SANDIA REPORT
SAND2022-12289
Printed September 13, 2022

Sandia National Laboratories

GraphAlign: Graph-Enabled Machine Learning for Seismic Event Filtering

Joshua Michalenko, Indu Manickam, Stephen Heck

Prepared by
Sandia National Laboratories
Albuquerque, New Mexico 87185
Livermore, California 94550

Issued by Sandia National Laboratories, operated for the United States Department of Energy by National Technology & Engineering Solutions of Sandia, LLC.

Available to DOE and DOE contractors from

U.S. Department of Energy
Office of Scientific and Technical Information
P.O. Box 62
Oak Ridge, TN 37831

Telephone: (865) 576-8401
Facsimile: (865) 576-5728
E-Mail: reports@osti.gov
Online ordering: http://www.osti.gov/scitech

Available to the public from

U.S. Department of Commerce
National Technical Information Service
5301 Shawnee Road
Alexandria, VA 22312

Telephone: (800) 553-6847
Facsimile: (703) 605-6900
E-Mail: orders@ntis.gov
Online order: https://classic.ntis.gov/help/order-methods

ABSTRACT

This report summarizes results from a 2 year effort to improve the current automated seismic event processing system by leveraging machine learning models that can operated over the inherent graph data structure of a seismic sensor network. Specifically, the GraphAlign project seeks to utilize prior information on which stations are more likely to detect signals originating from particular geographic regions to inform event filtering. To date, the GraphAlign team has developed a Graphical Neural Network (GNN) model to filter out false events generated by the Global Associator (GA) algorithm. The algorithm operates directly on waveform data that has been associated to an event by building a variable sized graph of station waveforms nodes with edge relations to an event location node. This builds off of previous work [3] where random forest models were used to do the same task using hand crafted features. The GNN model performance was analyzed using an 8 week IMS/IDC dataset, and it was demonstrated that the GNN outperforms the random forest baseline. We provide additional error analysis of which events the GNN model performs well and poorly against concluded by future directions for improvements.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

1. INTRODUCTION

This report details the project progress made on the Ground-based Nuclear Detonation Detection (GNDD) funded project GraphAlign. This report is organized as follows. Section 1 provides an introduction to the goals of the GraphAlign project. Section 2 provides background for the project's technical approach and introduces Graphical Neural Networks (GNN). Section 3 details the dataset, problem formulation, and provides some exploratory analysis of the International Monitoring System (IMS) data. Section 4 discusses improvements to the GNN model since the last progress report. Section 5 discusses test results between the baseline and GNN model and digs into event level error analysis. Lastly, section 6 provides conclusions and a discussion of future work.

Our work in this area is motivated by the following three characteristics of current automated seismic event building pipeline:

1. 42 % of events in the analyst reviewed bulletin are manually built by human analysts.
2. 38% of the events produced by the automated processing pipeline are rejected as false positives by human analysts.
3. It takes a human analyst 4 to 10 times longer to build an event missed by the automated processing pipeline than it does for the analyst to reject a false event produced by the automated pipeline.

The goal of the GraphAlign project is to improve the current automated seismic event system by utilizing machine learning models that can operate over the inherent graph data structure of a seismic sensor network. Specifically the GraphAlign project seeks to utilize prior information on which stations are more likely to detect signals originating from particular geographic regions to improve signal detection, and potentially signal association.

To date the GraphAlign team has successfully developed a GNN model for filtering out false events generated by the Global Associator (GA) algorithm. This builds off of previous work [3] where random forest models were used to do the same task. The GNN model performance was analyzed using an 8 week IMS/IDC dataset, and it was demonstrated that the GNN outperforms the random forest baseline.

2. BACKGROUND

2.1. Associator Systems

For the purposes of this project, ground-based nuclear explosion monitoring systems consist of two primary steps. In step 1, signal detection algorithms scan waveform data recorded by seismic sensor networks and produce signal detections consisting of time, azimuth, and slowness. In step 2, the event detection and association algorithm builds a set of sources/events consisting of latitude, longitude, depth, and time from the signal detections created in step 1.

Figure 2-1 describes the typical iterative processing pipeline of an associator system. Because operational systems monitor for seismic events in near real-time, *State Management* provides the necessary functionality for processing real-time data. The *Event Detection* component builds a set of event candidates that fit the input data (locations and phase arrivals). *Event Revision* resolves association conflicts, e.g. when two event candidates are associated to the same signal detection, and removes events that don't satisfy user-defined event quality thresholds. *Output Reconciliation* performs the processing required to 1) report new events that weren't detected previously, 2) make updates to previous reported events, and 3) remove previously reported events that are no longer viable. At the end of each data input iteration, events are written an automated bulletin (e.g. SEL3) that is reviewed by human analysts. Detailed information on this process can be found in Heck et al. [6].

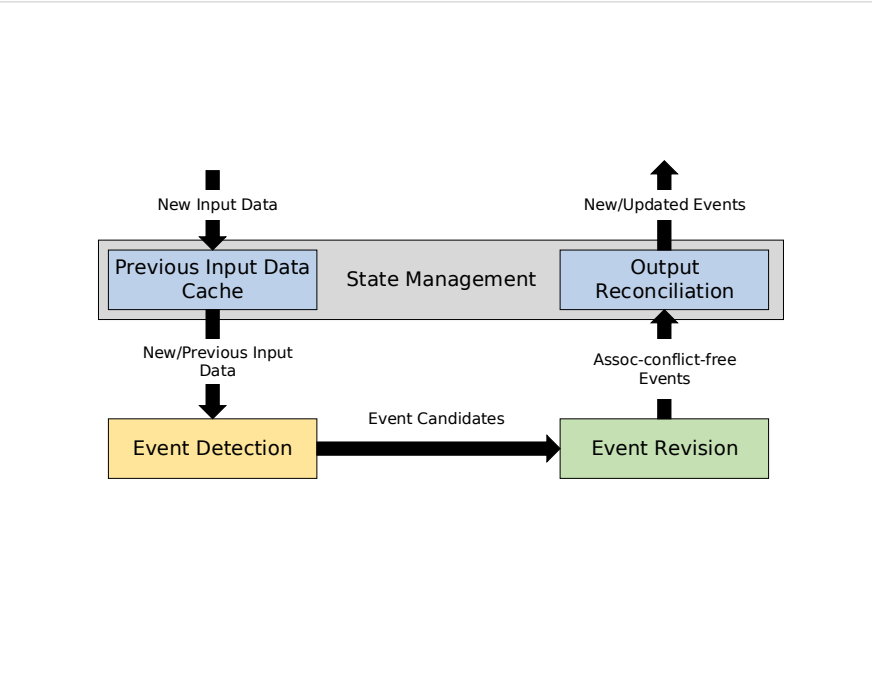


Figure 2-1 Typical building blocks and flow diagram of a seismic network associator algorithm. Recreated from [6]

2.2. Graphical Neural Networks

The GraphAlign algorithm is based on graphical neural networks which are a generalization of convolutional neural networks (CNNs) that can operate over graph data structures. At a high level, GNNs are differential message passing networks with nonlinearities. Similar to how a CNN uses convolution to aggregate information over the spatial context of an image, a GNN uses a neighborhood aggregation scheme to aggregate local information over an unordered graph structure.

The GNN includes a graph that is represented by a set of nodes and a set of edges between nodes. The edges may be weighted, with weight $e_{i,j}$ defined between a pair of nodes i and j. For each

layer k of the GNN, a node feature vector is defined over each node in the graph. Concretely, let $\mathbf{x}_i^{(k)}$ be the real valued node feature vector for the i^{th} node in the graph for the k^{th} network layer. To aggregate node features to pass on to the next layer of the GNN, we define a function $\phi^{(k+1)}$ between a pair of neighboring nodes i and j at layer k with node features $\mathbf{x}_i^{(k)}$ and $\mathbf{x}_j^{(k)}$ and edge weight $\mathbf{e}_{i,j}$. The array output of $\phi^{(k+1)}$ is also referred to as a node-to-node message. For all neighboring nodes of i, the set $\mathcal{N}(i)$, the node-to-node messages are aggregated using a differentiable, permutation invariant function ψ. The *update* for $\mathbf{x}_i^{(k)}$ is defined by equation 1 where $\gamma^{(k+1)}$ is a function that combines node features $\mathbf{x}_i^{(k)}$ with its aggregated neighborhood message to produce the $k+1^{th}$ layer's node feature embedding. [4]

$$\mathbf{x}_i^{(k+1)} = \gamma^{(k+1)}\left(\mathbf{x}_i^{(k)}, \psi_{j\in\mathcal{N}(i)}\,\phi^{(k+1)}\left(\mathbf{x}_i^{(k)}, \mathbf{x}_j^{(k)}, \mathbf{e}_{j,i}\right)\right) \tag{1}$$

A pictorial example of the node feature update described by equation 1 is shown in Figure 2-2 for a particular message passing operator ψ that equates to a simple multi-layer perception. After the update is applied, the structure of the graph remains intact, while the node features have been embedded in a lower dimension that can be visualized or used for prediction. There are dozens of message passing operators in the literature [10] specialized to the inherent graph structure.

Further details on how the seismic events is encoded as a graph and our spatio-temporal convolutional message passing operator (i.e. γ) are provided in Section 4.

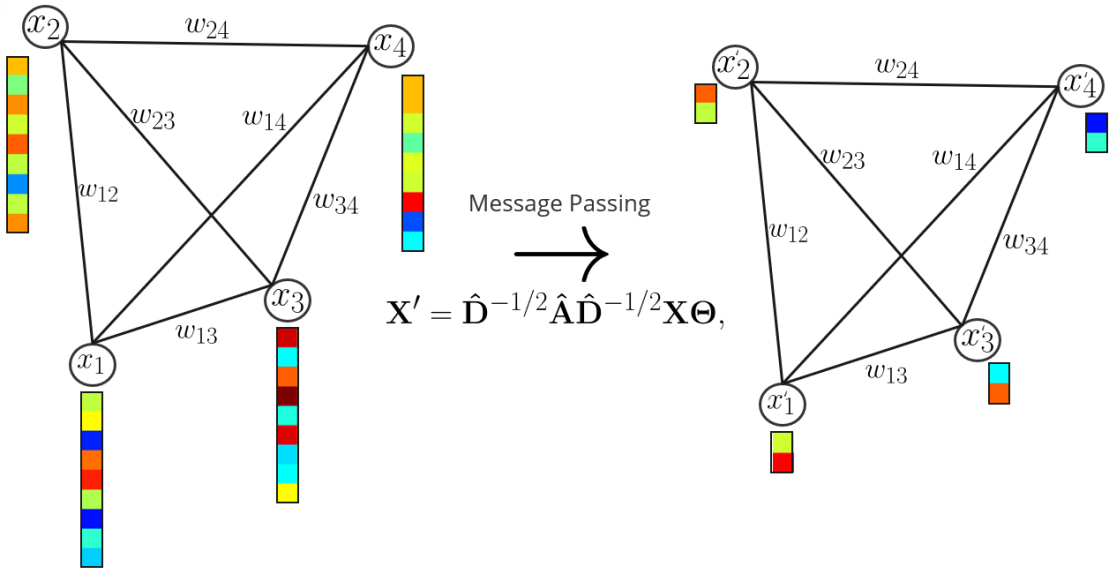


Figure 2-2 Example graph data representation before (*left*) and after message passing (*right*). The column of colored squares for each node represents the set of features. The message passing operation can reduce node feature dimension while training creates a node feature embedding space that linearly separates embedded nodes to particular classes.

3. DATA

3.1. IMS Data Set

The data used in the project is curated to evaluate the efficacy of data driven models that screen built events from the automated Global Associator system. The data is derived from the IDC's processing of a 8 week time interval of International Monitoring System (IMS) station data shown in Figure 3-1. From April 3, 2010 to May 29, 2010, approximately 7000 events were built by the automated Global Associator system. Roughly 3500, or half of the built events, survived analyst review (Figure 3-2). This time period was chosen because it overlaps with the Unconstrained Global Event Bulletin (UGEB)[7]. We treat the last week as our test set for evaluation.

3.2. Event Labeling

The Event Commonality Score (ECS) gives a [0,1] score of how well events in a bulletin (in our case SEL3) match those of a reference bulletin (LEB) [6, 2]. SEL3 Events with an ECS above 0.3 were labeled as well built events (class 1) while all others were labeled non-event (class 0).

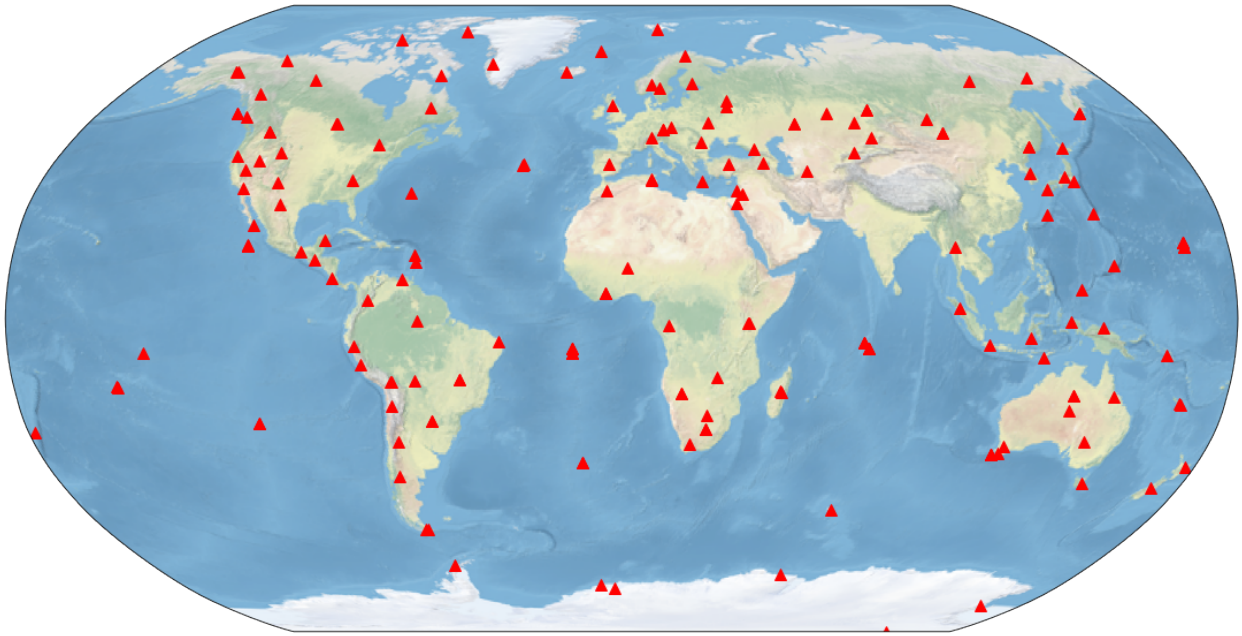


Figure 3-1 World map of primary and auxiliary seismic stations present within our IMS/IDC dataset.

3.3. Data Exploration

From previous work [3], the number of location-defining phases (*ndef*) was an important feature for predicting whether automatically built events were valid. Figure 3-3 shows the distribution of true and false events as a function of the number of defining phases. Almost all events that have

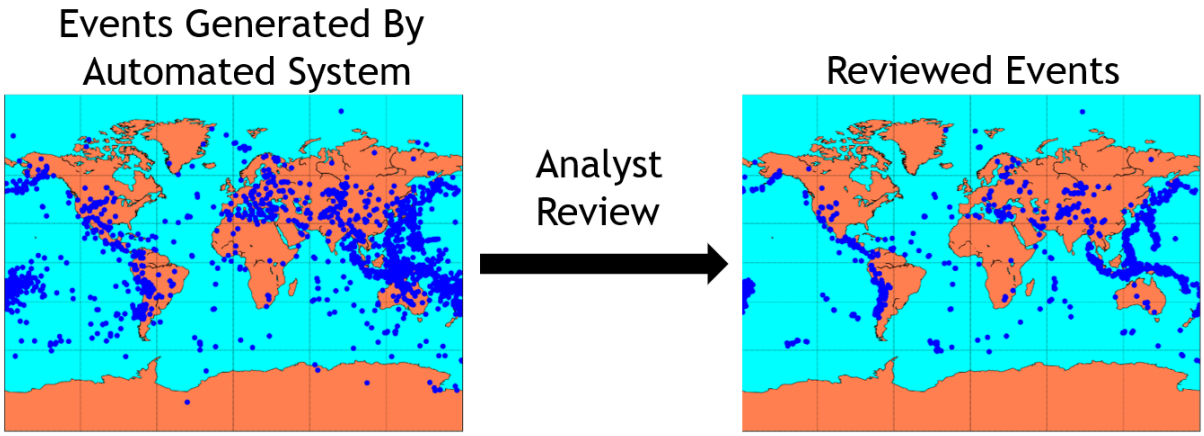


Figure 3-2 Geographical distribution of built events before *(left, SEL3)* and after *(right, LEB)* analyst review. Approximately half of the built events are discarded by analysts.

greater than 8 defining phases are labeled as true events. It is well known that the GA system performs remarkably well on events with ndef greater than 8. With this in mind, the greatest benefit from a filtering system would mostly likely be on smaller events with small ndef. From Figure 3-3 we can see there is a roughly equal split of valid and false events for ndef ≤ 8. A similar distribution of true and false events can be seen in Figure 3-4 which describes the events by the maximum signal to noise ratio (SNR) within the set of assocs for each event. As SNR increases, the likelihood of a true event is larger than if the SNR was small. This makes intuitive sense since one characteristic of a false event is that there are no associated signals with high SNR, whereas for a real event usually there is at least one high SNR associated signal.

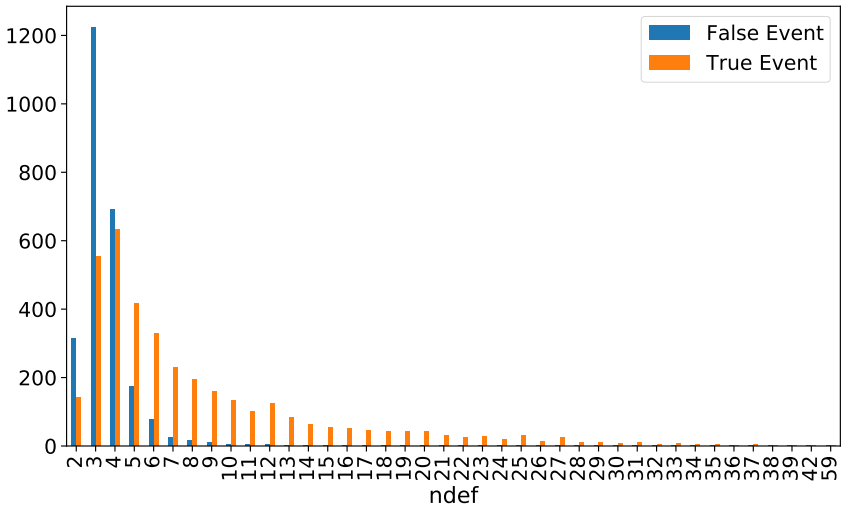


Figure 3-3 Distribution of false and true events based on number of associations (ndef) for the event. The LEB tends to assign events with larger number of associations (ndef ≥ 8) as true events. Designation as false and true is based on whether events have a SEL3-LEB ECS score greater than 0.3

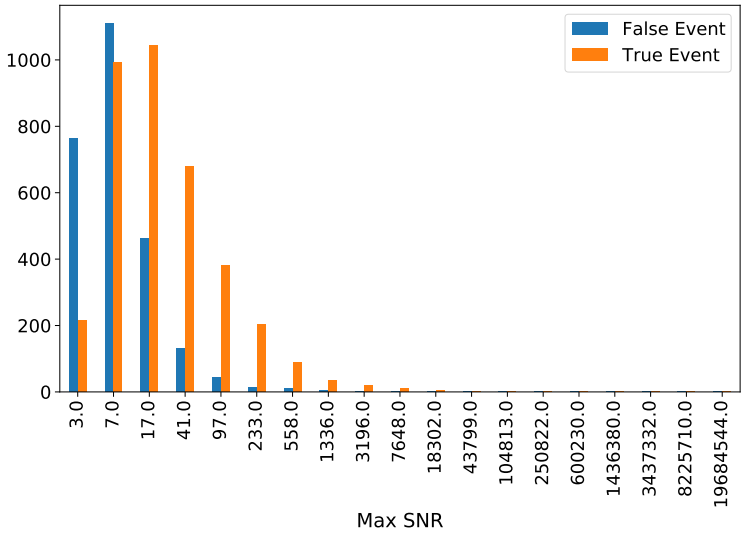


Figure 3-4 Distribution of false and true events based on max SNR over all associations for the event. Results indicate that the LEB tends to label events with higher SNR values as true events. Note that there are a small number of events with SNR greater than 5.0e4 that are not visible in the figure.

4. MODEL DEVELOPMENT

This section details the Year 2 GraphAlign GNN model. The input to the model is a graph representation of a single seismic event. Section 4.1 describes how these graphs were constructed, and compares the current graph structure to the Year 1 GraphAlign graph structure. The GNN model processes the graph inputs using multiple message-passing layers. Section 4.2 details the specific architecture used for the Year 2 GraphAlign GNN model. Section 4.3 provides an overview of the development process used to determine the final graph structure and model architecture including some of the trade-offs considered.

4.1. Representing Seismic Events as Graphs

4.1.1. Star Graphs

Similar to the Year 1 graph structure, stations are represented as nodes in the event graph. For array stations, the first arriving station was selected and represented as a single node. In addition to the station nodes, one additional node was created to represent the event origin. This additional node, which we refer to as the *event node*, serves as a parent node that aggregates waveform data across all stations. Each station node has a single bidirectional edge connected to the event node. This particular graph structure is referred to in literature as a *star graph*, and could also be considered a tree structure with a single parent node. An example star graph is shown in Figure 4-1.

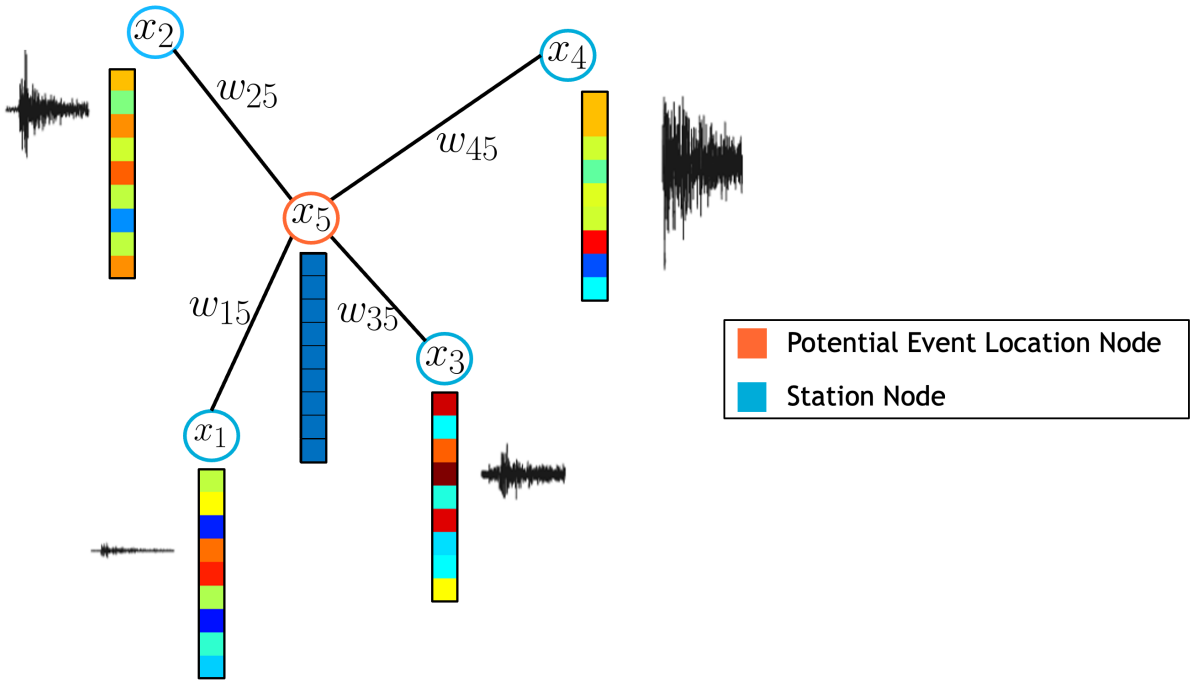


Figure 4-1 Example star graph structure. Nodes x_1 to x_5 represent single stations. The event node in the center of the star graph (x_5) represents the event origin. Each station has a bidirectional edge to the event node with edge weight w.

The underlying assumption that determined this particular graph structure is as follows. When analysts consider global seismic events, stations that are geographically proximate do not necessarily have strong correlation in terms of the event origins that they tend to detect. Because of the geophysical path effects, the set of stations that detect a particular event may be scattered across the globe. It therefore does not make sense to draw edge relations between station nodes in the graph, since station pairs may strongly correlate in picking up events from one geographic location and may exhibit weak correlation in detecting events from another region.

We instead consider that for a given event origin, each individual station has a probability of detecting seismic activity, which can be estimated based on historical seismic events. We therefore chose the star graph structure, where node pairs are between each graph and the event origin and the edge weights, ranging from 0 to 1, are the historical probability of that station detecting activity from the event origin. Further details on the probability of station detection model used to determine edge weights are included in Section 4.1.3.

The star graph has been been used previously in GNN models. A common approach in GNN architectures is to perform message passing on a star subgraph, by extracting a single node and all of its one-hop neighbors (e.g. [5]), or by developing a star graph representation of a larger network [8]. By directly using a star graph structure, we can utilize message passing algorithms designed for this type of graph structure.

The data contained at each station node is a spectrogram representing the station waveform data over a fixed time window. The spectrogram construction process is described in 4.1.2. In addition

to the spectragram, the station node also includes two event scalar values, the ndef and 3rd largest SNR value. These two scalar values were included in the model as they were the two most important features used in the random forest model described in [3], and to ensure that high ndef events are less likely to be rejected as false negatives. The event node features are also initialized as the average of all connected station spectrograms. Our experiments indicated that models where the event node was initialized as an averaged spectrogram outperformed models with the event nodes initialized with random values or to a constant.

For a given event, the stations represented as nodes in the graph include all stations that were associated with the event, as well as stations that meet a minimum probability of detection (pdet) threshold. The edge weight for associated stations are automatically set to 1, the maximum edge weight. For all other stations the edge weights are set to the sation pdet.

4.1.2. Station Spectrogram Construction

A large contributing factor in model performance is the quality of preprocessing performed on the data. In order to produce spectrograms that adequately capture waveform variability in both time and frequency, we use the following procedure.

All waveforms are initially downsampled to a common sampling rate (20hz) since teleseismic waveforms typically have very low SNR in frequencies above 6hz. The waveforms are then bandpass filtered with a lower/higher cutoff frequency of 0.5/6hz.

As mentioned in section 4.1, the homogeneous GNN models we've developed under the project require a fixed length feature vector at each node. One might expect we utilize an absolute window of time that would capture the complete record section of data the network sees. This would require a large amount of data be included in the data structure with little added inference value. Instead, we focus on a characteristic that an analyst would typical evaluate: how well aligned the observed signals are with predictions from a particular travel-time model. For a potential event location we utilize an AK-135 travel time model to create a fixed window around the first arriving seismic P phase to each station. An example set of waveforms we include for a potential event can be see in Figure 5-3. The window of time at each station is +/- 100 seconds around the first arriving phase. A well built event would likely have each of the arriving phases in the center of these windows, which is the case for this event.

Once we have a fixed length waveform at each station, a spectrogram is computed with a fixed set of parameters. We utilize a sliding window of 20 samples with an overlap of 80% between each window to compute FFTs over. The signatures are then put on a normalized dB scale to accommodate the large range of SNRs at teleseismic distances. An example of a seismic waveform and its computed spectrogram can be see in Figure 4-2. In both representations, the x-axis represents time while the y-axis represents amplitude and frequency of the time series and spectrogram, respectively. In the spectrogram, the onset of energy (shown at $x = 16$) manifests across nearly the entire spectral region and the coda can be seen in the higher frequencies. Our machine learning model should be engineered to learn to recognize these features to determine whether seismic activity has occurred.

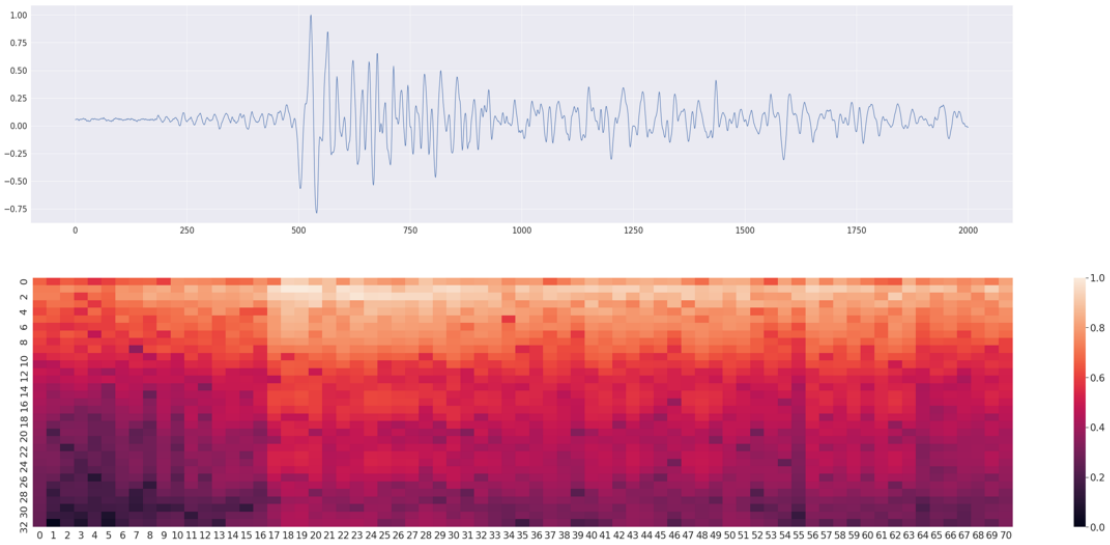


Figure 4-2 Example 100 second window of a seismic event waveform with an initial phase onset at $x = 500$ in the time series and $x = 16$ within the spectrogram

4.1.3. Station Detection Probability Model

The GraphAlign project utilizes station Probability of Detection (pdet) models that were were constructed under a previous SNL based project that utilized the PEDAL association algorithm [2]. The pdet models are constructed for each network station based on historical events in the IDC LEB catalog.

As an example, the pdet model for the ASAR station located in Australia is shown in Figure 4-3. Each grid point on the map represents a possible event origin. The coloring of the marker indicates the proportion of past events that have originated from this region and have an associated phase with the ASAR station. It is apparent from the figure that the event origin regions likely to be detected by ASAR include but are not limited to regions in physical proximity to the station.

Note that for some geographical regions seismic activity is highly sparse, in these regions the pdet is instead approximated with the average of high certainty pdet grid point estimates that are from a seismically active area that is from a different azimuth but a similar distance away from the station. This method provides one process to approximate station pdet and we acknowledge that other methods do exist.

In the GraphAlign GNN model, for a single event, we determine the pdet model geographical grid point that is closest to the event origin. The pdet value for each station at that grid point is passed to the GraphAlign model to construct the event graph.

4.1.4. Year 1 Graph Structure

We describe the Year 1 GraphAlign graph model to provide context on what changes we implemented this year. In contrast with the current model, the Year 1 graph structure included

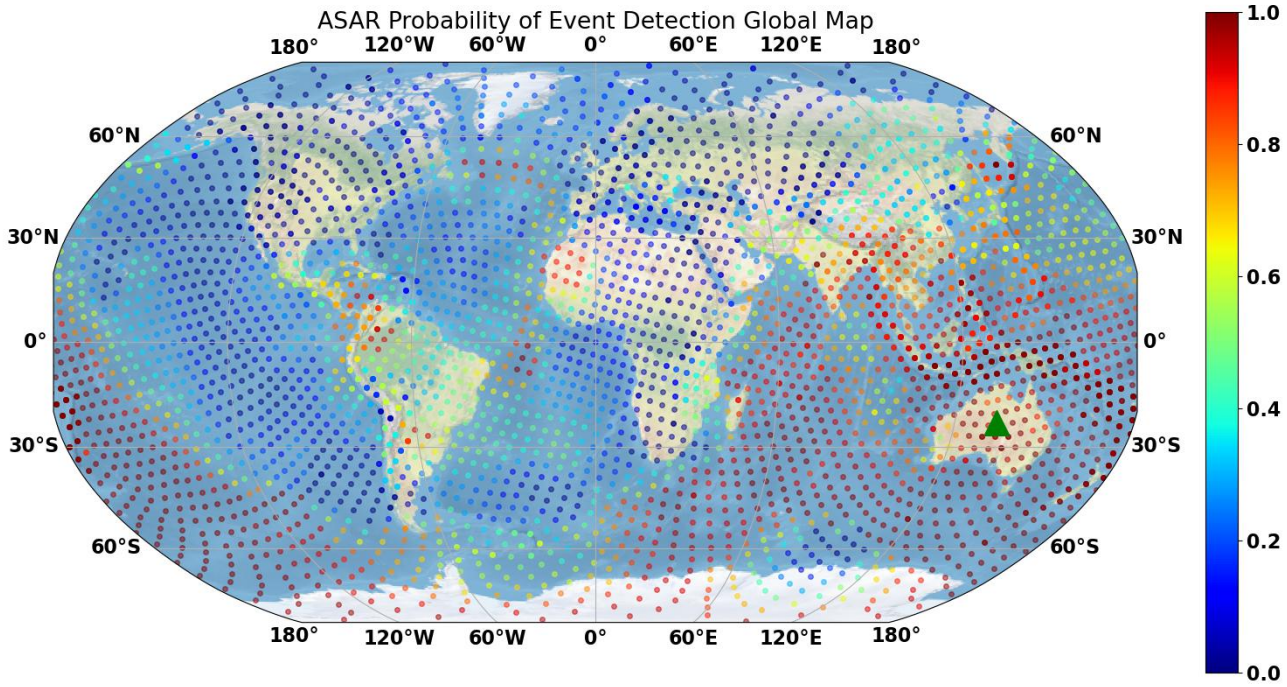


Figure 4-3 Example probability of detection map for the ASAR seismic array (green triangle) based in Australia.

only associated stations as nodes in the graph. The graph did not include an event node, and the graph was fully connected with edge weights between station pairs proportional to the distance between the stations.

Each node had a feature vector representing a spectrogram of a 100 second time series. Unlike the current model, which centers waveforms using the travel time model, the Year 1 model used waveforms centered around the arrival, with the window set to 10 seconds before the arrival and 90 seconds following.

4.2. GNN Model

The inputs to the GNN model include: (1) the stations and edge weights from the graph structure described in Section 4.1, (2) the scalar values for the ndef and 3rd highest SNR for each station node, (3) the spectrograms generated for each station node, and (4) the averaged spectrogram generated for the event node. The GNN architecture processes these inputs using the layers described in Section 4.2.2 and outputs a scalar model prediction value for the event. The key to the GNN architecture is the use of spatio-temporal layers, described in Section 4.2.1 which employ a message passing algorithm over the graph structure.

4.2.1. Spatio-temporal Convolution Message Passing

The Spatio-temporal Convolution (STC) message passing message structure is based on Covert et. al, [1], which in turn is based on the GraphSage algorithm [5]. In typical graph convolution, the features for a single node at the next layer are an aggregation of the node's current layer features and the current layer features of neighboring nodes. In this setup, the node's current layer features gets equal weight to each neighbor's current layer features (ignoring any weighting parameters). In the GraphSage model, the neighboring nodes are first aggregated, and then the aggregated neighbor features are combined with the node features in a second step. This is done to place higher relative importance on a node's own features.

The algorithm can be summarized as follows. At layer $k+1$, each station node (as well as the event node) are passed through a 1D convolution layer with kernel size 3 and stride 1, and a ReLU activation layer (Algorithm 1 State 1). For each node, the neighboring nodes are then aggregated. For ψ, the mean aggregation function was used (Algorithm 1 State 2). For station nodes, the only neighbor is the event node, which is multiplied by the edge weight (station-event pdet) . For the center node, a weighted average of all of the station nodes is determined, which weights each station by the probability of detect. Lastly, the node and aggregated neighboring node features are concatenated and passed through a final convolution plus ReLU activation layer with kernel size 1 and stride 1 (Algorithm 1 State 3).

Algorithm 1 STC Message Passing at layer $k+1$

Input:

- Graph $\mathcal{G}(V,E)$
- Node representations $x_i^k \ \forall i \in V$ from layer k
- Edge weights $e_{ij} \ \forall (i,j) \in E$
- Nonlinearity functions g_1 and g_2
- Neighborhood aggregation function ψ
- First convolution kernel and bias for layer $k+1$, $W_1^{k+1} * x_i^k + b_1^{k+1}$
- Second convolution kernel and bias for layer $k+1$, $W_2^{k+1} * x_i^k + b_2^{k+1}$

Output: Node representations $x_i^{k+1} \forall i \in V$

1: $a_i^{l+1} = g_1\left(W_1^{k+1} * x_i^k + b_1^{k+1}\right)$

2: $z_i^{l+1} = \psi_{j \in \mathcal{N}(i)}\left(e_{ij} \cdot a_i^{l+1}\right)$

3: $x_i^{k+1} = g_2\left(W_2^{k+1} * [z_i^{l+1} a_i^{l+1}] + b_2^{k+1}\right)$

4.2.2. GNN Architecture

The layers for the GNN model are described in Table 4-1 as well as the shape of each output layer for the example case of a seismic event with 15 nodes including the event node. The model is based off of the Temporal Graph Convolution Network (TGCN) developed by Covert et. al [1] to detect seizures using EEG waveform data captured by a network of lead sensors placed on patients' scalps.

The GNN architecture is similar in structure to a CNN. In a CNN, the first set of layers alternate between convolution layers and max pooling, followed by multiple linear layers, and finally a softmax layer is used to produce the scalar output. For the GNN, the convolution is replaced by a STC layer (see §4.2.1 for details), which includes message passing between each station node and the event node.

The input to the GNN are spectrograms of each station node. At each spatio-temporal layer, the spectrograms are first individually passed through a convolution layer, and then an aggregation (message passing) step takes place where information is shared between nodes. The spatio-temporal layer is used to extract features across the frequency channels of the spectrogram and aggregate information across the station nodes. The spatio-temporal layer is followed by a max pooling layer which uses a kernel of size 2 to aggregate the data over the temporal dimension.

After the final layer of max pooling, the model selects and flattens the output at the event node to propagate into subsequent linear layers. This is in contrast to Covert et al., which averages the node output over all nodes as the last step before the linear layers. GraphAlign takes this

approach because the event node has aggregated data across all of the station nodes through the message passing process.

Each linear layer is followed by an activation function. The first linear layer includes a dropout layer as a regularization step. After the second linear layer, which has output size of 16, the output is passed through a batch normalization layer, and the two normalized scalar values, ndef and the third SNR, were appended to data before it is passed through the final linear layer.

Table 4-1 Layers of the GNN model. An example of the output shape of each layer is provided for an event with 15 stations in the event graph.

Layer Name	Layer Output [Stations, Frequency Channels, Timesteps]
Time Series Input	[15, -, 269]
Spectrogram	[15, 128, 269]
STC	[15, 128, 269]
Max Pool 1D	[15, 128, 134]
STC	[15, 128, 134]
Max Pool 1D	[15, 128, 67]
STC	[15, 16, 67]
Max Pool 1D	[15, 16, 33]
Select Event Node + Flatten	528
Linear -> Dropout -> ReLU	64
Linear -> ReLU -> BatchNorm	16
Add Scalars	18
Softmax	2

4.3. Model Selection

To train the model we used an ADAM optimizer and cross binary entropy loss function. Model weights were determined by evaluating the model at each epoch on the validation dataset and selecting the weights which produced the minimum validation loss. This process was done to prevent overfitting to the training set. An example of the training and validation loss over training is shown in Figure 4-4

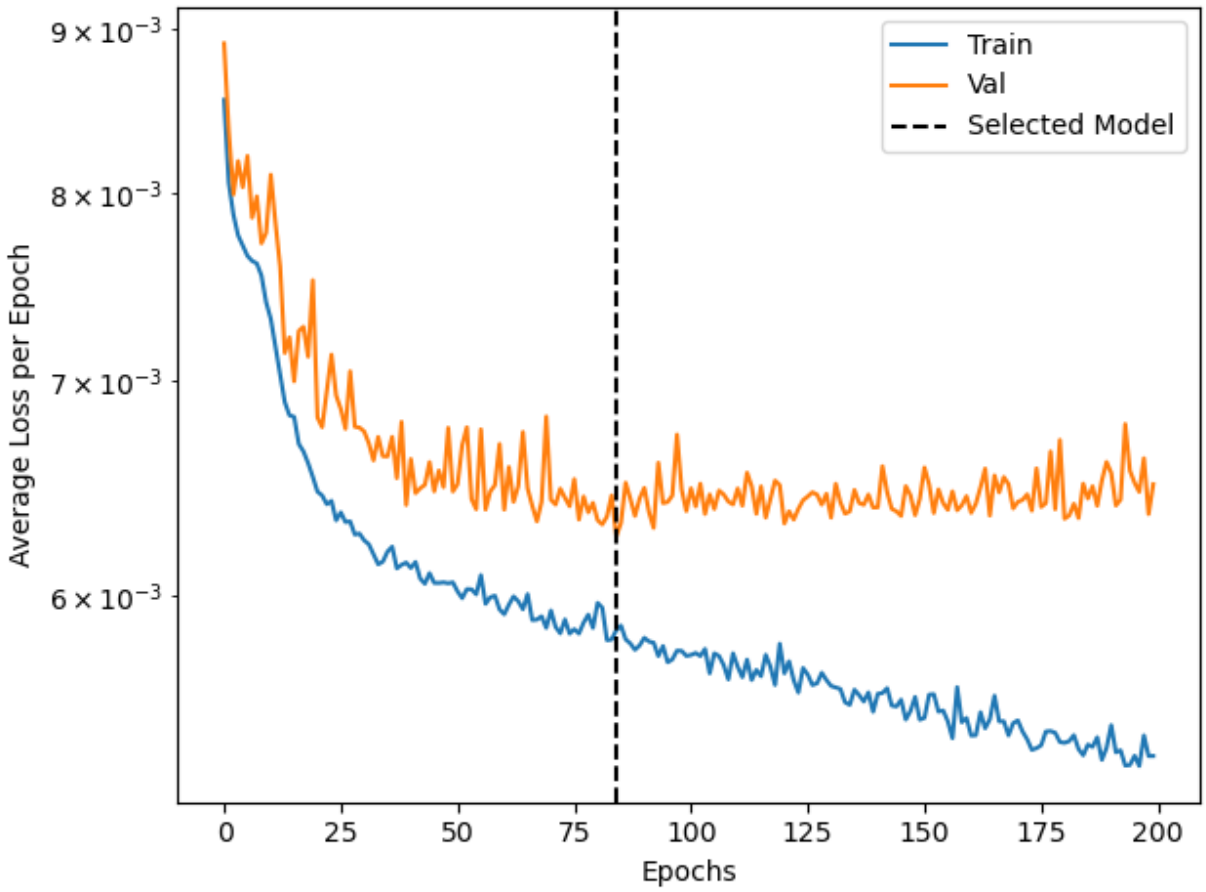


Figure 4-4 Training and validation loss per epoch for the GNN model. The epoch which produced the minimum validation score is indicated with a dashed line.

For model selection, 7-fold cross validation was used to compare model performance with different model configurations as shown in Table 4-2. Each fold represented a 1 week period over a total of 7 weeks, so that 1 week was used for the validation set and the remaining 6 were used for training.

We compared GNN models with different minimum pdet thresholds to determine which stations should be included in the graph. We also compared including stations with associations in the event versus only considering stations with sufficiently high pdet. These configurations are represented with asterisks in the table. The results indicated that we achieve the best performance with respect to the F1 score [9] when both including associated stations in the event graph and only including additional stations with pdet greater than 0.4.

	*pdet	*assocs	F1	TNR	Precision	Recall
Random Forest	–	–	0.803 ± 0.006	0.825 ± 0.017	0.830 ± 0.022	0.779 ± 0.018
GNN_1	0.25	–	0.814 ± 0.017	0.866 ± 0.018	0.864 ± 0.020	0.770 ± 0.021
GNN_2	0.25	Yes	0.822 ± 0.013	0.857 ± 0.045	0.862 ± 0.031	0.788 ± 0.034
GNN_3	0.4	Yes	0.822 ± 0.017	0.839 ± 0.042	0.849 ± 0.019	0.798 ± 0.037

Table 4-2 Comparison of GNN models using k-fold cross validation. * indicates model features. **pdet* represents the minimum threshold probability detect used to include stations. **assocs* indicates whether assoc stations (GA products) were included regardless of station pdet value. Mean metric values ± standard deviation on the validation set are reported. TNR represents the true negative rate.

5. EXPERIMENTS

5.1. Model Performance Evaluation against Baseline

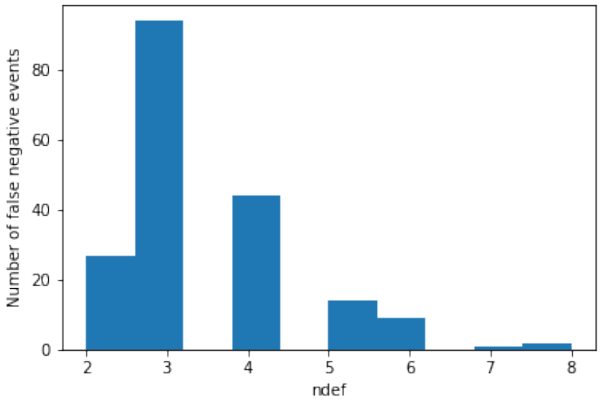
After performing model selection as discussed in Section 4.3, the final GNN model was evaluated against the test set. A random forest baseline model from [3], which was tuned using the model selection process, was also evaluated against the test set. The results are shown in Table 5-1.

While the performance difference between the random forest and GNN is not large, the results indicate that on both the validation set and the test set, the GNN model outperforms random forest on the precision and recall scores, as well as the true negative rate.

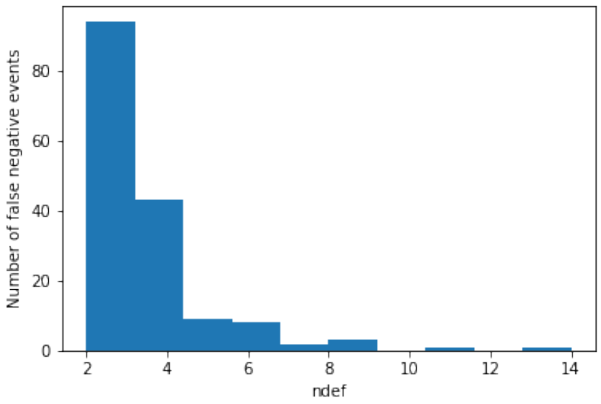
In addition, in Figure 5-1, we examine the distribution of ndef values for events that were marked as false negatives for the random forest, and separately for events marked as false negatives by the GNN. Given the considerable effort on the part of analysts to make ndef as large as possible for each event and the expectation that high ndef events are true seismic events, we expect that models should be biased towards correctly classifying high ndef events. As indicated in the figure, the GNN has a higher max false negative ndef of 14 compared to 8 for the random forest. However, there are only a small number of misclassified events for the GNN model that have ndef greater than 8. In addition, the GNN outperforms random forest on events with ndef > 2. This may be because the GNN is aided by the addition of the high pdet stations that were not associated with the event.

	F1	TNR	Precision	Recall
Random Forest - Draelos et al.	0.797	0.824	0.863	0.740
GNN	0.825	0.830	0.874	0.781

Table 5-1 Comparison of GNN and Random Forest performance on a test set.



(a) Random Forest



(b) GNN

Figure 5-1 Distribution of events by ndef for events labeled as false negatives in the random forest model (*top*) and the GNN model *bottom*)

5.1.1. GNN Inferance on Particular Events

One attractive aspect of the GNN model is that no feature engineering is required in order to generate model predictions. Although we include some GA products in the various models of section 5.1, we can instead train a model similar to GNN_1 with no included GA associated stations or scalar features. This results in a model with similar performance but since the input data structure purely depends on a given location and travel time model, we can artificially create new potential events and model predictions for any point in geographical space-time.

Our team trained such a model and selected 4 unique events to gain intuition into which features of the graph data structure the model is picking up on. These 4 events with their associated ORID, ndef, ECS Score, and GNN prediction can be found in Table 5-2. The events represent a true positive, true negative, false positive and false negative prediction from the GNN model. Per our labeling scheme described in section 3, true events are described by ECS scores > 0.3. For each event, we show two figures to represent the event and one to demonstrate GNN predictions.

orid	Location	ndef	ECS Score	GNN $p(y=1\|x)$	Why Unique?
6310974	Indonesia	35	.84	.730	Large TP event
6362728	Alaska	7	.728	.364	Large ndef FN event
6349783	South America	9	.231	.585	Large ndef FP event
6351738	East Russia	8	.02	.366	Large ndef TN event

Table 5-2 Four events within the SEL3 bulletin we highlight in our error analysis. Ndef is the number of defining phases of each event. ECS Score is a [0,1] score of how this event relates to the LEB reference bulletin. GNN $p(y=1|x)$ is the GNN predicted probability.

Origin ID: 6310974 True Positive Indonesia Event The first example is a large true positive event with the ORID 6310974 per the SEL3 bulletin where GA built an event near Indonesia. Figure 5-2 shows the event with a red star and all potential seismic stations in our dataset with triangles. Triangles are color coded by the associated probability of detection of the station in relation to the Indonesia event, with red having a higher probability than blue. For this particular model, a pdet threshold of 0.25 was used but all stations that meet a pdet > 0 threshold have bold text above them. Lastly, stations that are associated with the GA built event have a green hue around the triangle. Figure 5-3 shows an accompanying seismic record section for the event. All of the waveforms in Figure 5-3 meet the 0.25 pdet threshold and are used in the GNN data structure. The waveforms are initially color coded with the associated teal/magenta coloring scheme to denote the pdet of each station but GA associated stations are masked in orange.

On initial inspection, the arriving phase in a majority of the waveforms can easily be seen in the center of the travel time delayed window. Most analysts would deduce this is a well built event and our model tends to agree with a prediction of 0.73. In an effort to further understand which features the GNN is leveraging for prediction, and to evaluate model spatio-temporal sensitivity, we decided to leverage the key data-driven attribute of the GNN and produce a grid of potential event locations for the same time instance of the Indonesia Event. The result can be seen in the

geographically zoomed in region of Figure 5-4 with the event location (red star) in the center of the grid. Each point represents a new potential event location, with a new data structure built using the travel-time model that is fed to the trained GNN model. Event locations are colored by the GNN's prediction of whether the event is well built or not. An obvious artifact of the prediction landscape is a concentric high-probability circle around the ASAR and WRA stations, which lines up well with the Indonesia event, but obviously has cause for future investigation. One of our strongest hypothesis for what causes these station centered arcs/rings has to do with the graph edge weights. WRA and ASAR are primary IMS stations with a high probability of detection for this region based on the methodology from section 4.1.3. Since pdet is used as both a selection criteria but also to weight graph edges, the WRA and ASAR station nodes will have much higher weighting and aggregation power in the STC message passing algorithm. The result is that stations with high pdet and well aligned phases can heavily bias the model which allow high pdet stations to dominate the prediction landscape. To remedy this, the team would like to make the graph edge weighting a higher priority in the rest of FY22.

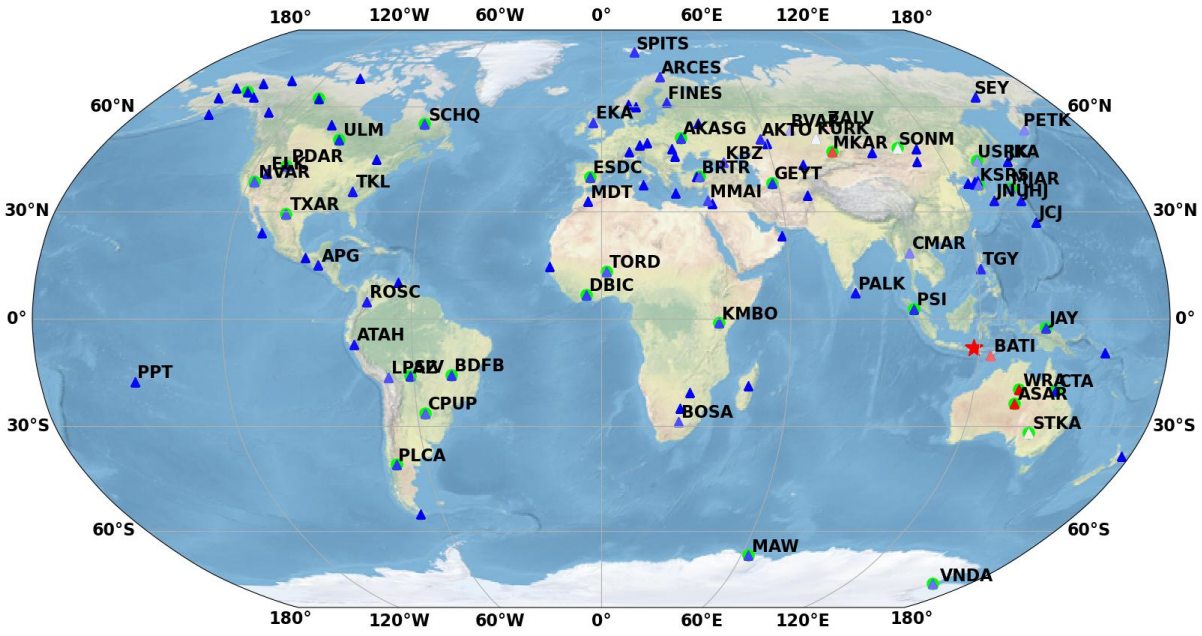


Figure 5-2 World map of the IMS seismic network with respect to the 6310974 orid event based around Indonesia (red star). Triangles represent potential stations included in our data structure, which are colored by the probability in our pdet model. Blue represents a station would typically have a low probability of detecting an event in the potential event location and red would be a high probability. Stations with a green circle represent stations the GA system associated with the event, in this case the number of defining phases is 35. Lastly, stations with pdet>0 have text labels.

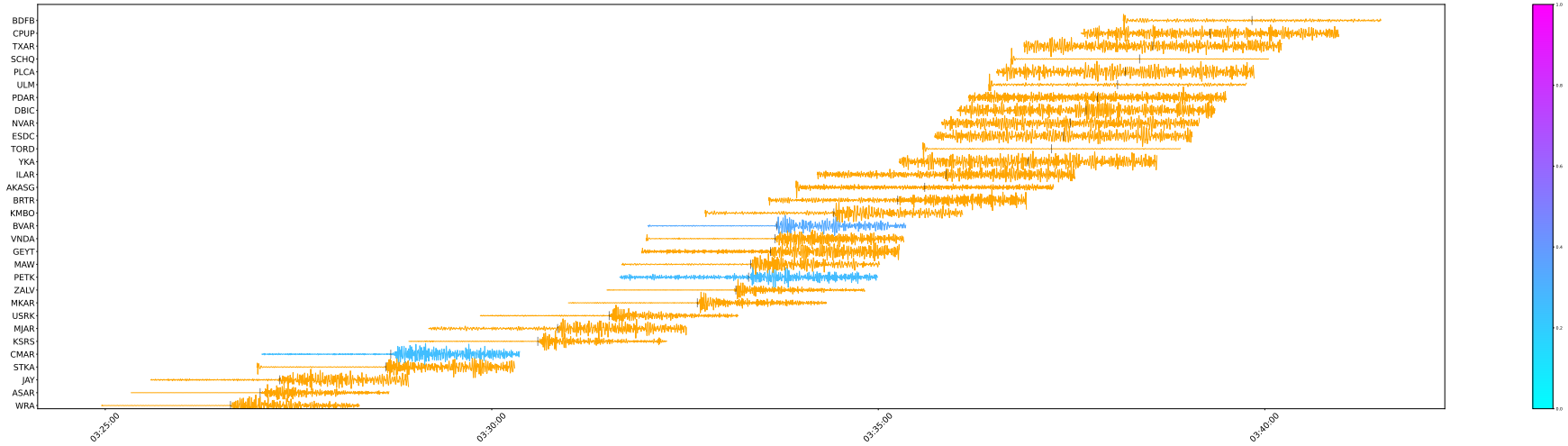


Figure 5-3 Seismic run out plot for the large Indonesia event. Station waveforms with pdet>0.25 are shown and color coded by their respective pdet value to the GA event location. Waveforms colored in orange represent GA associated stations.

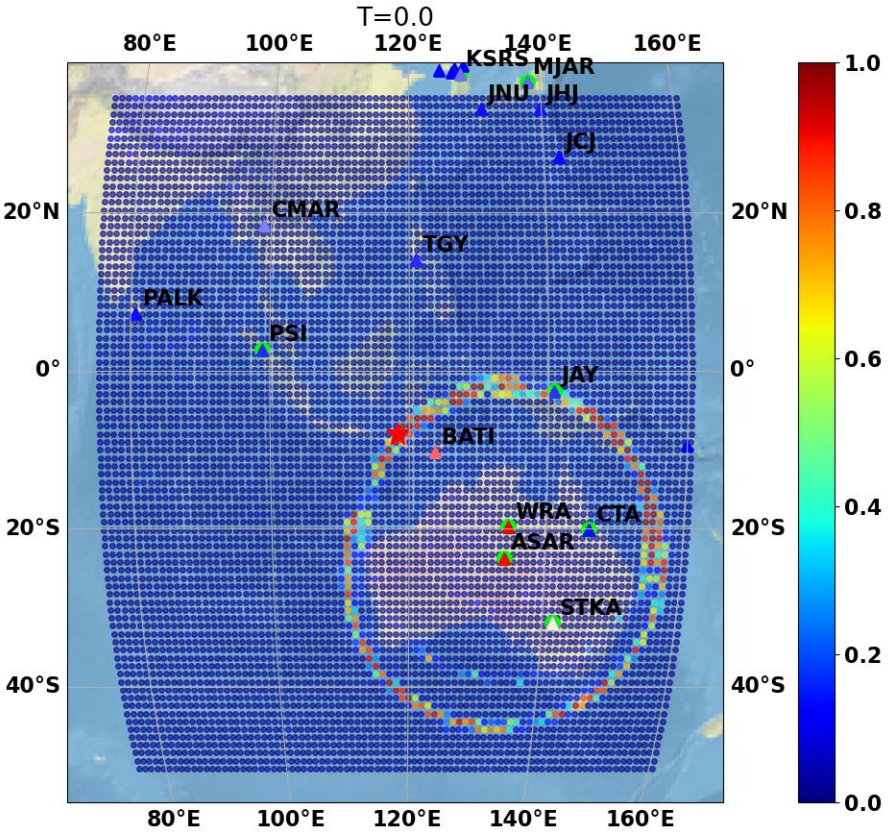


Figure 5-4 GNN probability of detect map for the 6310974 orid event located near Indonesia (red star).

Origin ID: 6349783 False Positive South America Event We can see a similar example for the the false positive event based in South America in Figures 5-5, 5-6, and 5-7. A high probability ring is centered around the LPAZ station, which most likely occurs because of the high pdet associated with LPAZ in this region and a well aligned phase along the path. This leads us to hypothesize that the model is picking up on the seismic energy in this region and there is plausibly a real event in the LEB bulletin during this spatio-temporal window. We also note that the timing of this ring does not line up well with the potential event location. Yet, there is a very small cluster of GNN high probability for this region that the model prediction is ultimately inferred from. One potential new direction the team would like to investigate is how the size and shape of these probability masses across space-time can aid better filtering predictions. E.g. There may be simple heuristics to differentiate between a sporadic noise-like prediction and an actual signal.

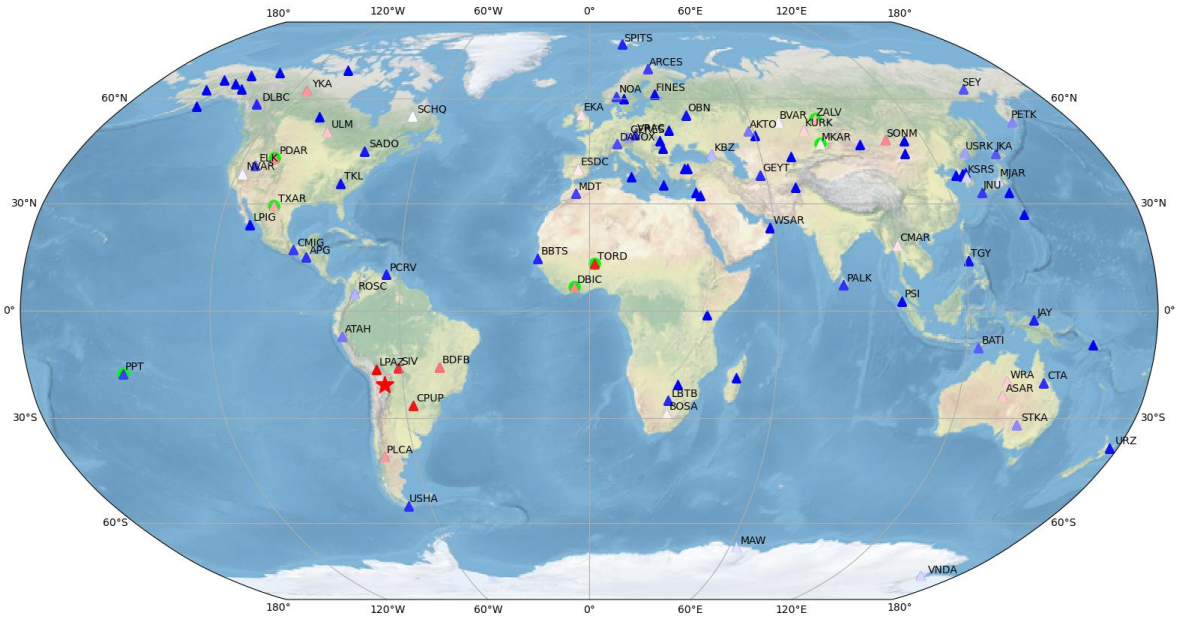


Figure 5-5 World map of the IMS seismic network stations with respect to the 6349783 orid event based in South America (red star). Triangles represent potential stations included in our data structure, which are colored by the probability in our pdet model. Blue represents a station would typically have a low probability of detecting an event in the potential event location and red would be a high probability. Stations with a green circle represent stations the GA system associated with the event, in this case the number of defining phases is 9. Lastly, stations with pdet>0 have text labels.

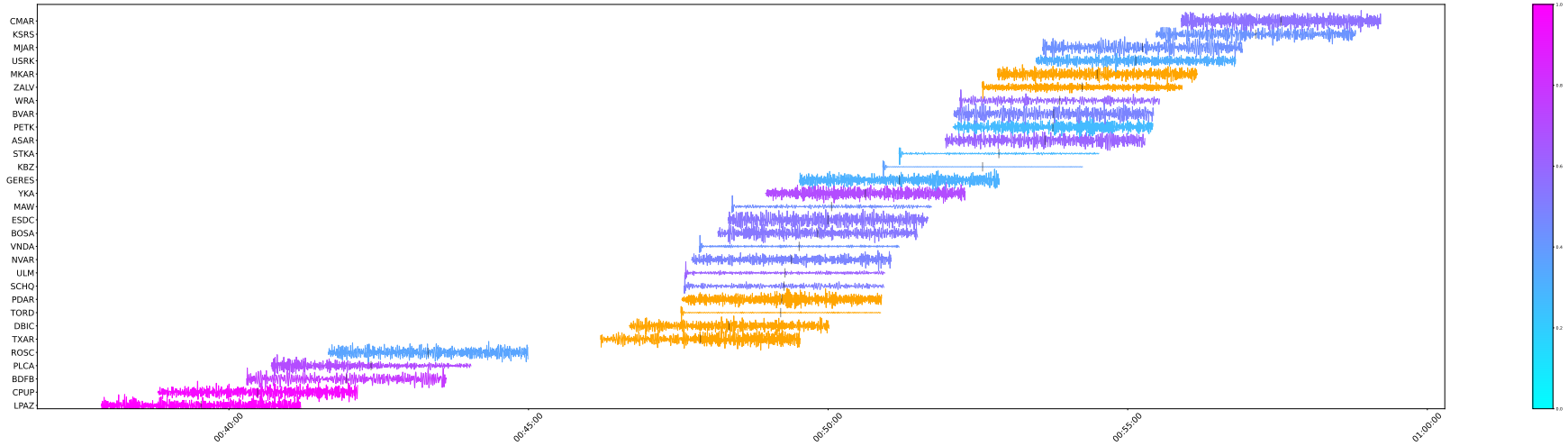


Figure 5-6 Seismic run out plot for the South American true negative event. Station waveforms with pdet>0.25 are shown and color coded by their respective pdet value to the GA event location. Waveforms colored in orange represent GA associated stations.

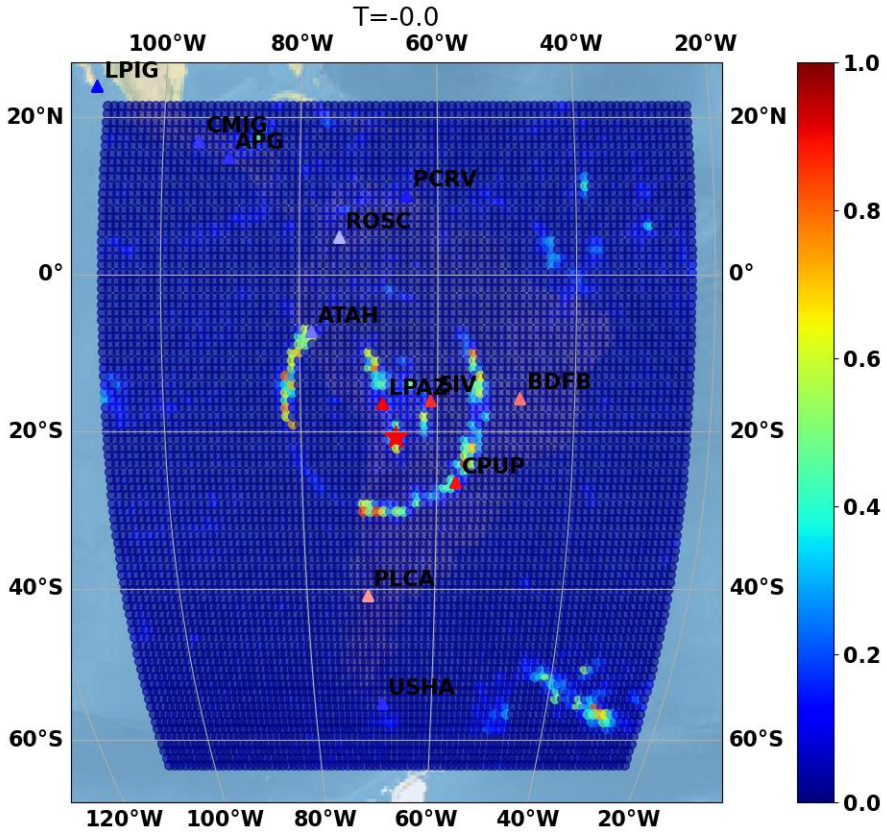


Figure 5-7 GNN probability of detect map for the South American False Positive event (red star).

Origin ID: 6362728 False Negative Alaska Event The next set of Figures (5-8, 5-9, and 5-10) showcase an event built near Alaska with the ORID of 6362728. This event represents a false negative from our GNN model and if we look further into the data structure in Figure 5-9 this becomes more evident. There is a clear, well-aligned phase in the record section for the AKTO (Kazakhstan) station, yet the associated weighting for this station is very low (from figure 5-8. This low weighting of the AKTO station in combination with a lack of other well aligned phases for this event are potential reasons for why the model predicts this event as a non-event. Adjustment of the weighting scheme mentioned above could have potentially changed the model's overall outcome.

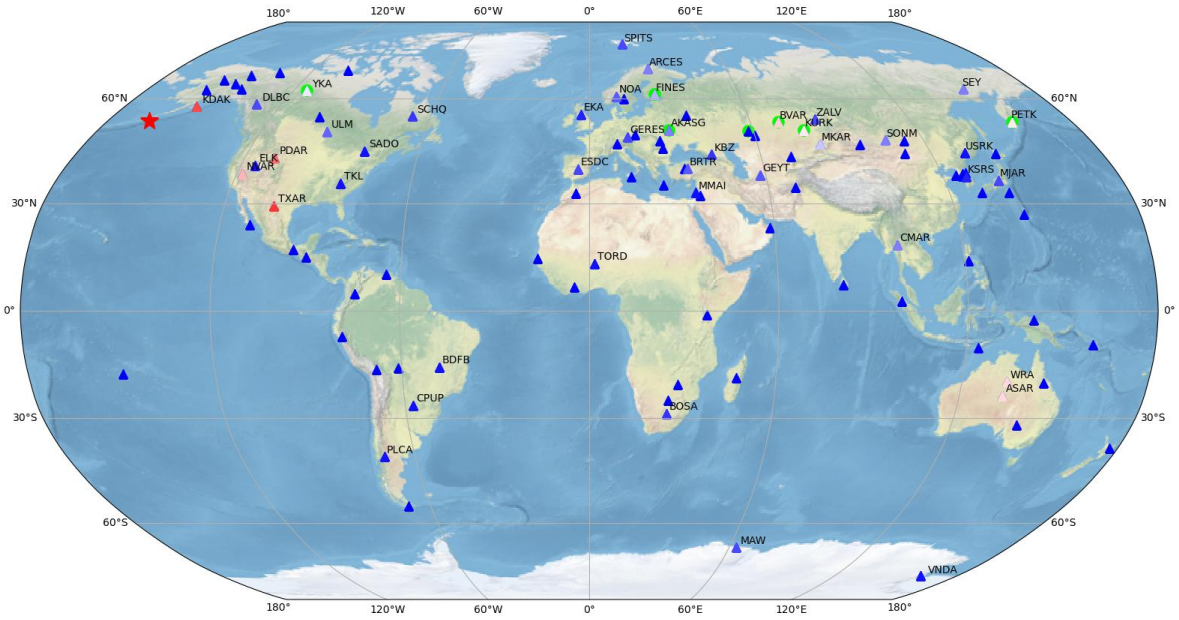


Figure 5-8 World map of the IMS seismic network stations with respect to the 6362728 orid event based around Alaska (red star). Triangles represent potential stations included in our data structure, which are colored by the probability in our pdet model. Blue represents a station would typically have a low probability of detecting an event in the potential event location and red would be a high probability. Stations with a green circle represent stations the GA system associated with the event, in this case the number of defining phases is 7. Lastly, stations with pdet>0 have text labels.

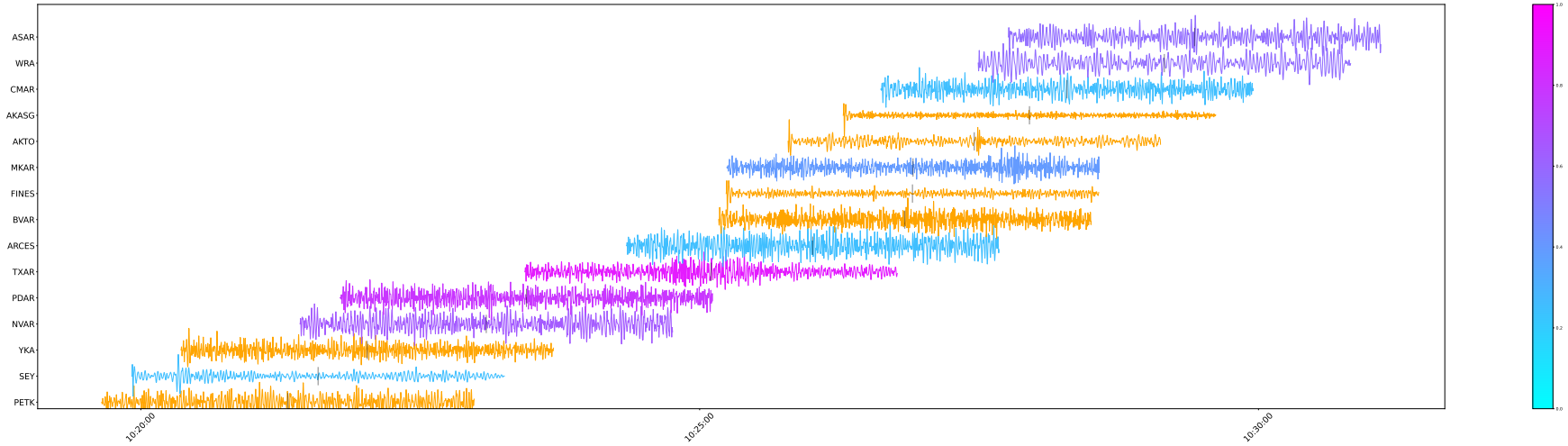


Figure 5-9 Seismic run out plot for the non-existant Alaska event. Station waveforms with pdet>0.25 are shown and color coded by their respective pdet value to the GA event location. Waveforms colored in orange represent GA associated stations.

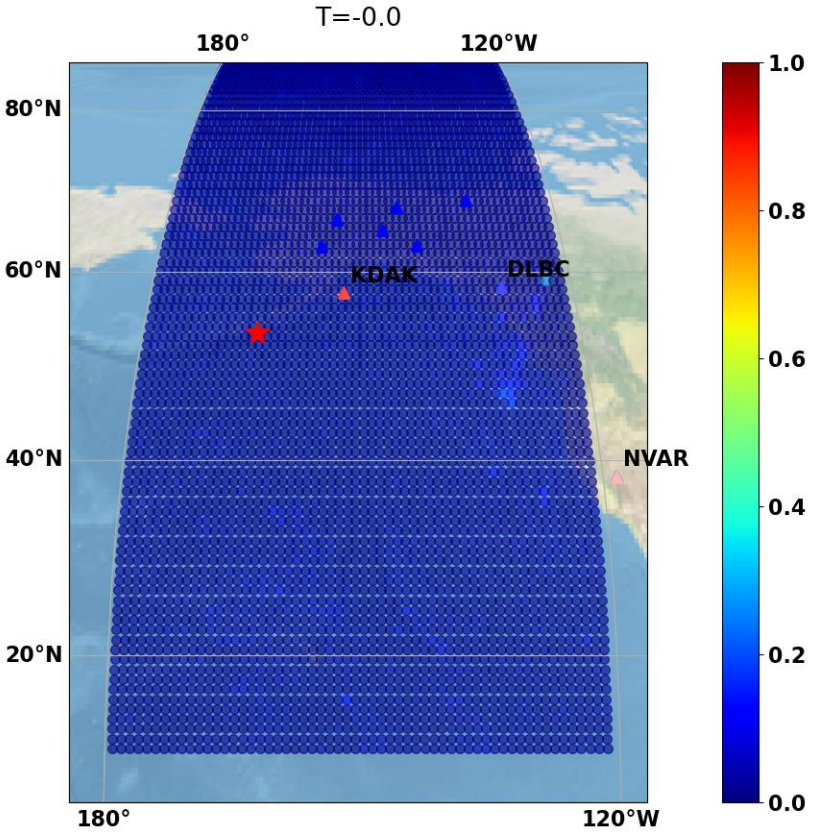


Figure 5-10 GNN probability of detect map for the potential Alaska Event (red star).

Origin ID: 6351738 True Negative East Russia Event The last event demonstrates a true negative event from the GNN model and is located in Eastern Russia. The associated station map (Figure 5-11), seismic record section (Figure 5-12), and GNN probability map (Figure 5-13) can be seen below. This event is remarkably interesting because if we look at the record section, there are two obvious phases that look well aligned to the travel time model (AKASG & GERES), but yet the analyst and GNN both agree that this event was poorly built. One potential explanation could be there is a large set of stations that historically picked up events in this region (16 stations have a pdet higher than 50%), yet it appears only a small number of stations have a clearly visible phase. Analysts are known to use historical station detection priors to inform their decisions and this could be a case where our edge weighting scheme boded well for the end prediction.

We've now discussed 4 GNN model predictions of events built by GA. Two correct (true positive/true negative) and two incorrectly (false positive/false negative) predicted events, both of which have a reasonable explanation that is heavily dependant upon the graph edge weighting scheme. In the correct predictions the weighting scheme can been viewed to benefit the predictive process while simultaneously the incorrect predictions could be caused by the same weighting scheme. In the next steps of this project we will be looking at new ways to improve this weighting

scheme. The current pdet model has limitations especially in regards to geographical areas with sparse seismic activity. One potential improvement could be to utilize pdet as a selection criteria to create the graph topology, but leave the edge weighting constant (or reduced range) as to not overly bias model predictions to low or high pdet stations.

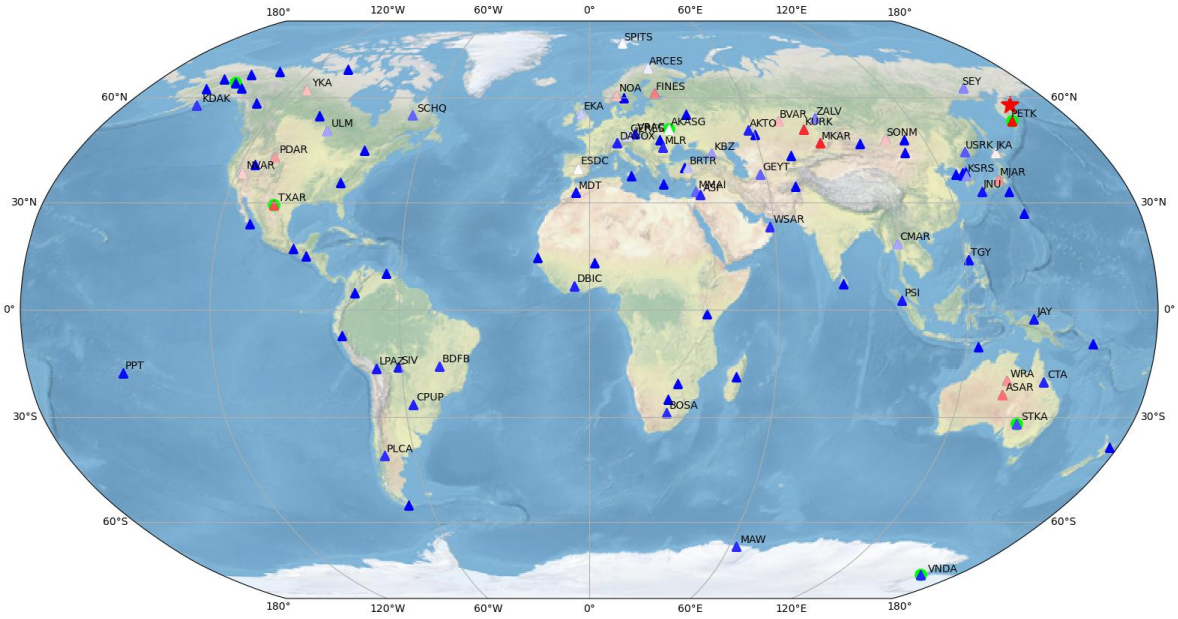


Figure 5-11 World map of the IMS seismic network stations with respect to the 6351738 orid event based in Eastern Russia (red star). Triangles represent potential stations included in our data structure, which are colored by the probability in our pdet model. Blue represents a station would typically have a low probability of detecting an event in the potential event location and red would be a high probability. Stations with a green circle represent stations the GA system associated with the event, in this case the number of defining phases is 8. Lastly, stations with pdet>0 have text labels.

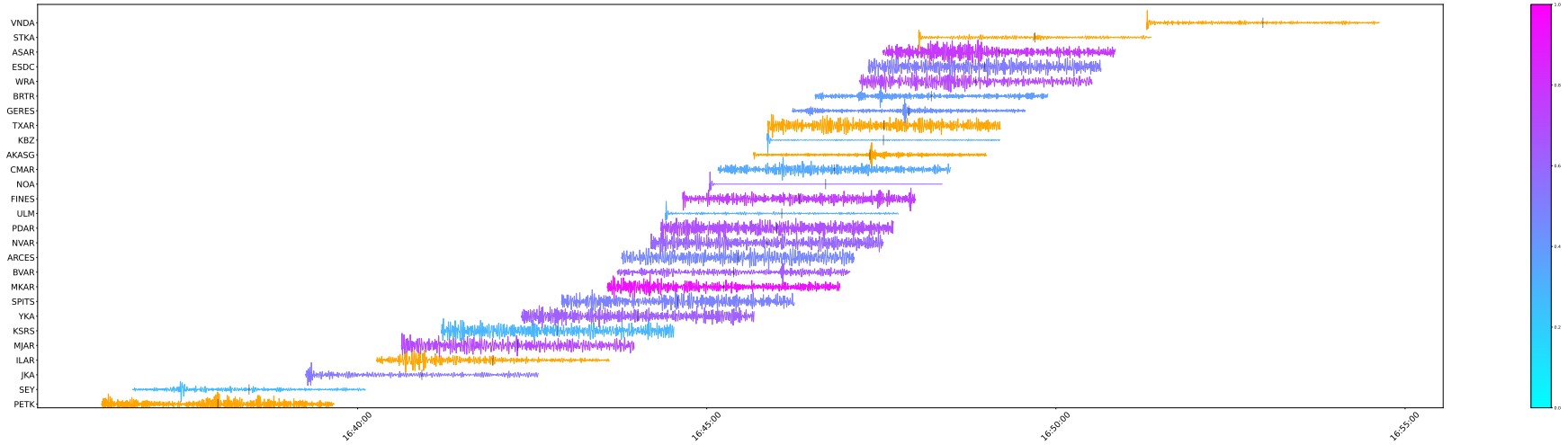


Figure 5-12 Seismic record section for the East Russian false positive event. Station waveforms with pdet>0.25 are shown and color coded by their respective pdet value to the GA event location. Waveforms colored in orange represent GA associated stations.

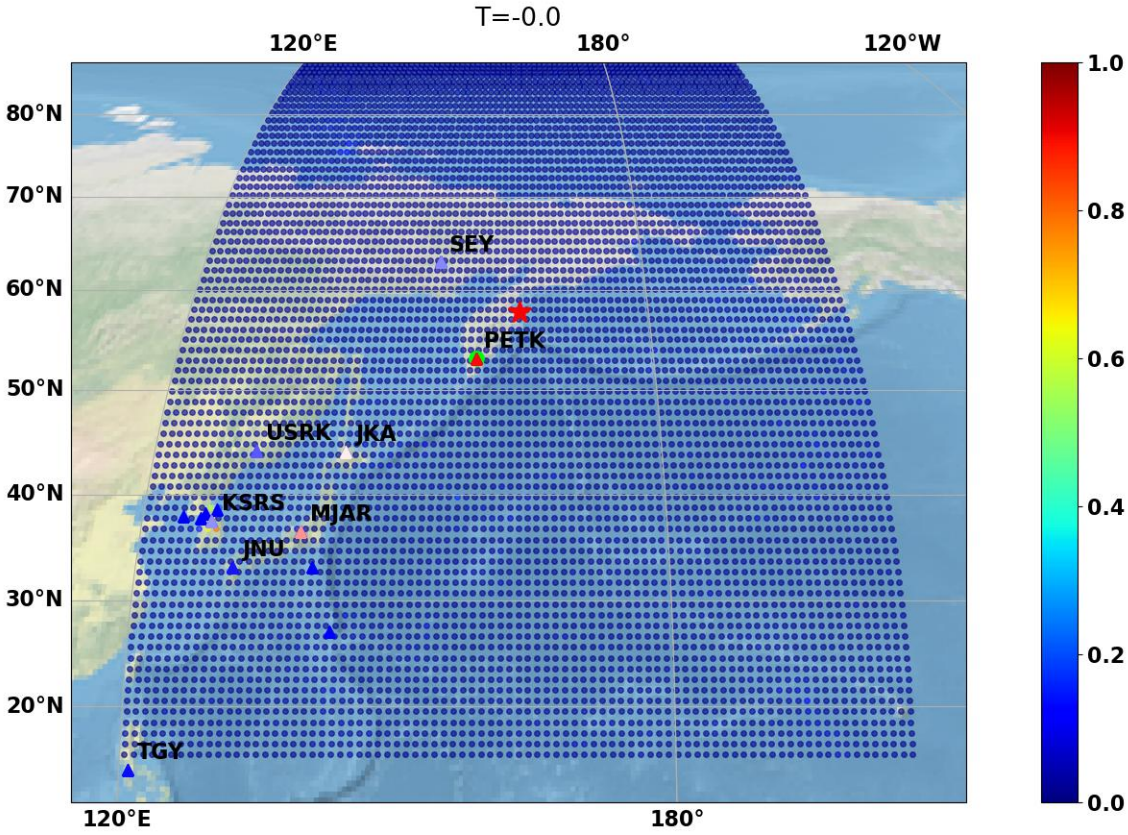


Figure 5-13 GNN probability of detect map for the east Russian true negative event (red star).

6. CONCLUSIONS AND FUTURE WORK

In this report we have summarized the current progress of the GraphAlign project, which has the goal of providing an enhanced, alternative processing pipeline via graph based machine learning. Over the course of this project, we have created an event filtering problem using IMS/IDC data. We have developed a GNN model that is customized to process station waveforms, and we have evaluated this model against an existing baseline model. Our evaluation demonstrates that the GNN model outperforms the baseline in terms of F1, precision, and recall. In addition to aggregate model statistics, we perform an event level error analysis over two correct and two incorrectly predicted samples from our dataset. By visualizing the exact station waveforms (graph node features) and associated station pdet (graph edge weights) we analyze how features in each graph correlate to GNN model predictions. The data driven aspect of the GNN model allows us to create a geographical grid around each potential GA event location to evaluate the spatial fidelity of model predictions. We found our model to have significant radial ambiguity around stations that have high pdet and well aligned phases according to an AK135 travel time model. This leads us to believe our model is heavily biased towards stations that have high pdet and that our edge

weighting scheme has a significant influence on model predictions. This is consistent with how our model makes predictions, which relies on aggregating information from each station proportional to pdet.

In future work, we will include analysis to improve our understanding of what waveform features influence the model's classification decision. In particular, we intend on pursuing the degree of influence individual stations have on the model output, and we will consider readjusting data and model structure if a subset of stations are found to dominate the model's classification decision. In addition, we will continue to consider techniques to improve model performance. Potential areas of improvement include modifying the current edge weighting scheme to emphasize higher quality stations while also allowing rare and low SNR signals to have significant prediction influence. We also intend to evaluate how our model performs on higher quality bulletins such as the LEB or the SNL UGEB bulletin.

REFERENCES

[1] Ian C. Covert, Balu Krishnan, Imad Najm, Jiening Zhan, Matthew Shore, John Hixson, and Ming Jack Po. Temporal graph convolutional networks for automatic seizure detection. In *Proceedings of the 4th Machine Learning for Healthcare Conference*, Proceedings of Machine Learning Research, pages 160–180, 2019.

[2] Ballard S. Young C.J. Draelos, T.J. and R. Brogan. Pedal: A new method for producing automated seismic bulletins: Probabilistic event detection, association, and location. *Bulletin of the Seismological Society of America*, 105:5, 2015.

[3] Timothy Draelos, Michael Procopio, Jennifer Lewis, and Christopher Young. False event screening using data mining in historical archives. *Seismological Research Letters*, 83(2):267–274, 2012.

[4] Matthias Fey and Jan Eric Lenssen. Fast graph representation learning with pytorch geometric. *arXiv preprint arXiv:1903.02428*, 2019.

[5] William L. Hamilton, Rex Ying, and Jure Leskovec. Inductive representation learning on large graphs. *arXiv preprint arXiv:1706.02216*, 2017.

[6] Stephen Heck, Chris Young, and Brogan Chip. Comparing traditional and deep learning signal features for event detection in the utah region. *Submitted to Bulletin of the Seismological Society of America*.

[7] Lisa Linville, Ronald Chip Brogan, Christopher Young, and Katherine Anderson Aur. Global-and local-scale high-resolution event catalogs for algorithm testing. *Seismological Research Letters*, 90:1987–1993, 2019.

[8] Rajat Talak, Siyi Hu, Lisa Peng, and Luca Carlone. Neural trees for learning on graphs. In *Advances in Neural Information Processing Systems*, volume 34, pages 26395–26408, 2021.

[9] Wikipedia. F-score — Wikipedia, the free encyclopedia. http://en.wikipedia.org/w/index.php?title=F-score&oldid=1099088638, 2022. [Online; accessed 14-July-2022].

[10] Jie Zhou, Ganqu Cui, Zhengyan Zhang, Cheng Yang, Zhiyuan Liu, Lifeng Wang, Changcheng Li, and Maosong Sun. Graph neural networks: A review of methods and applications. *arXiv preprint arXiv:1812.08434*, 2018.

DISTRIBUTION

Email—Internal [redacted]

Name	Org.	Sandia Email Address
Technical Library	1911	sanddocs@sandia.gov

Sandia National Laboratories is a multimission laboratory managed and operated by National Technology & Engineering Solutions of Sandia LLC, a wholly owned subsidiary of Honeywell International Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.