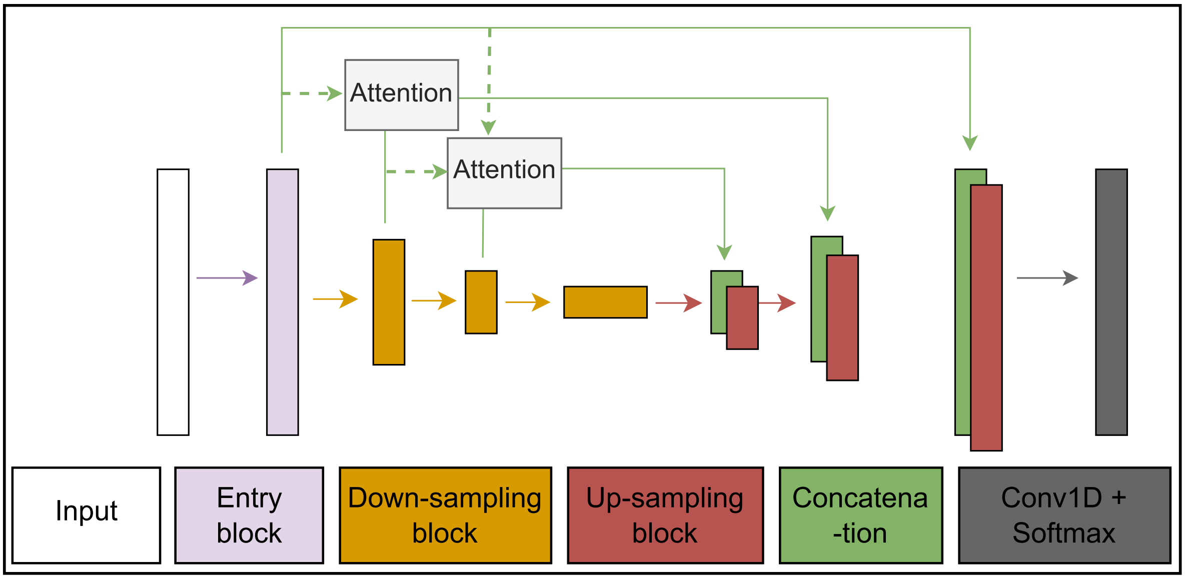

Figure 3. EPick architecture. Similar to PhaseNet as seen in Fig. 2; however, with attention mechanisms between each of the outputs of the down-sampling blocks. The output attention is used as skip-connection rather than the output from each down-sampling block as for PhaseNet.

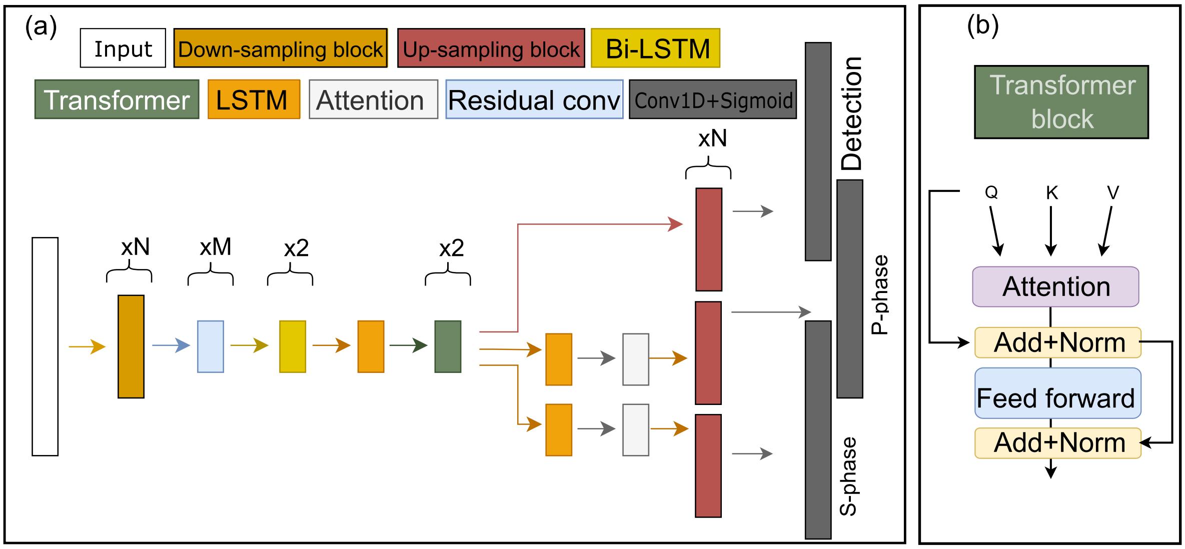

Figure 4. (a) EQTransformer architecture. The input is down-sampled (using N convolutional blocks) similar to PhaseNet before a series of M residual convolutional blocks are applied. Then two Bi(directional)-LSTM layers followed by a directional LSTM are used before two transformer blocks are used as shown in (b). This constructs the common feature extractor. The network is then split into three branches which use the extracted features to predict detection, P-phase and S-phase. An LSTM layer followed by attention is used before the up-sampling layers in the P-phase and S-phase branches. The final layers have a single output per time-step and a sigmoid function is applied to get a probability of detection, P-phase, and S-phase. (b) Transformer block. Query (Q), Key (K) Value (V) are inputs and the output has the same shape as the query. The residual connections from Q are added to the output of the attention, and the result is normalized. This is then passed to a feed forward network (dense layers), and the residual is added and normalized again.

picker models, that is the proven and powerful PhaseNet architecture, adding transformers instead of only attention to it (EPick) as a well-proven element in, for example natural language processing models, and using transformers in a different way than EQTransformer. These modifications are supposed to enhance performance when processing long records of regional events. We arrived at the final TPhaseNet model through an iterative process of trying different model architectures.

3.3 Model training

We use events from the years 2000–2020 for training (124 708 waveforms), events of the year 2021 for validation (13 632 waveforms) and 2022 for testing (12 832 waveforms). As mentioned above, this avoids using the same events during training and validation/testing. After each iteration (epoch) during model training, the performance (validation loss) is evaluated with respect to the validation data. The

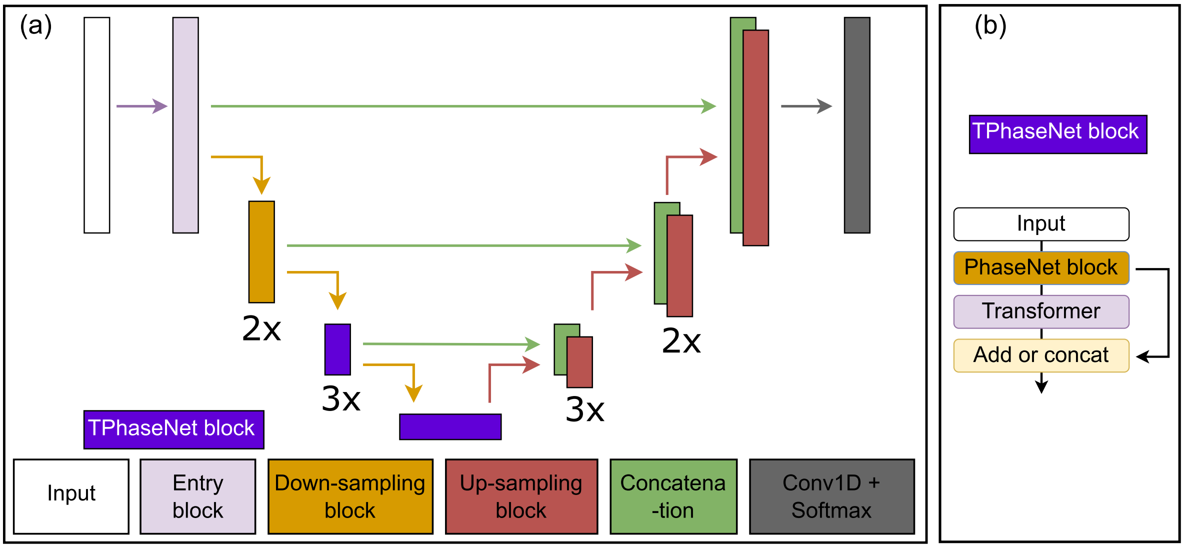

Figure 5. (a) TPhaseNet architecture. It is similar to PhaseNet, but with more blocks and the last four down-sampling blocks (TPhaseNet blocks) include transformers. For the sake of simplicity of the figure, multiple PhaseNet and TPhaseNet blocks are not drawn, that is 2× means the block is included twice. (b) TPhaseNet block: It includes a version of the PhaseNet down-sampling block (Fig. 2b) and the transformer shown in Fig. 4(b). Finally, the residual is concatenated.

test data set is unseen during training and will provide a final unbiased evaluation of the models.

Note that we implement all models from scratch in `keras` (Chollet *et al.* 2015) and do not use the published code of previous models directly. In that way we ensure comparability and flexibility when it comes to hyper-parameter setting and retraining, and we can consistently assess the models using the same software libraries. This also allows us to increase the duration of input waveforms which is required when regional events are the main targets. Originally, previous models were trained on local events. For all our models we use a 5-min-long time window.

All models, except EQTransformer, use categorical cross-entropy as the loss function to optimize during training,

$$CCE(Y, \hat{Y}) = -\frac{1}{NT}\sum_{i=1}^{N}\sum_{t=1}^{T}\sum_{c=1}^{3} Y_{itc}\log(\hat{Y}_{itc}), \tag{2}$$

where N is the number of samples and T is the number of time steps. $\hat{Y}$ holds all predictions, that is $\hat{Y} \in (0, 1)^{N\times T\times 3}$.

EQTransformer uses binary cross-entropy for each of the two outputs (or three if event detection is used):

$$BCE(Y, \hat{Y}) = -\frac{1}{NT}\sum_{i=1}^{N}\sum_{t=1}^{T}[Y_{it}\log(\hat{Y}_{it}) + (1 - Y_{it})\log(1 - \hat{Y}_{it})]. \tag{3}$$

The sum of the losses of all outputs is the final loss.

All models are trained in the same manner using the same dropout, optimizer, learning rate, weight decay and batch size (Table A1). We use a batch size of 32, maximum epochs of 200, starting learning rate of 10^{-3} and L_1 and L_2 normalization factor of 10^{-3}. The learning rate is reduced by a factor of $\sqrt{0.1}$ after seven epochs of no improvement in the validation loss. We use early stopping and the minimum delta of the early stopping is set to 10^{-4}. We stop training when there is no improvement for 15 epochs in validation loss.

The hyper-parameters for each model are described in Appendix A and listed in Table A2. They where chosen based on trial and error, however, they were kept similar between all models. Certain parameters are not shared between models. In this case, we based our choices on the original publication or the model size, that is hyperparameters are used which achieve similar model sizes as the other models. All models were trained on a single NVIDIA A40 GPU, with the final PhaseNet model taking 4 hr, TPhaseNet model 17 hr, EPick 7 hr and EQTransformer 15.5 hr to converge. The models including transformers take longer to converge due to a higher number of weights to be learned.

3.4 Evaluation metrics

We evaluate the models using precision $Prec$, recall Rec and F_1 metrics for binary prediction for different decision thresholds as well as how accurate the models are for picking arrivals times, that is whether a prediction is close to the analyst pick (residual). The metrics are defined as follows:

$$Prec = \frac{TP}{TP + FP}, \quad Rec = \frac{TP}{TP + FN}, \tag{4}$$

where TP are True Positives, that is number of peaks above the decision threshold within a distance ±2 s to a true arrival pick. False Negatives FN is the number of true picks without a peak above the decision threshold within that distance. False Positives FP are those predicted peaks not matching any true arrival within 2 s. An important note must be made for these instances. In our data set not all arrivals of an event included are necessarily picked by the analyst, particularly when multiple regional phase arrivals are present (Pg, Pn, Sg, Sn). Hence, precision may appear lower than it actually is. We present a more thorough evaluation in the next section, where a manually reviewed continuous time period is used for testing. Finally, F_1 is the harmonic mean between precision and recall, that is $2 \cdot Prec \cdot Rec/(Prec + Rec)$. The mean and standard deviations of time residuals are computed for all True Positives.

The model could have similar precision and recall but we would also like to measure the uncertainty, described by the width of the probability distribution around each predicted peak. In other words, we want to use the complete times-series of P and S arrival probabilities provided by the model output to evaluate the performance.

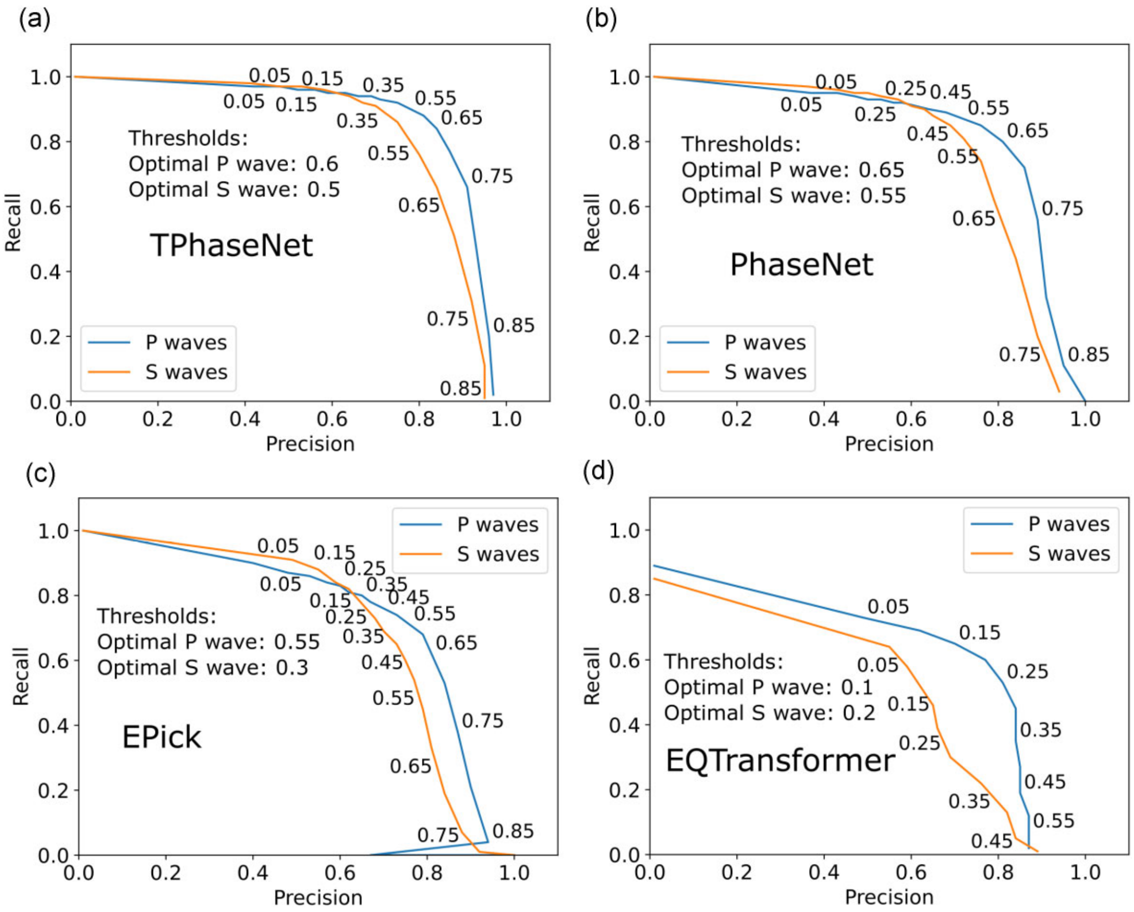

Figure 6. Performance metrics for all the trained models for different decision thresholds shown as number along the curves. The closer the displayed curves bend towards the upper-right corner, the better the performance. Optimal thresholds are given.

For this we use the Jensen–Shannon divergence (JSD), which is a symmetric version of the Kullback–Leibler divergence (KLD). Symmetry is important here as we would not like the models to predict S arrivals in the presence of P arrivals. The KLD does not penalize this scenario, while the JSD does. JSD is described as

$$JSD(P||S) = \frac{KLD(P||M) + KLD(S||M)}{2}, \quad M = \frac{P+S}{2}, \tag{5}$$

where KLD is

$$KLD(P||S) = \sum_{x \in X} P(x) \log\left(\frac{P(x)}{S(x)}\right), \tag{6}$$

and P and S are two normalized distributions to be compared, here the true and predicted probabilities of P or S arrivals.

4 RESULTS

We will first give a quantitative evaluation by presenting the results of applying all trained models to the test data set, which was not used for training (events in year 2022). Then, we will show examples of individual events to assess the model performance qualitatively. Finally, we apply the best model to continuous data from selected single stations within the ARCES array to evaluate performance in continuous processing.

4.1 Prediction results for test data

To evaluate the model performance we first need to find an optimal decision threshold for each model and phase type. There is a trade-off between precision and recall; a higher threshold will reduce False Positives, that is increases precision, but also increases False Negatives, that is reduces recall. Fig. 6 illustrates this behaviour for all trained models. Finding the optimal decision threshold is equivalent to maximizing the F1 score. The optimal thresholds shown in Fig. 6 are used to compute the final performance metrics for each model after predicting on the test data set (Table 2). Fig. 7 shows a more detailed overview of how model performance depends on different seismic stations, bulletins and epicentral distance ranges. To facilitate assessment of the results the number of phase arrivals in the test data set for each category are shown as histograms.

Based on the computed metrics it is clear that TPhaseNet and PhaseNet perform better than both EPick and EQTransformer on our data set of regional events. Moreover, adding transformers to PhaseNet clearly improves results since TPhaseNet is overall the best performing model when it comes to all computed classification metrics as well as pick time residuals. The improvement compared to PhaseNet is most significant for P wave recall (from 0.8 to 0.88) and S wave recall (from 0.81 to 0.86). We observe that precision is generally lower than recall, that is false detections are present. This result may be related to the already mentioned missing picks in the test data set, which we will come back to in the next sections.

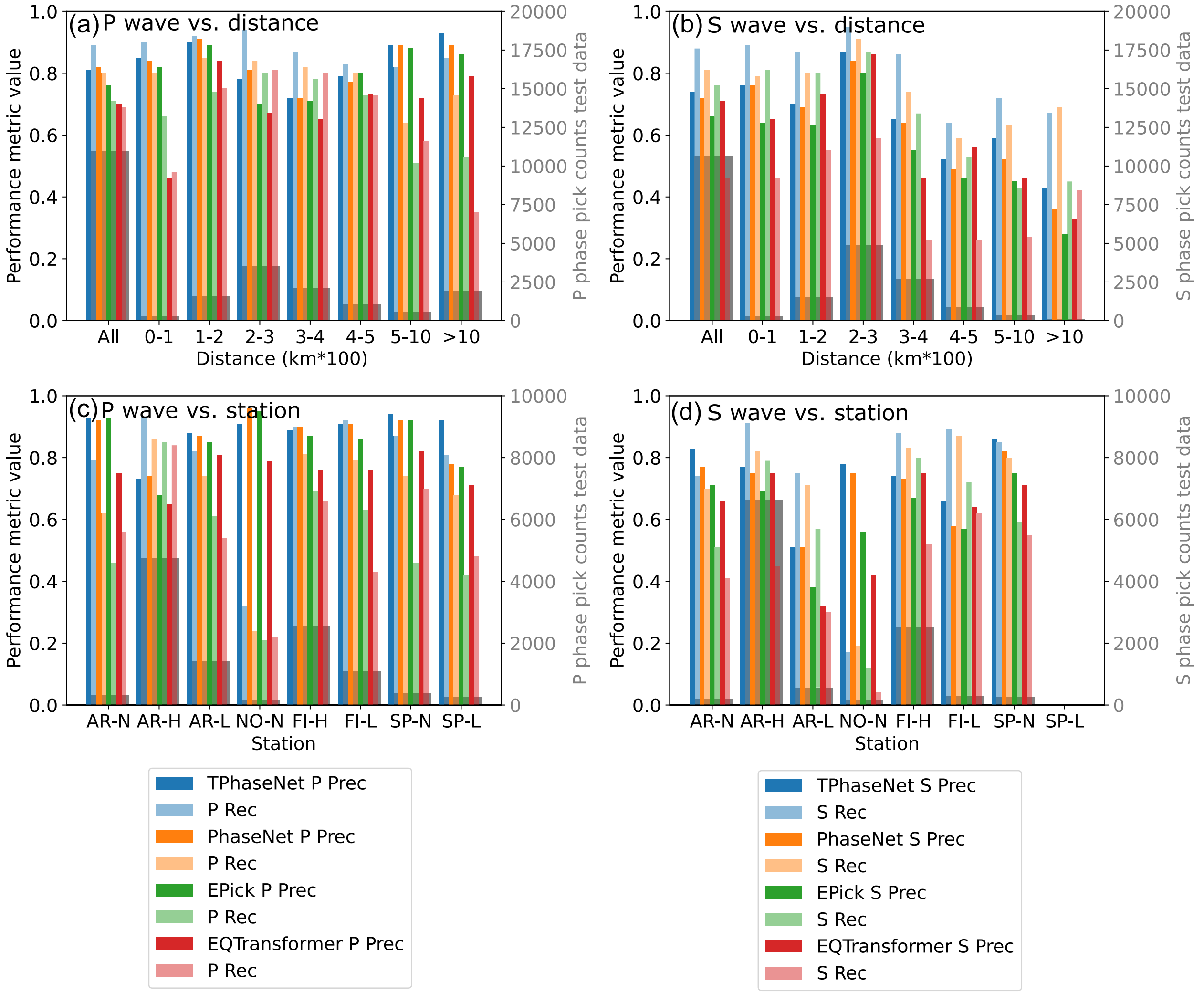

Figure 7. Performance metrics for all trained models for different epicentral distance ranges, seismic stations and event bulletins. H stands for Helsinki (HRB), N for NORSAR (NRB) and L for IDC LEB. The optimal decision thresholds are used for each model and phase type (see Fig. 6). Number of arrivals in each category are shown as grey histograms in the background. Legend below (c) is also valid for (a). Legend below (d) is also valid in (b).

Table 2. Metrics for optimal decision thresholds. Precision, recall, and F_1 are calculated with a time tolerance of 2 s. Residuals are computed for True Positives only. Bold indicates the best metric. Arrows indicate if increasing or decreasing metrics indicate better performance.

Model	Precision (P/S) ↑	Recall (P/S) ↑	F_1 (P/S) ↑	Residual ($\mu \pm \sqrt{\sigma}$) (P/S) ↓	JSD (P/S) ↓
TPhaseNet	**0.81/0.75**	**0.88/0.86**	**0.84/0.8**	$\mathbf{0.02s/0.07}s$ ($\mathbf{\pm0.46/0.57}$)	**0.25/0.34**
PhaseNet	**0.81**/0.72	0.8/0.81	0.81/0.76	$0.06s/0.16s$ ($\pm0.49/0.61$)	0.29/0.37
EPick	0.76/0.66	0.71/0.76	0.74/0.71	$0.09s/-0.16s$ ($\pm0.58/0.63$	0.33/0.36
EQTransformer	0.62/0.65	0.69/0.46	0.65/0.54	$0.36s/-0.08s$ ($\pm0.66/0.79$)	0.46/0.44

The *P* and *S* wave residuals of the correctly predicted arrivals decrease by more than 50 per cent for both *P* and *S* waves when TPhaseNet is used instead of PhaseNet. However, the residuals of PhaseNet are already quite small, that is less than 0.1 s. Fig. A2 (note logarithmic scale) shows that the distribution of residuals is symmetric for all models except EQTransformer, which tends to pick too late arrivals. EQTransformer also requires a low decision threshold for the best performance, and does not perform well compared to the three other models for all metrics. The reason for this is not entirely clear. It may be related to the model output of event detection probability in addition to phase probabilities which may not be optimally tuned or suitable for our regional events.

As expected stations with few data samples such as NRA0, and to some extent also SPA0, perform worst (Fig. 7). This mostly affects recall, while precision remains more stable. The issue is therefore mainly missing predictions of true arrivals rather than falsely predicting noise or mixing *P* and *S* labels. This shows that a phase detection model cannot always simply be transferred to those stations which are not well represented during training since each

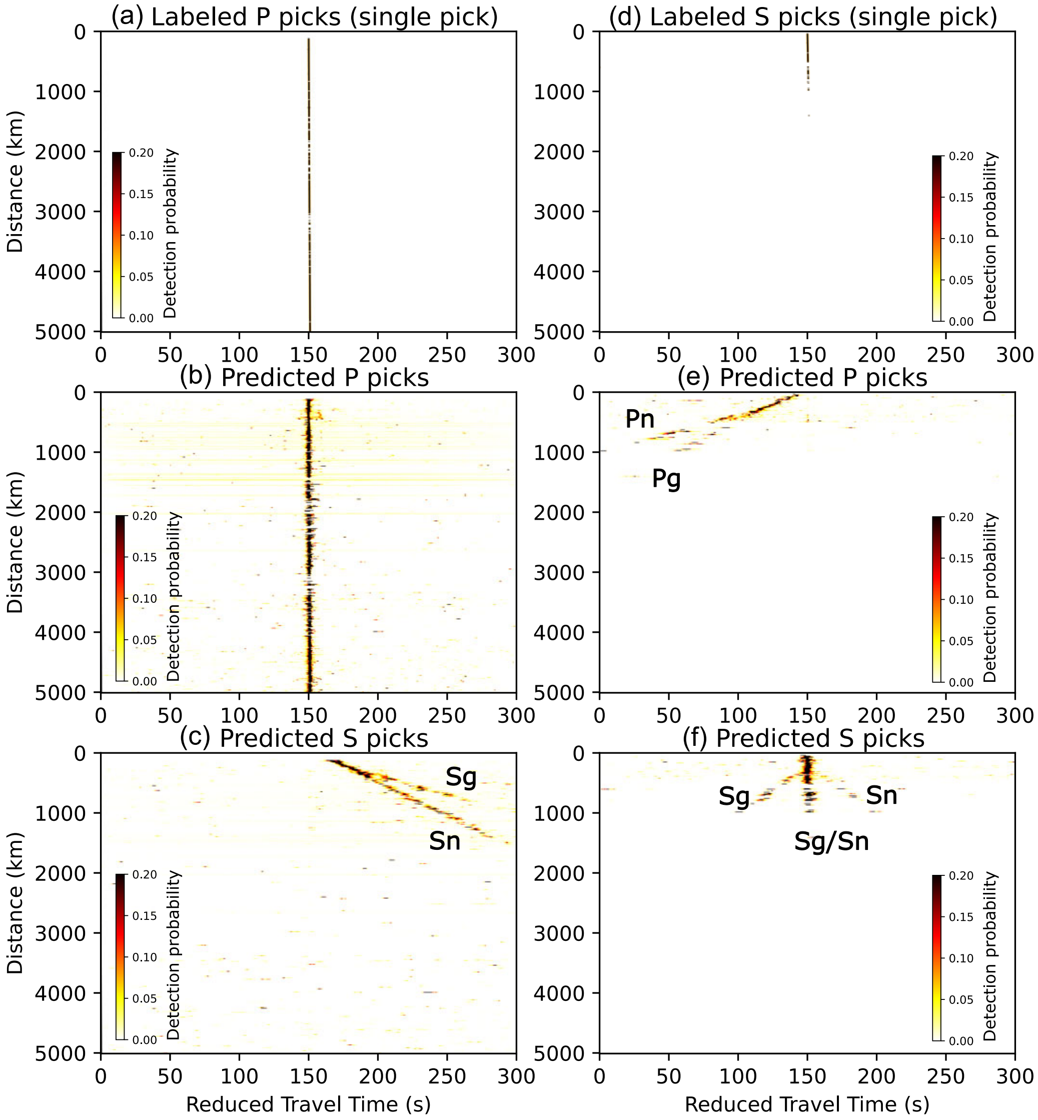

Figure 8. *P* and *S* wave output probabilities predicted by TPhaseNet for the test data set waveforms sorted by distance and stacked in 10 km bins. White is low and black is high probability. Colour scale is saturated at probability = 0.2 to enhance visibility. Time axis represents the reduced traveltime using a velocity of 7 km s^{-1} for P (a–c) and 3.5 km s^{-1} for *S* waves (d–f). Panels (a) and (d) are the analyst labels for events with a single *P*- or *S*-wave pick (Gaussian probability distribution around picks). b) and e) are predicted P arrivals for both cases. Panels (c) and (f) are predicted *S* arrivals for both cases. Text indicates traveltime branches.

site has its specific characteristics. Furthermore, there is a clear performance drop for underrepresented long distances for *S* waves, as well as for *P* waves in the 0–100 km distance range. Interestingly, *S*-wave detection seems not to be much affected in the latter range. We see also that *P* wave precision for arrivals at station ARA0 in the IDC LEB and the NRB are higher than for the HRB (see small drop in Fig. 7c). A possible explanation is that the HRB includes a higher number of problematic events, that is arrivals with low SNRs that are not picked by the analyst. Those events are not routinely considered for inclusion in the IDC LEB and the NRB.

4.2 Identifying missing picks in the training data

As mentioned above, a possible reason for obtaining low precision computed from labels in the test data set, are missing picks in the labelled data. In other words, apparent false detections, which can in fact be real arrivals, could decrease the precision. As an alternative to manual repicking of all events, which is not an option here given the large number of events (see Table 1) and lack of human resources, we suggest a qualitative assessment of this issue. For this, we plot the *P* and *S* wave output probabilities of TPhaseNet versus epicentral distance for those events in the test data set which only received a single *P* wave pick (Figs 8a–c; 3104 of 12 594 event waveforms) or a single *S*-wave pick (Figs 8 d–f; 2568 of 12 594 event waveforms) from the analyst. Probabilities are stacked in distance bins of 10 km. The results for predicted *S* picks in the case of only *P* labels in Fig. 8(c), as well as for *P* picks in the case of only *S* labels in Fig. 8(e), show branches of high probabilities, that is non-vertical lines, corresponding to the differences to the expected seismic velocities. Vertical lines correspond to reduced traveltimes using 7.0 km s^{-1} for P (Figs 8a–c) and 3.5 km s^{-1} for *S* waves (Figs 8d–f). A closer look reveals different traveltime branches for the unpicked arrivals. These correspond to

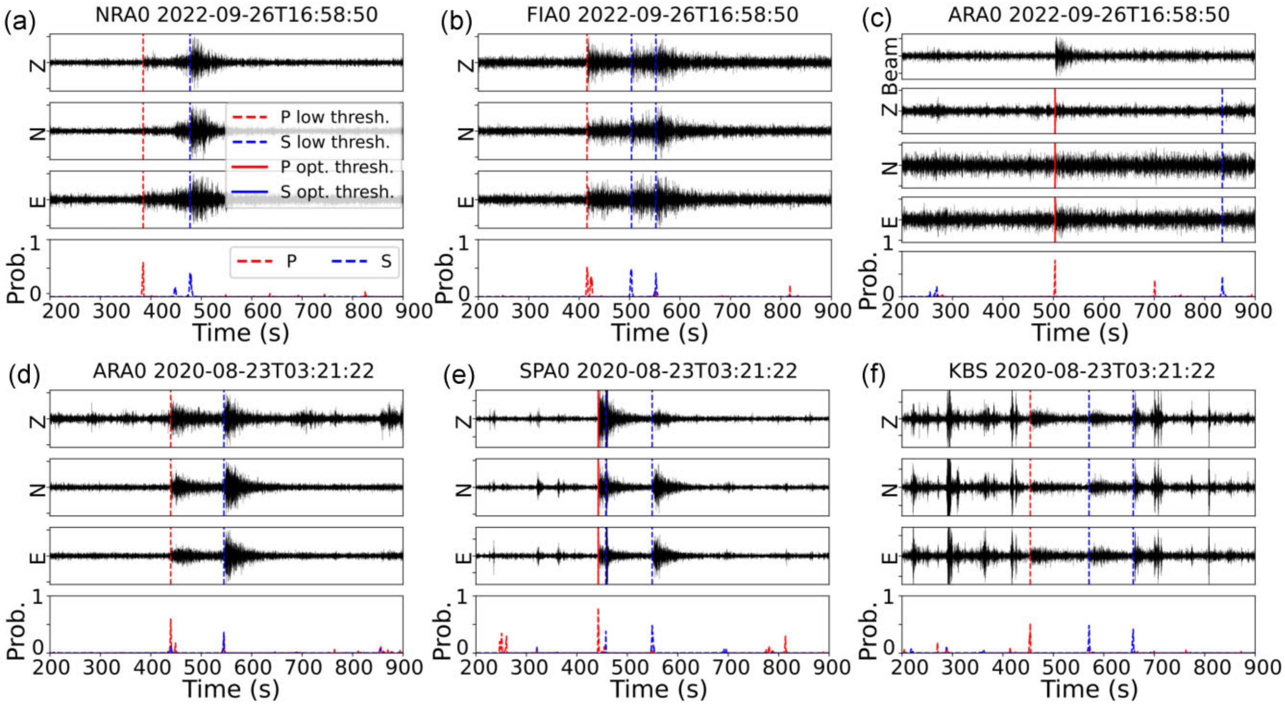

Figure 9. Three-component waveform examples filtered between 2 and 8 Hz. TPhaseNet output probabilities for *P* and *S* waves are show below. Red (*P* wave) and blue (*S* wave) vertical lines on the waveforms show when the optimal (solid lines) and lower (dashed lines) detection thresholds are exceeded. Respectively, these thresholds are 0.6 and 0.4 for the *P* wave, and 0.5 and 0.35 for the *S* wave. The legend is only given in (a) but is valid for all panels. In (c) the vertical component *P* wave beam of the ARCES array is added. (a–c) Nord Stream explosion event. (d–f) Earthquake in Novaya Zemlya region.

the regional phases Pg, Pn, Sg and Sn. Depending on their velocity, these additional predictions can occur before or after the analyst picks. The reason for the unlabelled but detected arrivals moving out of the detection time window for larger distances (see Figs 8c and e), is that the training and test data waveforms are centred on the first arrival if only one arrival is labelled. This will not happen in continuous processing, which is introduced in the next sections. Our results give clear indications that many predicted arrivals not matched by analyst-labelled picks are in fact correctly detected by TPhaseNet. This implies that precision is indeed underestimated in our results above.

4.3 Prediction results for event examples

In the following, we use TPhaseNet, which is the best performing model, given the quantitative assessment above. In contrast to the previous evaluation, we now produce the phase detection probabilities by predicting with a sliding time window, that is we use a segment of continuous data which is longer than the model input (5 min). We use a 15-min-long record centred around each event and choose an overlap of 10 s between the 5-min-long time windows. From all sliding windows we compute mean, median, standard deviation and 25 per cent percentile of the output probabilities for each time sample. In the following figures the median probability is shown, which we found to produce the best results when averaging the probabilities. None of the examples is included in the training and validation data set.

Fig. 9 shows examples of two seismic events of special interest which a detector should not miss, that is the waveforms of the explosion event at the Nord Stream pipeline in the Baltic Sea on 26.09.2022 recorded on stations NRA0, FIA0 and ARA0 (Köhler *et al.* 2023; Köhler & Myklebust 2023) and an earthquake in the Novaya Zemlya region, the location of a former Soviet nuclear weapon test site, recorded on stations ARA0, SPA0 and KBS. KBS is a Global Seismic Network station in Svalbard which is not included during model training. The output probabilities for *P* and *S* waves are shown, and the time samples above different detection thresholds are indicated in the waveform panels. We use the optimal thresholds derived above and in addition lower thresholds, that is 0.4 for *P* waves and 0.35 for *S* waves. As the following examples will show, although the lower values increase the false detection rate as expected, we miss a few true arrivals with the higher threshold.

In the case of the explosion event in Figs 9(a)–(c), *P* and *S* waves are detected for the two closest stations using the low threshold. Interestingly, the *P* wave is detected also at ARA0 with high probability, even though it is hardly visible in the single station waveforms. The ARCES array beam for the *P* wave on the other hand clearly reveals the presence of the arrival. In the case of the Novaya Zemlya earthquake in Figs 9(d)–(f) again all *P* and *S* wave arrivals are detected with a low threshold setting. The SPITS array as well as KBS are known for frequent cryogenic seismic events, that is calving (Köhler *et al.* 2015, 2022) and frost quakes (Romeyn *et al.* 2022), which are clearly visible as transient signals in the waveform plots. Notably, TPhaseNet does not pick these events, except of one *S* pick at KBS. For KBS the performance is impressive, particularly since *P* and *S* arrivals have lower SNRs than the calving events before and after, and KBS data were not used during training of the model. As for the Nord Stream event, however, the optimal threshold we derived from the test data set must be decreased to detect all arrivals of the earthquake.

Figs 10 and 11 show selected events from the NRB and HRB randomly distributed throughout 2022 and 2023. The events in

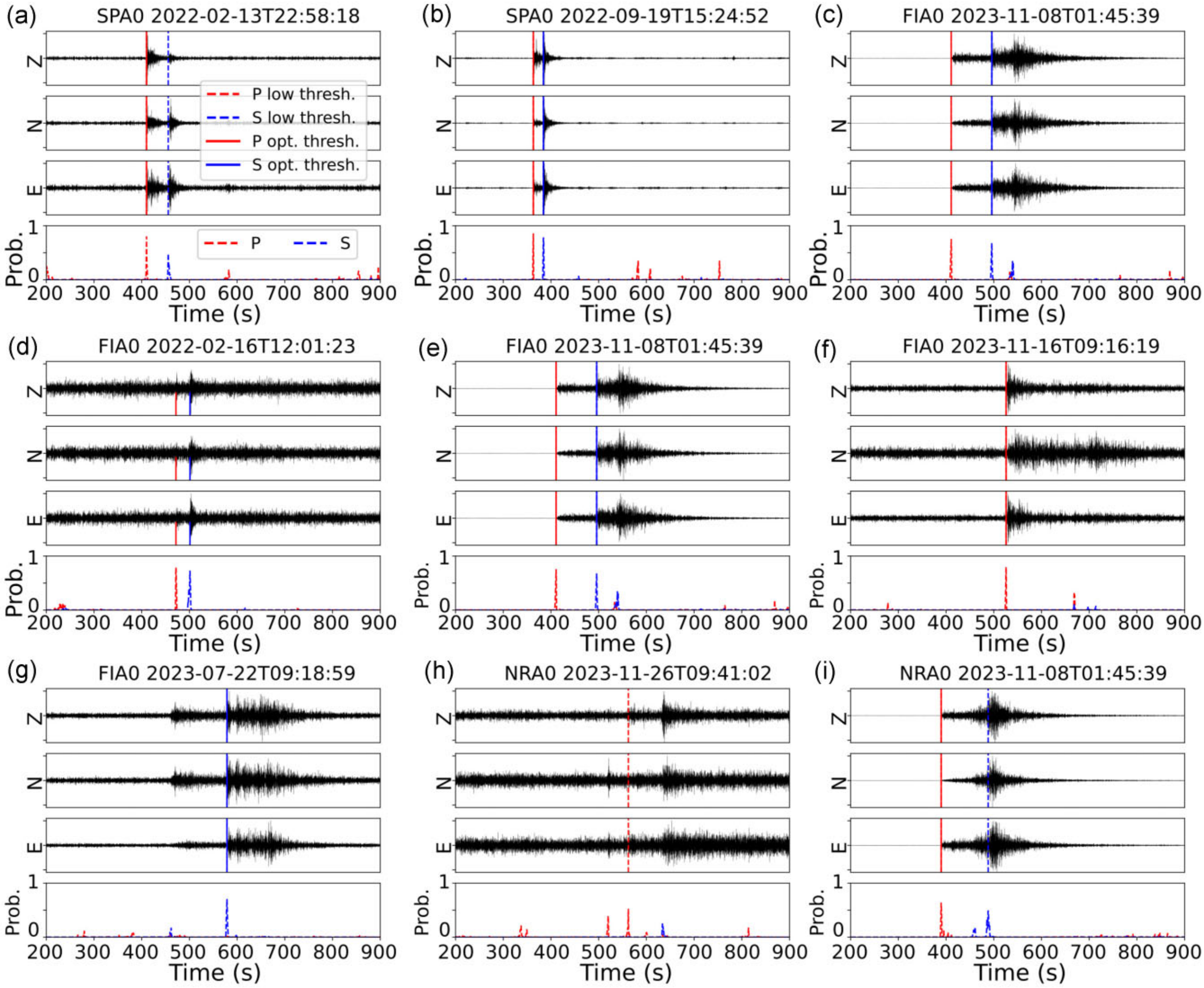

Figure 10. Examples of TPhaseNet results for event waveforms not included in the training data. Examples are shown for FIN0, SPA0 and NRA0. The three-component waveforms are filtered between 2 and 8 Hz.

Figs 10(a)–(c) and e observed on stations SPA0 and FIA0 have high SNRs. *P* and *S* wave are correctly detected using the optimal threshold setting. The event in Fig. 10(d) on station FIA0 is very weak but real. Both arrivals are recognized using the optimal threshold. The same holds for the *P*-wave arrival in Fig. 10f. However, here the *S* wave is hardly visible and not recognize by TPhaseNet. Furthermore, it is not clear why the *P* wave is missed for the next events on station FIA0 (Fig. 10g). However, in both cases the probabilities of the correctly detected phases are high. Fig. 10(h) shows another event with low SNR, this time on station NRA0. The model detects the *P* wave with a low threshold, but misses the *S* wave, although it is visually clearer. Finally, we added an example from station NRA0 where the S wave is detected with low probability (Fig. 10i). While the overall performance of the model seems to be good, we also conclude that threshold setting is critical and low SNR, not surprisingly, makes it more challenging for TPhaseNet to detect arrivals.

Fig. 11 shows only events observed on station ARA0. The challenge at the ARCES array is that often multiple events, mostly mining induced events, are observed within the same input time window of the phase detection model. First, we therefore focus on how TPhaseNet deals with those cases (Figs 11a–c,e). In Fig. 11(a) two *P* and two *S* waves with low SNRs are detected, the phase arrivals corresponding to the second event hardly being visible in the record of station ARA0. The vertical component beam computed using back-azimuth and velocity derived for the first P wave, clearly reveals two events from a mine in Sweden about 340 km to the South of the ARCES array. Although the second event is only detected using the low thresholds, this underlines the impressive performance of TPhaseNet. If ARA0 would not have been part of an array, this event would have most likely been missed by conventional STA/LTA detection algorithms because of the low SNR. Fig. 11(b) shows records that include three events. The first one is the weakest and TPhaseNet does not detect the *P* wave. However, all other arrivals are recognized with high probability. Multiple, partially overlapping events are shown in Fig. 11(c). Even for an analyst it would be a challenging task to associate the *P* and *S* waves correctly in this case. We observed at least four different events, the first two overlapping. When using a low threshold setting, TPhaseNet is able to detect all 4 *P* waves and the *S* waves of the last two events. However, it confuses or misses *S* waves for the overlapping events. In the last example with multiple events in (Fig. 11e), we are able to confirm a detected *P* wave which follows a clear event with *P* and *S* arrivals in the middle of the time window. Figs 11(d)

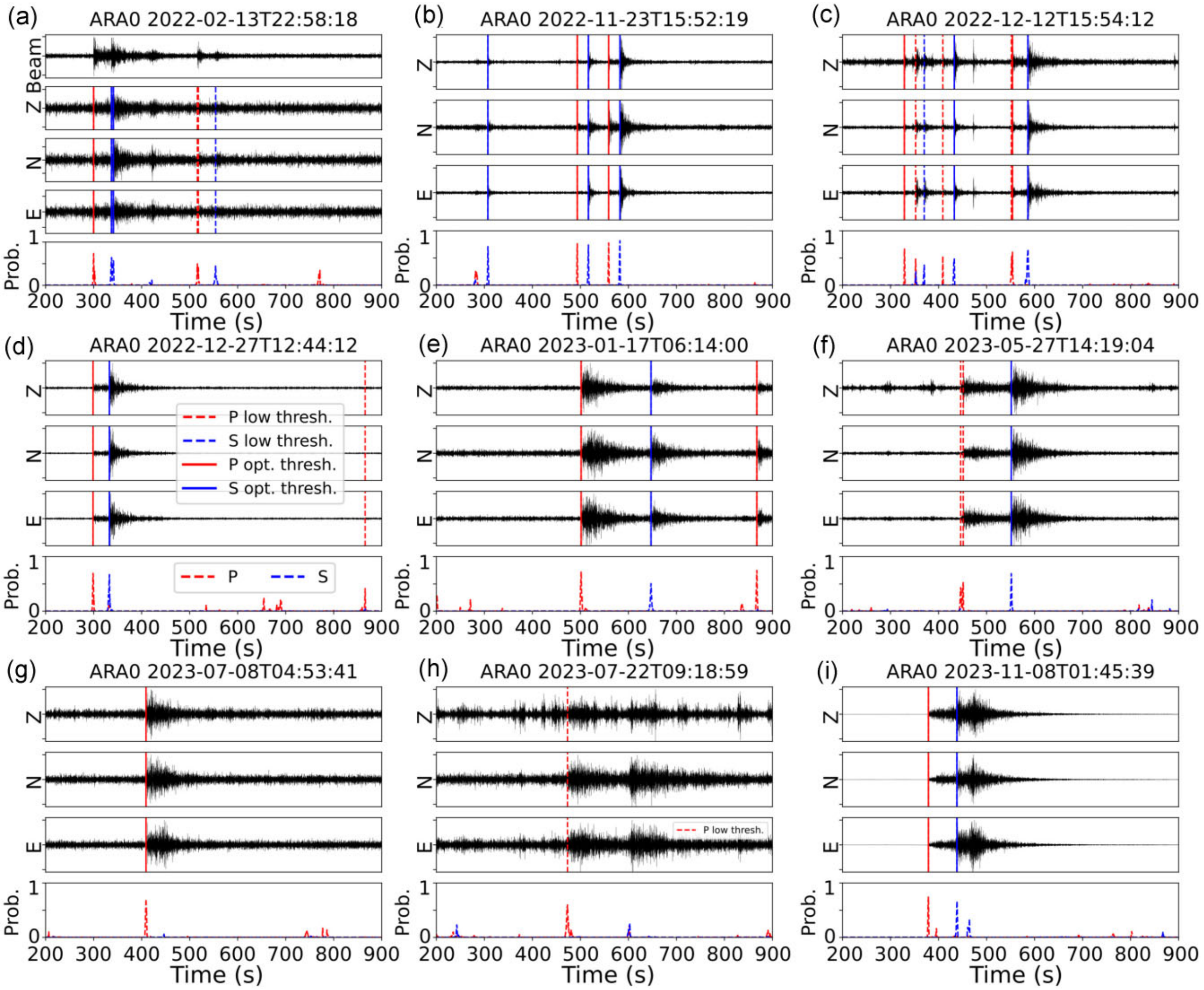

Figure 11. Examples of TPhaseNet results for events recorded at ARA0, which are not included in the training data. In (a) the vertical component *P* wave beam of the ARCES array is added (filtered 3–10 Hz for optimal signal-to-noise ratio).

and (i) show events with high SNRs and high phase detection probabilities. For events with lower SNR, arrival detection requires a lower threshold (Fig. 11f). In Fig. 11(h) we see an event with even lower SNR where the *S* wave arrival is not detected. Finally, we show an example of an event where only the visible *P* wave is detected and the *S* wave is neither detected nor visible within the strong *P* wave coda (Fig. 11g). Overall, we can conclude that TPhaseNet handles multiple events well, but that, as described above, low SNRs make it more challenging for the model to detect arrivals with high probability.

4.4 Prediction results for continuous data

Next, we have a closer look at the TPhaseNet model performance when processing longer records of continuous data. The focus is on recall and precision for all phase arrivals observed at station ARA0 of the ARCES array. We chose this station since a large variety of events (earthquake and mining signals) with locations in the Arctic and down to Southern Scandinavia are observed at ARCES due to its central location in our study region. For the precision metric it is important not to rely solely on the reviewed event bulletins because weak events observed only at the ARCES array are often not included. Therefore, we do not use bulletin information but instead visually screen 4 d of continuous data starting on January 1st in 2023 to manually pick all *P* and *S* phase arrivals associated with seismic events. We use individual selected stations from the ARCES array as well as the time series of maximum array beampower to guide the picking. This provides us with a completely labelled data set for testing TPhaseNet. We encounter many arrivals coherently observed on all array stations which are not clearly associated with an event, that is a clearly associated *P*–*S* wave pair. We pick those 367 picks as a third category of unknown type in addition to 244 *P* and 260 *S* waves. Even though our model is only trained with ARA0 data from ARCES (in addition to stations from other arrays), we also apply it to additional station elements from the array to test the generalization of the model: ARA1 (170 m distance from ARA0), ARA2 (146 m distance) and ARC3 (290 m distance).

For further evaluation, we calculate SNRs for all picked arrivals by dividing the maximum absolute amplitude of each arrival by the average absolute noise amplitude just before the arrival on the vertical component in two different frequency bands (2.5–10 Hz and 5–20 Hz). The maximum SNR of both bands is then used for

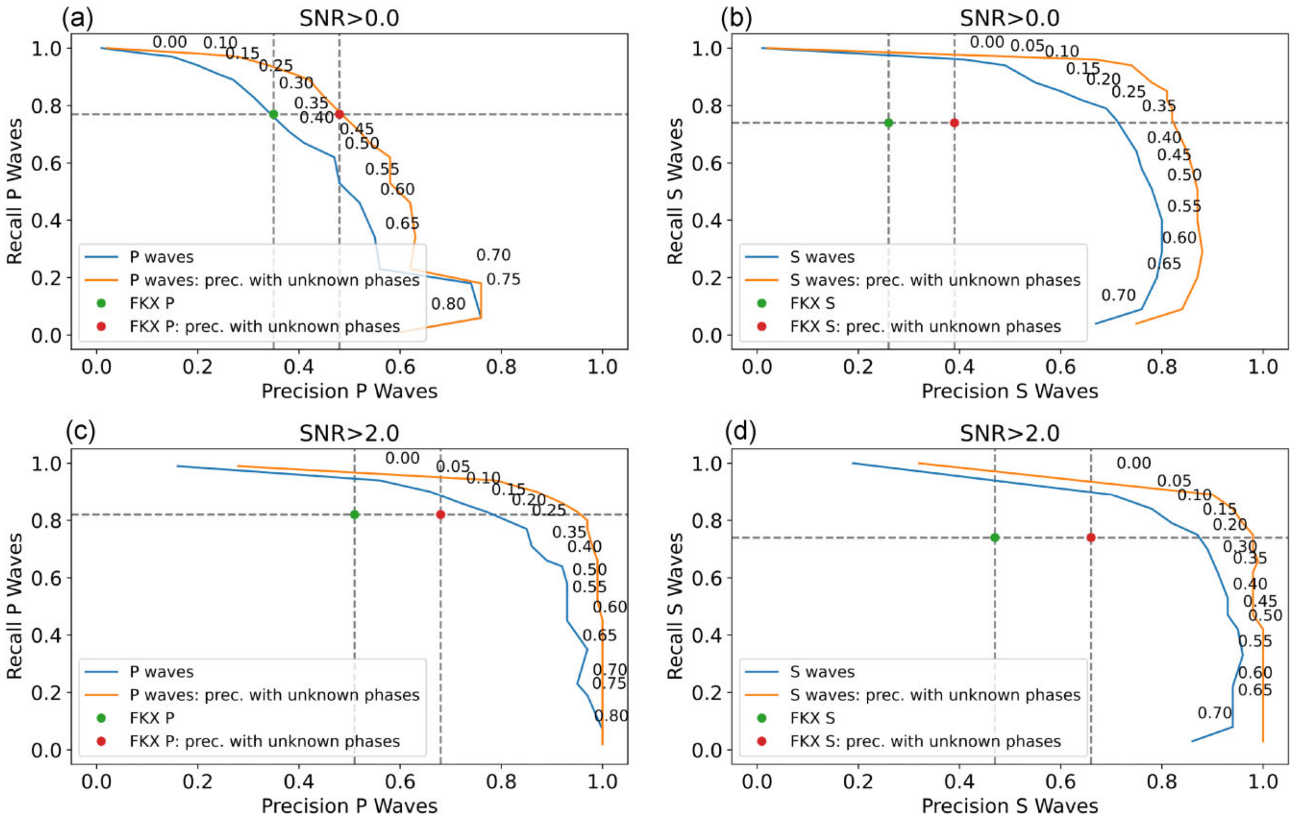

Figure 12. Performance metrics of TPhaseNet at station ARA0 for different decision thresholds (shown as number along the curves), without (a and b) and with SNR threshold (c and d). The closer the displayed curves bend towards the upper-right-hand corner, the better the performance. Recall and precision for FKX phase detection in operation at ARCES are added as symbols (no detection threshold data available). For each precision and recall pair a second one is added where arrivals labelled as 'unknown' are counted as either *P* or *S* waves for calculating precision.

further analysis. Note that many of the picked arrivals have very low SNRs, that is 80 per cent have SNRs smaller than 5.0. For ARC3 the percentage is even higher since this station has the highest noise level of all ARCES stations in the considered time period. Therefore, in addition to computing performance metrics for all picked arrivals, we also provide results for using a SNR threshold of 2.0 to focus only on the clearer signals. As in the previous section, we also use different probability thresholds for computing the performance metrics and producing the recall-prediction curves. In addition, to compare our results with an established detector, we compute metrics also for NORSAR's existing array processing at ARCES (Schweitzer *et al.* 2012), which produces phase detections from array beams (Detection Processing: DPX), measures back-azimuth and apparent velocity, and labels arrivals (Signal Attribute Processing or F-K Analysis Processing: FKX). Note that we increase the time difference tolerance for matching predictions and picked arrivals from 2 to 5 s to take into account that FKX arrivals may be less accurately picked than manual picks.

Fig. 12 shows a summary of all results for station ARA0, while the results for stations ARA1, ARA2 and ARC3 can be found in the Appendix (Figs A3, A4 and A5). The blue curves show that performance in terms of precision when using all detections and manual picks is modest. While high recall can be achieved with a low detection threshold, the precision lays only between 0.4 and 0.7, that is a considerable amount of false detections are obtained. However, imposing a low SNR limit on predictions and picks improves results considerably. We obtain recalls above 0.9 and higher precision values above 0.8. Counting the unknown arrival picks as *P* or *S* waves, respectively, when computing precision, further improves performance (orange curves). The reasoning behind the latter is that a post-classification for all detected arrivals could be done in the case of array processing, correcting for *P* waves classified as *S* or vice versa. Overall, TPhaseNet performs very well, given the fact that many arrivals have amplitudes just above the noise level.

Interestingly, TPhaseNet performs similarly for *P* wave detection when no SNR threshold is applied, and seems to outperform the existing array detector for *S* waves and higher SNRs (FKX, Figs 12a and b). However, note that no data for different detection thresholds are plotted for the FKX results, since we simply use the output of the detector in continuous operation at NORSAR. Hence, only two data points are added in each panel of Fig. 12. For detection thresholds achieving similar recall values for TPhaseNet and FKX, the precision of TPhaseNet is much higher. Likewise, FKX has a lower recall for detection thresholds that achieve similar precision for FKX and TPhaseNet. The reason for the latter is that FKX arrivals, although correctly detected, are often miss-classified or dismissed due to low SNR. Fig. 13 shows an example of a weak seismic event, the *P* wave hardly visible on the single station, where P and S waves are correctly picked by TPhaseNet with low thresholds (0.4 for *P* and 0.35 for *S* wave) but FKX only detected the S wave. The ARCES P wave beam is shown in addition, verifying the presence of a *P* wave, and also a second event shortly after. We confirmed the same backazimuth and expected velocities for both events by F-K analyses on ARCES. As seen in the output probabilities, both arrivals of

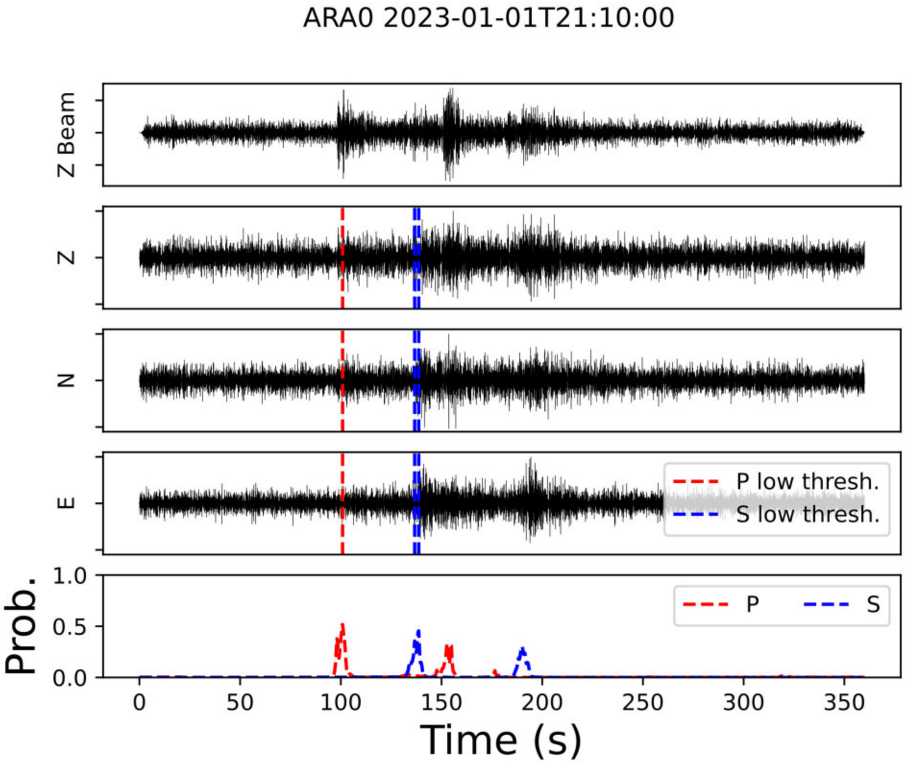

Figure 13. Example of TPhaseNet applied to three-component waveforms from ARA0, filtered between 3 and 8 Hz for optimal signal-to-noise ratio. TPhaseNet detection probabilities are shown below the waveforms, with vertical lines showing the *P*-wave detection (red) and *S*-wave detection (blue) for thresholds of 0.4 and 0.35, respectively. For reference, the top panel shows the vertical component *P*-wave beam from the ARCES array. Only the first *S*-wave is detected by the standard FKX detector for this event.

the second event would also have been detected by TPhaseNet with a lower threshold.

Performance is more variable for the additional ARCES stations. Although none of those stations was used during model training, a significant drop in performance is only observed for ARC3 (Fig. A5). This is due to the higher noise level at this particular element of the ARCES array during the test period.

5 DISCUSSION

We present results for seismic phase prediction for regional events using three state-of-the-art and one newly developed machine learning method. Overall, the new architectures of TPhaseNet outperforms the selected baseline models with respect to all performances metrics, that is recall, precision and pick residuals. This demonstrates the power of attention and transformer mechanisms which improve the already popular PhaseNet model. For our regional data set, we observe that EQTransformer does not perform as well, that is we obtained worse performance metric values compared to the other models (see Figs 6, 7 and A2). This would indicate that skip-connections are highly important for phase picking at regional distances. However, it must be emphasized that our model comparison here is valid for events in a specific region and for selected stations. Future studies to establish the best deep learning method for regional phase picking could include extended data sets such as the CREW data set (Aguilar Suarez & Beroza 2024), and modified versions or training strategies of the existing baseline models (Park *et al.* 2024), which could be adapted to regional data. The CREW data was not yet available during our study.

In general, hyperparameters can have a huge impact on the performance of machine learning models. However, here we did not perform an extensive study of hyperparameter settings, although we did experiment with different model sizes. We focus on the new TPhaseNet architecture, and therefore, we chose the same parameters for all models. In this way, we do not bias one method over another due to imperfect hyper-parameter choices. More tuning studies are however recommended in order to find the model best suitable for phase picking in regional event records.

There are multiple approaches to improve our model and to generalize its applicability which are beyond the scope of this study. Extension of the model to teleseismic arrivals would be beneficial not only for generating global seismicity catalogues, but also for including these signals in the training data sets for regional monitoring. For example, detecting teleseismic *P* wave arrivals as a separate class can help to distinguish those from regional phases. Curating a training data set from global event bulletins, such as the IDC LEB or the ISC Bulletin (International Seismological Centre 2024), would be required.

Furthermore, using all available waveform data from stations regardless of sensor upgrades or replacements would increase the training data set. In our study, this concerns SPA0, NRA0 and FIA0, where we only use data from the sensors currently in operation, since those are most relevant for real-time processing. Removing the instrument response would be the obvious solution, however, this would also increase preparation time of training data. Nevertheless, since normalized waveforms are used as model input, a changing instrument response is not expected to affect model performance much in our case since we filter in a frequency band where the response in flat. Furthermore, having waveforms of the same station from different sensors may help the model to generalize phase picking at unseen stations.

In this study we focus on using single three-component stations from seismic arrays. Increased generalization ability would benefit from adding more single stations in the region to increase the training data set. However, we want to emphasize that we already tested TPhaseNet for stations not used for training with encouraging results (KBS and additional elements of the ARCES array). Adding more stations would not solve the issue related to the unbalanced data set, that is that some stations (ARA0) dominate the training data set. Augmenting training data for underrepresented stations is an option to be explored. Here, we only use augmentation as described above which increases the training data for each station by the same percentage. Using station-dependent detection thresholds may partially compensate for the issue of bad performance at underrepresented stations. For regions with low seismicity, unbalanced data is a general challenge when training regional phase detectors. Hence, future models will most likely be trained with combined regional event data sets from different globally distributed networks such as the novel CREW data set (Aguilar Suarez & Beroza 2024).

As discussed in Section 2, there are shortcomings with the event bulletin sources that we used to create the training data set. Ensuring completeness, that is that all observed phase arrivals are picked, is challenging and would require manually reviewing all events, or some sort of iterative retraining of the model with verified automatic picks from an initial prediction on the training data. We have already demonstrated qualitatively how unlabelled but real predicted arrivals can be identified in the test data. This implies that the actual model precision is higher than our computed metrics suggests, and increases therefore confidence in the performance of TPhaseNet. Wrong picks should be rare in reviewed bulletins, but cannot be excluded. Simply using a SNR threshold before including arrival picks in the training data is one option. However, doing so introduces the risk of loosing genuine low SNR arrivals which the model needs to be trained on for good performance on new data. Hence, incompleteness as mentioned above would again become an issue. Again, a complete and curated dateset such as CREW can help to overcome these issues in future studies.

In this study, we decided to pre-filter the waveforms in the frequency band that we think is most suitable for regional monitoring in our study area. The original implementation of PhaseNet for local events did not include this preprocessing step, since a convolutional network has the ability to learn appropriate filters implicitly. Our main reason for pre-filtering the waveform data for training the models used in this study is the limited size of the training data set, which favours smaller model sizes. However, for a future study it would be interesting to test the performance of models trained with unfiltered data.

It can be argued that the performance metric values for TPhaseNet obtained here should be higher, that is at least above 0.9 for recall and precision, so that it can be deployed for operation. However, we showed that the obtained precision is most likely underestimated when computed from the event test data set, due to missing labels. Furthermore, we want to emphasize again that we deal with a more complex data set than previous studies that focused on local events only. Regional event data in our study area are unbalanced when it comes to backazimuth, epicentre distances and station coverage (Fig. 1). Overlapping events and low SNRs are common (see examples in Figs 9–11). Established automatic detection methods have to deal with the same issues. In this context it is intriguing that we have found evidence that the existing array processing for ARCES does perform similarly for *P* waves and worse for *S* waves compared to TPhaseNet for low SNR arrivals. However, this needs further investigation beyond the 4 d of continuous data that were tested, and with the changes to the array processing detection threshold also evaluated.

Since we see the benefit of using machine learning for a single station of an array, the obvious next step is to develop array processing including deep learning phase pickers. We expect that doing so will improve performance considerably by increasing both precision (false signals only detected at a few stations can be eliminated) and recall (SNR can be decreased by beamforming). This would allow us to use a lower detection probability threshold than the optimal threshold suggested by the recall-precision curves, a need we have already seen when evaluating event examples visually. We hypothesize that a low threshold will increase the recall, and array processing could remove false detections. There is also the option to combine the phase detection with a subsequent additional machine learning and array data-based phase classification step using the method suggested by Köhler & Myklebust (2023). Where no array data are available, sorting out false arrivals can also be achieved by multiple station processing (phase association), that is by identifying detections not associated with any event observed on a seismic network.

Our training data includes analyst picks of Pg/Sg, Pn/Sn or unclassified *P*/*S* waves. Mai *et al.* (2023) provide an option to include these regional arrivals as separate classes in the deep learning phase picker. However, in our case, analysts tend to only pick the first *P* arrival, either the direct or the Moho-refracted head wave. For *S* waves the second arrival (Sg) is often preferably picked, even beyond the cross-over distance due to higher SNR. Hence, not all arrivals are picked for all events, and it is difficult to provide a training data set to classify both categories. For a phase picking model to learn to discriminate between regional phase arrivals, it needs to be presented with enough event samples with all arrivals being picked consistently. A possibility to overcome this issue in future could be to add theoretical arrival time picks. At this point, however, our method includes only a two-class model, that is *P* and *S* waves, where the model tends to detect the first *P* arrival (either Pg or Pn) and the *S* wave with higher SNR (Sg or Sn). Nevertheless, we found examples where *P* wave probability exhibits two peaks, coinciding with Pg and Pn arrivals. This shows that our model is not biased towards only allowing a single *P* wave to be picked for each event. In future detection pipelines, distinguishing between regional phase types can be done by array processing after the initial detector stage (ArrayNet, Köhler & Myklebust 2023), or by the event association algorithm. In the case of ArrayNet, the labelled phase arrivals are independent samples in the training data set. Therefore, it does not matter if events exhibit missing picks.

6 CONCLUSION

We have developed a modified deep learning method based on the PhaseNet architecture for seismic phase detection in regional event waveforms, a task for which deep learning has yet to be applied. We called the method TPhaseNet since it adds transformer layers to the neural network. The training data set includes phase arrivals obtained from three different reviewed event bulletins. Five-min-long three-component waveforms from single stations of four seismic arrays are used as input. We evaluated the performance of TPhaseNet and compared it to three state-of-the-art models for phase detections.

We found that our newly proposed model architecture outperforms the baseline models when tested on unseen seismic event records. In a final test, we apply TPhaseNet to continuous data from stations of the ARCES array. A comparison of classification metrics showed that our new method increases detection rate and decreases the number of false detections with respect to the existing array detector at ARCES. These results were obtained with only 4 d of data, and we therefore suggest more investigations in future studies focusing on integrating deep learning into array processing pipelines. This would allow operating TPhaseNet with a lower detection threshold and let array processing remove false signals. Future studies should also include more model evaluation, including hyper-parameter tuning as well as retraining with an improved, that is larger and more balanced training data set.

Overall, our study is a successful and crucial first step towards integration of machine learning into the regional event detection pipeline at NORSAR, with future implications for other data centres including the CTBT verification with IMS stations. Furthermore, we are confident that phase detection using deep learning models has the potential to replace the STA/LTA trigger currently in use in the automatic array processing.

ACKNOWLEDGMENTS

We thank one anonymous reviewer, Natalia Poiata and Ben Dando for their very helpful comments and suggestions, and Brandon Schmandt whose idea of plotting phase pick probabilities versus distance inspired us to do the same. We thank Karoline Aastrup-Köhler for spellchecking the paper. We acknowledge funding from AFRL.

DATA AVAILABILITY

Seismic data processing was partly done using Obspy (Beyreuther *et al.* 2010). Fig. 1 was generated using the Generic Mapping Tools (Wessel & Smith 1995). ARCES and SPITS waveform data are available via IRIS (Albuquerque Seismological Laboratory (ASL)/USGS 1988) (https://doi.org/10.7914/sn/iu) or the Norwegian EIDA node (Ottemöller *et al.* 2021) (https://www.orfeus-eu.org/data/eida/nodes/UIB_NORSAR/). All data are stored at NORSAR (NORSAR 1971b) (https://doi.org/10.21348/d.no.0001). Reviewed seismic event bulletins are available from the Finnish National Seismic Network (Institute of Seismology 1980a, b; Veikkolainen *et al.* 2021) and from NORSAR (NORSAR 1971a). All IDC products (waveform data of FINES and IDC LEB) can be requested via the CTBTO vDEC system (https://www.ctbto.org/specials/vdec/). The code for model training and application is available at https://github.com/NorwegianSeismicArray/tphasenet. The trained models and part of the training data set can be downloaded from https://www.doi.org/10.5281/zenodo.11231543 (Köhler & Myklebust 2024).

REFERENCES

Aguilar Suarez, A. L. & Beroza, G., 2024. Curated regional earthquake waveforms dataset, *Seismica,* **3** (1), 1–17.

Albuquerque Seismological Laboratory (ASL)/USGS, 2014. Global seismograph network (GSN - IRIS/USGS) [Data set], International Federation of Digital Seismograph Networks, https://doi.org/10.7914/SN/IU.

Bahdanau, D., Cho, K. & Bengio, Y., 2016. Neural machine translation by jointly learning to align and translate, https://doi.org/10.48550/arXiv.1409.0473.

Bai, C.-Y. & Kennett, B., 2000. Automatic phase-detection and identification by full use of a single three-component broadband seismogram, *Bull. seism. Soc. Am.,* **90**(1), 187–198.

Bergen, K.J., Chen, T. & Li, Z., 2019. Preface to the focus section on machine learning in seismology, *Seismol. Res. Lett.,* **90**(2A), 477–480.

Beyreuther, M., Barsch, R., Krischer, L., Megies, T., Behr, Y. & Wassermann, J., 2010. ObsPy: a Python toolbox for seismology, *Seismol. Res. Lett.,* **81**(3), 530–533.

Chen, J. *et al.*, 2021. TransUNet: transformers make strong encoders for medical image segmentation, https://doi.org/10.48550/arXiv.2102.04306.

Chollet, F. *et al.*, 2015. Keras, https://github.com/fchollet/keras.

García, J.E., Fernández-Prieto, L.M., Villaseñor, A., Sanz, V., Ammirati, J.-B., Díaz Suárez, E.A. & García, C., 2022. Performance of deep learning pickers in routine network processing applications, *Seismol. Soc. Am.,* **93**(5), 2529–2542.

Institute of Seismology, 1980a. The Finnish National Seismic Network [Data set], https://www.seismo.helsinki.fi/bulletin/list/norBull.html.

Institute of Seismology, 1980b. The Finnish National Seismic Network, https://doi.org/10.14470/SA879454.

International Seismological Centre, 2024. On-line Bulletin, https://doi.org/10.31905/D808B830.

Kalinowski, M.B. & Mialle, P., 2021. Introduction to the topical issue on nuclear explosion monitoring and verification: scientific and technological advances, *Pure appl. Geophys.,* **178**(7), 2397–2401.

Köhler, A. *et al.*, 2023. Relative locations and moment tensors of the nord stream pipeline events, EGU General Assembly 2023, Vienna, Austria, 24–28 Apr 2023, EGU23-7019, https://doi.org/10.5194/egusphere-egu23-7019.

Köhler, A. & Myklebust, E.B., 2023. ArrayNet: A combined seismic phase classification and back-azimuth regression neural network for array processing pipelines, *Bull. seism. Soc. Am.,* **113**(6), 2345–2362.

Köhler, A. & Myklebust, E.B., 2024. Phase picker models and training data for paper "Deep learning models for regional phase detection on seismic stations in Northern Europe and the European Arctic', https://www.doi.org/10.5281/zenodo.11231543.

Köhler, A., Myklebust, E. & Mæland, S., 2022. Enhancing seismic calving event identification in Svalbard through empirical matched field processing and machine learning, *Geophys. J. Int.,* **230**(2), 1305–1317.

Köhler, A., Nuth, C., Schweitzer, J., Weidle, C. & Gibbons, S.J., 2015. Regional passive seismic monitoring reveals dynamic glacier activity on Spitsbergen, Svalbard, *Polar Res.,* **34,** 26178.

Kong, Q., Trugman, D.T., Ross, Z.E., Bianco, M.J., Meade, B.J. & Gerstoft, P., 2019. Machine learning in seismology: turning data into insights, *Seismol. Res. Lett.,* **90**(1), 3–14.

Li, W., Chakraborty, M., Fenner, D., Faber, J., Zhou, K., Rümpker, G., Stöcker, H. & Srivastava, N., 2022. EPick: attention-based multi-scale UNet for earthquake detection and seismic phase picking, *Front. Earth Sci.,* **10,** doi:10.3389/feart.2022.953007.

Mai, H., Audet, P., Perry, H.C., Mousavi, S.M. & Zhang, Q., 2023. Blockly earthquake transformer: a deep learning platform for custom phase picking, *Artif. Intellig. Geosci.,* **4,** 84–94.

Michelini, A., Cianetti, S., Gaviano, S., Giunchi, C., Jozinović, D. & Lauciani, V., 2021. INSTANCE–the Italian seismic dataset for machine learning, *Earth Syst. Sci. Data,* **13**(12), 5509–5544.

Mousavi, S.M. & Beroza, G.C., 2023. Machine learning in earthquake seismology, *Annu. Rev. Earth planet. Sci.,* **51,** 105–129.

Mousavi, S.M., Ellsworth, W.L., Zhu, W., Chuang, L.Y. & Beroza, G.C., 2020. Earthquake transformer–an attentive deep-learning model for simultaneous earthquake detection and phase picking, *Nat. Commun.,* **11**(1), 3952.

Mousavi, S.M., Sheng, Y., Zhu, W. & Beroza, G.C., 2019. STanford earthquake dataset (STEAD): a global data set of seismic signals for AI, *IEEE Access,* doi:, https://doi.org/10.1109/ACCESS.2019.2947848

Münchmeyer, J. *et al.*, 2022. Which picker fits my data? A quantitative evaluation of deep learning based seismic pickers, *J. geophys. Res.,* **127**(1), e2021JB023499, doi:10.1029/2021JB023499.

Münchmeyer, J., Saul, J. & Tilmann, F., 2024. Learning the deep and the shallow: deep-learning-based depth phase picking and earthquake depth estimation, *Seismol. Res. Lett.,* **95**(3), 1543–1557.

Ni, Y., Hutko, A., Skene, F., Denolle, M., Malone, S., Bodin, P., Hartog, R. & Wright, A., 2023. Curated pacific northwest AI-ready seismic dataset, *Seismica,* **2**(1), 1–15.

NORSAR, 1971a. NORSAR seismic bulletins, https://doi.org/10.21348/b.0001.

NORSAR, 1971b. NORSAR station network [data set], https://doi.org/10.21348/d.no.0001.

Oktay, O. *et al.*, 2018. Attention U-Net: learning where to look for the pancreas, https://doi.org/10.48550/arXiv.1804.03999.

Ottemöller, L., Michalek, J., Christensen, J.-M., Baadshaug, U., Halpaap, F., Natvik, Ø., Kværna, T. & Oye, V., 2021. UiB-NORSAR EIDA node: Integration of seismological data in Norway, *Seismol. Soc. Am.,* **92**(3), 1491–1500.

Park, Y., Delbridge, B.G. & Shelly, D.R., 2024. Making phase–picking neural networks more consistent and interpretable, *Seismic Rec.,* **4**(1), 72–80.

Romeyn, R., Hanssen, A. & Köhler, A., 2022. Long-term analysis of cryoseismic events and associated ground thermal stress in Adventdalen, Svalbard, *The Cryosphere,* **16**(5), 2025–2050.

Ronneberger, O., Fischer, P. & Brox, T., 2015. U-Net: Convolutional networks for biomedical image segmentation, in *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015. MICCAI 2015. Lecture Notes in Computer Science,* Vol. **9351**, Springer.

Schweitzer, J., Fyen, J., Mykkeltveit, S., Gibbons, S., Pirli, M., Kühn, D. & Kværna, T., 2012. Seismic arrays, in *New Manual of Seismological Observatory Practice (NMSOP-2),* 2nd (revised) edn, pp. 1–80, ed. Bormann, P., Deutsches GeoForschungsZentrum GFZ.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L. & Polosukhin, I., 2017. Attention is all you need, in *Proceedings of the 31st International Conference on Neural Information Processing Systems, NIPS'17,* pp. 6000–6010, Curran Associates Inc.

Veikkolainen, T., Kortström, J., Vuorinen, T., Salmenperä, I., Luhta, T., Mäntyniemi, P., Hillers, G. & Tiira, T., 2021. The Finnish national seismic network: toward fully automated analysis of low-magnitude seismic events, *Seismol. Res. Lett.,* **92**(3), 1581–1591.

Wang, J., Xiao, Z., Liu, C., Zhao, D. & Yao, Z., 2019. Deep learning for picking seismic arrival times, *J. geophys. Res.,* **124**(7), 6612–6624.

Wessel, P. & Smith, W. H.F., 1995. New version of the generic mapping tools, *EOS, Trans. Am. geophys. Un.,* **76,** 329–329.

Withers, M., Aster, R., Young, C., Beiriger, J., Harris, M., Moore, S. & Trujillo, J., 1998. A comparison of select trigger algorithms for automated global seismic phase and event detection, *Bull. seism. Soc. Am.,* **88**(1), 95–106.

Zhu, W. & Beroza, G.C., 2018. PhaseNet: a deep-neural-network-based seismic arrival-time picking method, *Geophys. J. Int.,* **216**(1), 261–273.

APPENDIX A: HYPER-PARAMETERS

Tables A1 and A2 contain the hyper-parameters used for the training and augmentation. *Filters* denotes the number of convolutional filters in each block, for example 8, 8, 8, 8 means that the model has four convolutional blocks, all with eight filters. The *filter size* denotes how large the filters are in each block. Most models use a default value of 7, but EQTransformer reduces the filter size based on depth in the network. We tested the impact on PhaseNet with the same filter sizes as EQTransformer and found no differences in performance. *Attention* denotes the size of each head in the multi-head attention, in each block. A value of 0 denotes no attention in that block. EQTransformer has parameters *Residual filters, residual filter sizes, LSTM filters* and *transformer sizes* and these are specific to the residual blocks, the bidirectional LSTM layers, and the size of each head in the multi-head attention and the feed-forward network in the transformers. All models use a dropout of 0.4, max pooling (size 4, stride 2), and swish activation.

Table A1. Augmentation and global model hyper-parameters.

Name	Value
Optimizer	Adam
Learning rate	0.001
Weight decay	0.01
Early stopping patience	15
Reduce learning rate patience	7
Class weights (N, P, S)	0.05,0.40,0.55
Normalization mode	Interchannel
Normalization type	Standard deviation
Add noise	0.3
Add event	0.3
Drop channel	0.2
Add gap	0.2
Max gap size	0.1
Taper	0.01

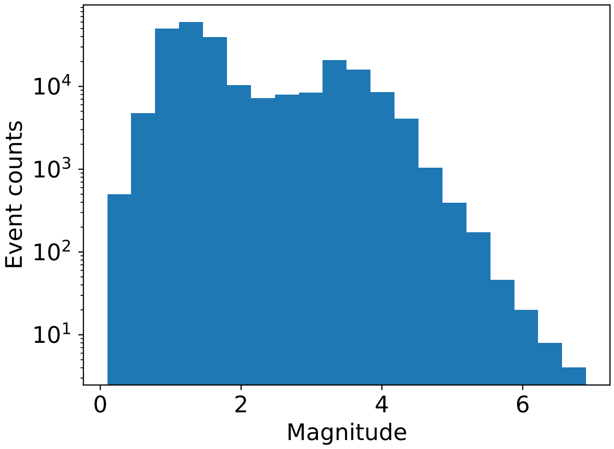

Figure A1. Magnitude distribution of seismic events used in this study.

Table A2. Model hyperparameters.

HP	PhaseNet	EPick	EQTransformer	TPhaseNet
Parameters	6.7*M*	17.2*M*	21*M*	10.4*M*
Filters	64,64,128,128,256,512	64,64,128,128,256,512	32,64,64,128,128,256,256	64,64,128,128,256,512
Filter sizes	7,7,7,7,7,7,7	7,7,7,7,7,7,7	11,9,7,7,5,5,3	7,7,7,7,7,7,7
Attention units		0,0,0,32,32,32,64		
Residual filters			256,256,256,256,256	
Res. filter sizes			3,3,3,2,2	
LSTM units			256,256,256	
Transf. units			256,256	0,0,0,32,32,32,64
Activation	`swish`	`swish`	`swish`	`swish`

APPENDIX B: MULTI-HEAD ATTENTION

Mathematically, attention is expressed as a scaled dot-product (Vaswani *et al.* 2017):

$$Attention(Q, K, V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V, \tag{B1}$$

where Q, K, V are query, key and value matrices, and d_k is the key dimension. Query is the embedded input sequence, that is a sequence of feature vectors which in natural language processing can represent a sequence of words (a sentence) and in our case represents the event waveform time window. The key is another sequence of feature vectors. The query is then compared to each feature vector (word or part of seismogram) in the key via dot product to find the most relevant part of the sequence (word or part of seismogram). To enhance the impact of the most important part, the value matrix is used to compute the final attention vector. In general, key and value are trainable but often self-attention is used where $Q = K = V$. In EPick $K = V$ which is the output of the previous convolutional block (dotted green arrow in Fig. 3), while Q is the output of the current convolutional block (solid green line in Fig. 3).

Attention is powerful in itself, but throughout this study, we use multihead attention (Vaswani *et al.* 2017). Multihead attention in neural networks can be understood as having multiple sets of 'eyes' or attention mechanisms looking at different aspects of the input data simultaneously. By doing this, the model can capture diverse patterns and relationships within the data, enhancing its ability to understand and represent complex information. Multihead attention is formulated as

$$MultiHead(Q, K, V) = Concat(head_1, head_2, \ldots, head_h)W_0, \tag{B2}$$

where $head_i = Attention(QW_i^Q, KW_i^K, VW_i^V)$, (B3)

where $W_i^Q \in \mathbb{R}^{d_{\text{input}} \times d_k}$, $W_i^K \in \mathbb{R}^{d_{\text{input}} \times d_k}$, $W_i^V \in \mathbb{R}^{d_{\text{input}} \times d_v}$ and $W_0 \in \mathbb{R}^{hd_v \times d_{\text{input}}}$. The value of d_k and d_v are typically chosen as d_{input}/h, keeping the computational cost the same as scaled dot-product attention.

APPENDIX C: SUPPLEMENTARY PLOTS

Fig. A1 shows the magnitude distribution of seismic events used in this study. Fig. A2 shows pick time residuals for all tested models. Figs A3, A4 and A5 show TPhaseNet performance for three different stations of the ARCES array.

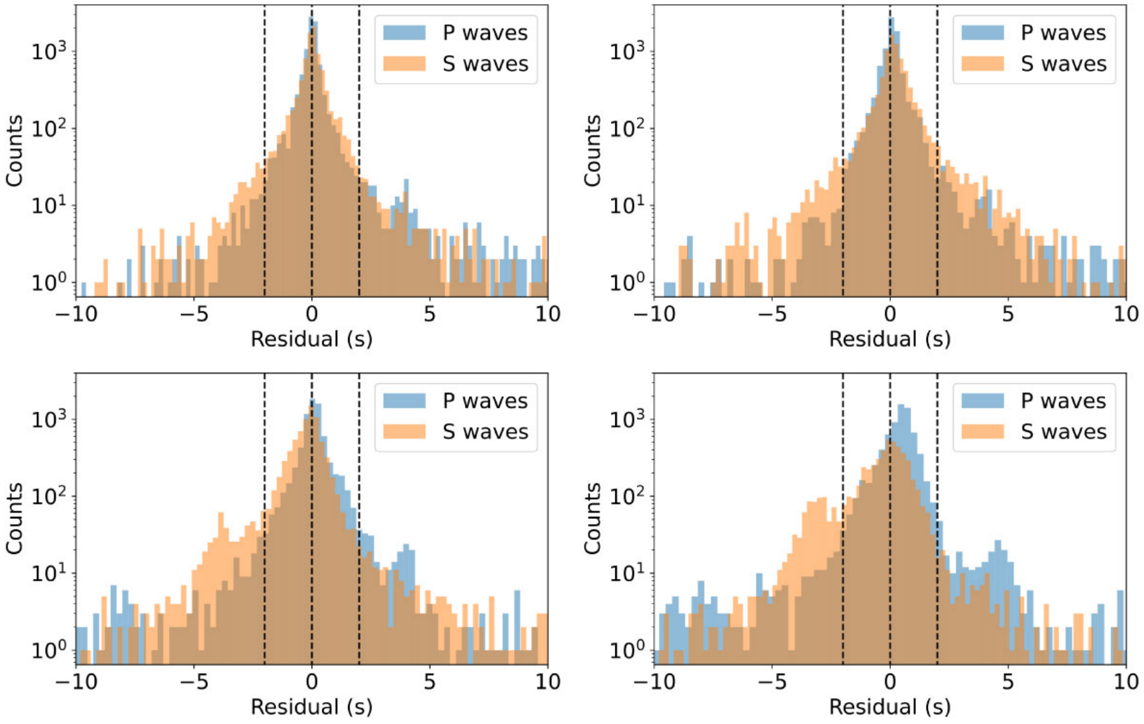

Figure A2. Pick time residuals with respect to picks of the analysts for test data set for all models.

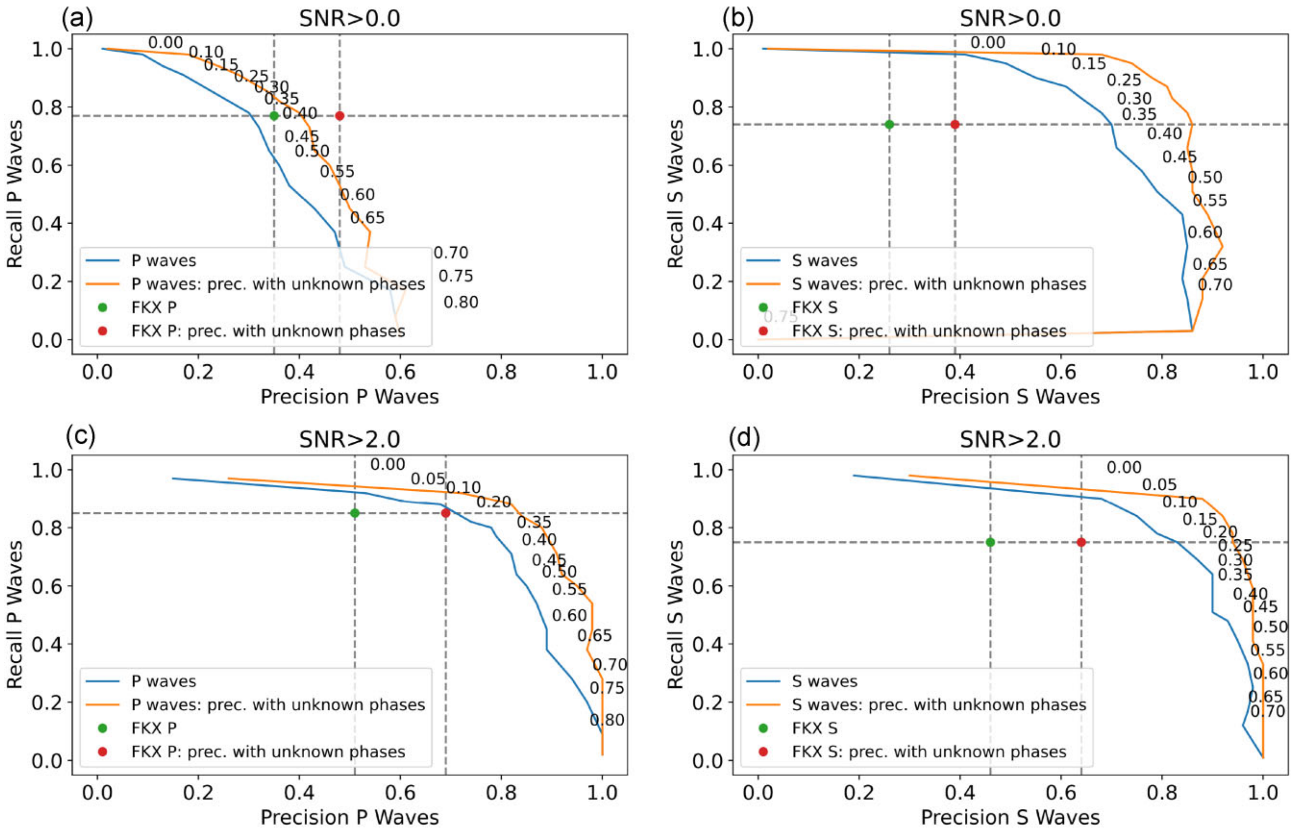

Figure A3. Performance metrics of TPhaseNet at station ARA1 for different decision thresholds, without (a and b) and with SNR threshold (c and d). The closer the displayed curves bend towards the upper-right corner, the better the performance. Recall and precision for FKX phase detection in operation at ARCES are added as symbols (no detection threshold data available). For each precision and recall pair a second one is added where arrivals labelled as 'unknown' are counted as either *P* or *S* waves for calculating precision.

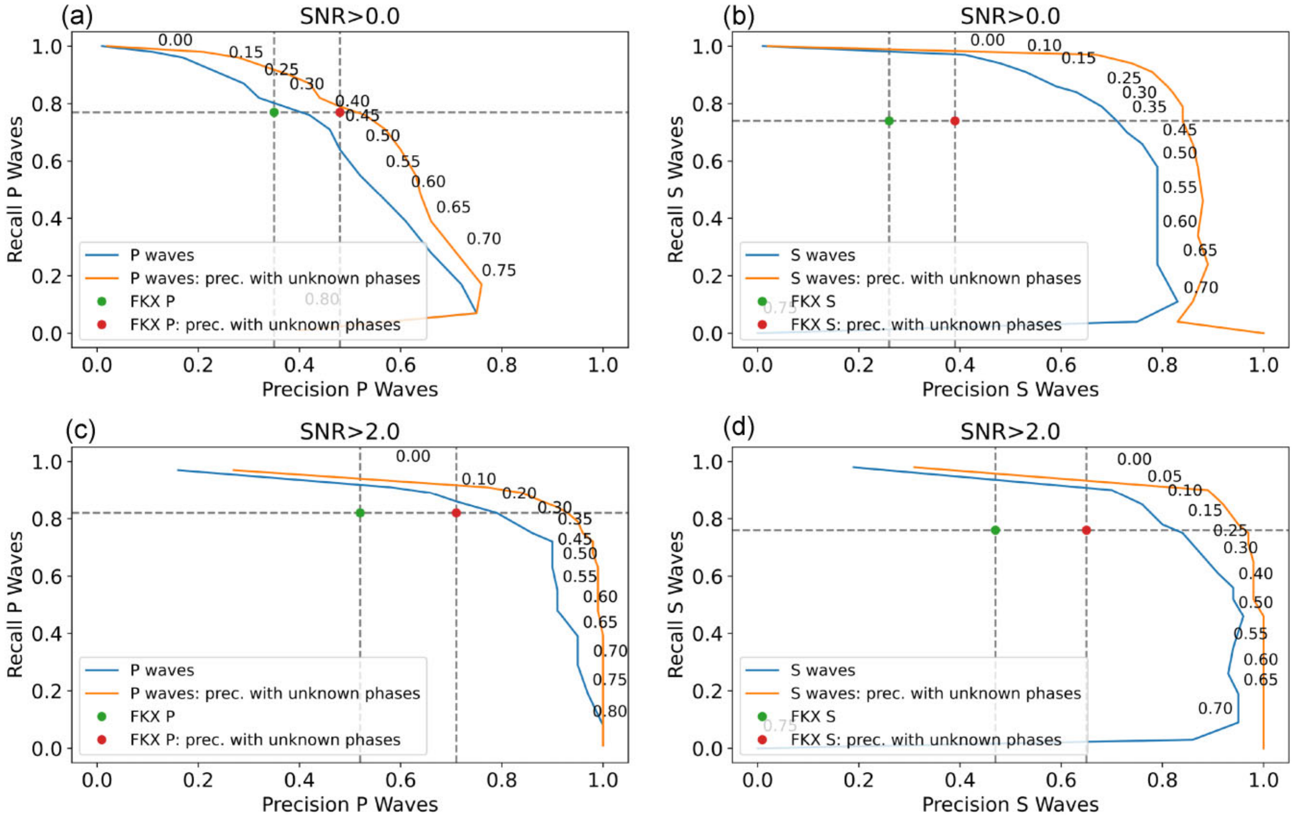

Figure A4. Same as Fig. A3, but performance metrics of TPhaseNet at station ARA2 for different decision thresholds.

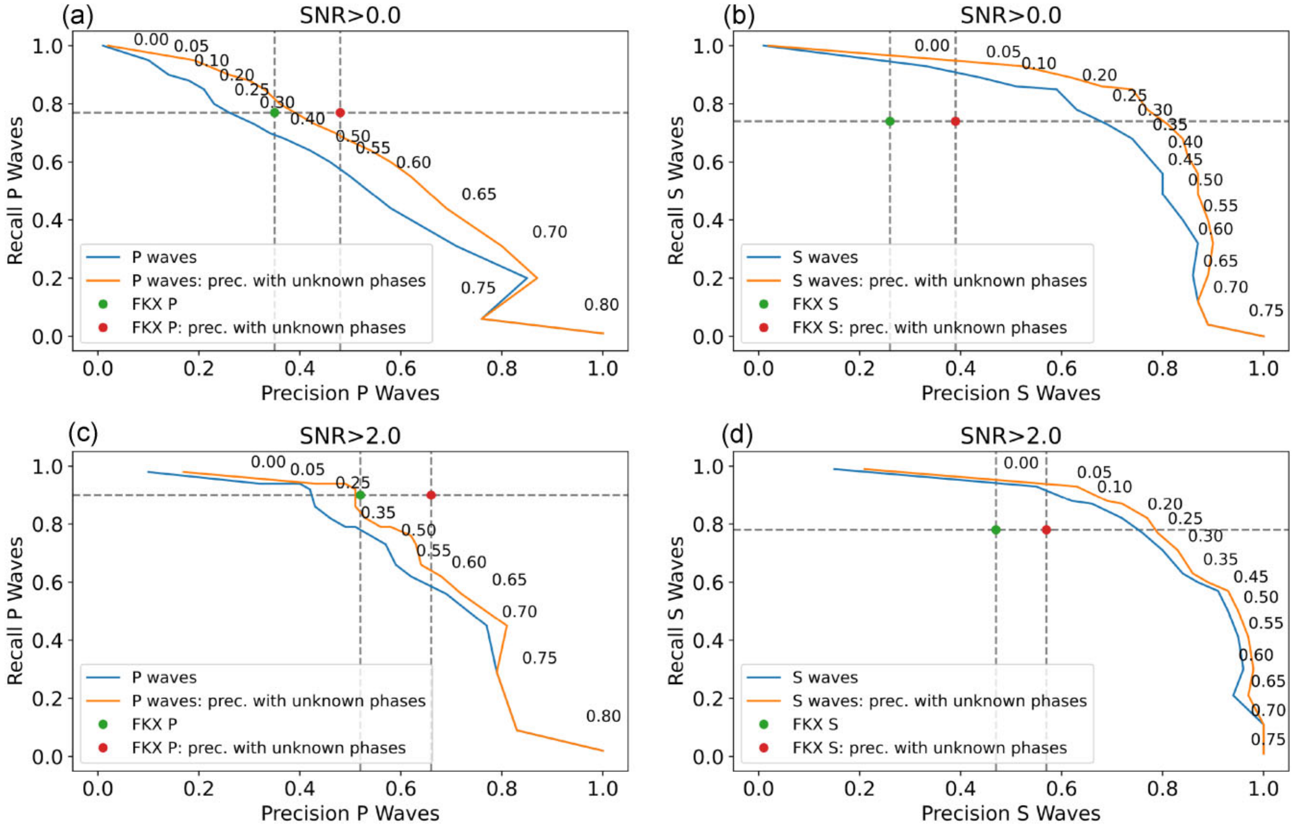

Figure A5. Same as Fig. A3, but performance metrics of TPhaseNet at station ARC3 for different decision thresholds.