Life Could Be a Dream: Somnial Units for Earthquake Signal Detection (Five-Minute Version)

Gregory Lim, Ryan Cheong

May 4, 2025

What is earthquake signal detection?

Every day, seismic signals are recorded by seismometers across planet Earth.

Earthquake signal detection can be defined as the task of determining whether a seismic signal corresponds to an earthquake.

Why earthquake signal detection?

Detecting earthquakes in real time from seismic signals can be useful for early warning and building fortification systems.

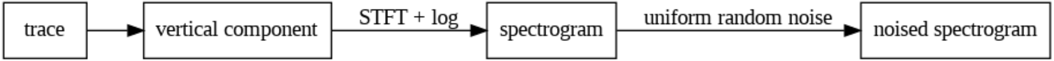
Traditional algorithms which use Short-Term Average over Long-Term Average (STA/LTA) or template matching can struggle in noisy environments [2].

Dataset Source

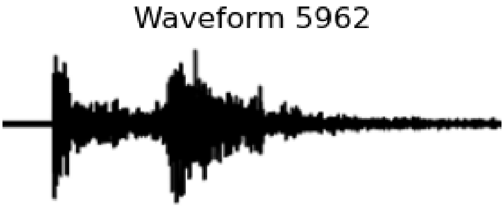
We used the STanford EArthquake Dataset (STEAD).

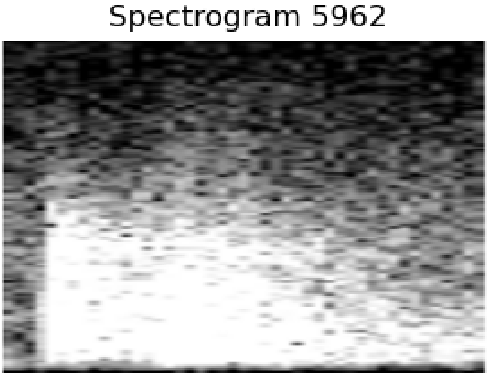
- **Description of elements.** Seismic waveforms measured at various seismic stations, where each waveform is 60s long, sampled at 100Hz, and labelled as either noise (class-0) or earthquake (class-1).
- **Subset used.** Concatenation of chunks 1 and 6, from which 5000 class-0 waveforms and 5000 class-1 waveforms were randomly selected. This was done due to hardware limitations, and for each model to learn about both classes in equal measure.

Dataset Preprocessing (Spectrogram Method)

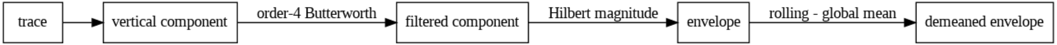


Dataset Preprocessing (Spectrogram Method)

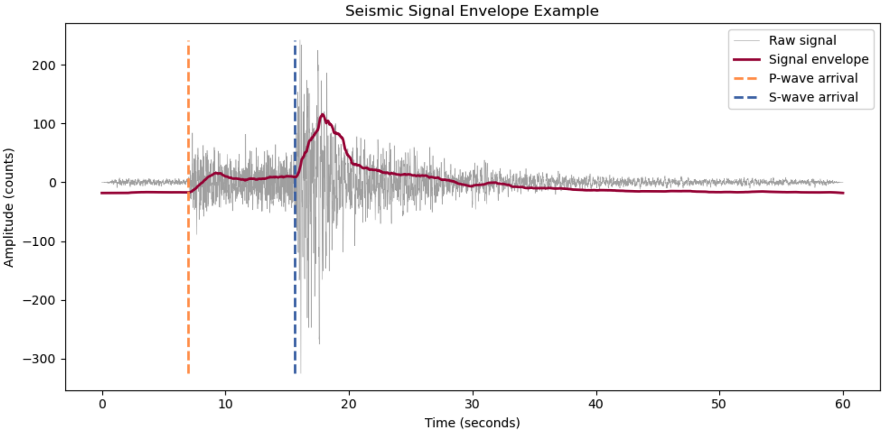




Dataset Preprocessing (Envelope Method)



Dataset Preprocessing (Envelope Method)



Dataset Split

Each preprocessed dataset was split into training, validation, and test sets in a ratio of $8:1:1$, in a stratified way.

Proposed Architectural Unit: Somnial Unit

"Image creation in the brain involves significant neural activity downstream from eye intake, and it is hypothesised that the visual imagery of dreams is produced by activation during sleep of the same structures that generate complex visual imagery in waking perception [1]."

To support our goal of accurate and efficient earthquake signal detection, we propose a novel architectural unit inspired by neurobiological notions of dreaming in animals.

We define the memory buffer as follows.

$$\mathcal{M}_t = \{\mathrm{x}_s : \max(0, t - L) \leq s \leq t\}$$

Proposed Architectural Unit: Somnial Unit

Let $\mathrm{W}_\rho \in \mathbb{R}^{1\times 1\times C\times C}$ be a recollector weight, and $\boldsymbol{b}_\rho \in \mathbb{R}^C$ be a recollector bias. We define the recollector as follows.

$$\rho : \mathbb{R}^{H\times W\times C} \to \mathbb{R}^{H\times W\times C}$$
$$\rho(\mathrm{x}) = \mathrm{x} \circledast \mathrm{W}_\rho + \boldsymbol{b}_\rho$$

Finally, we define the modulator as follows.

$$\mu : \mathbb{R}^{H\times W\times C} \times \mathbb{R}^{H\times W\times C} \to (0,1)^{H\times W}$$
$$\mu(\mathrm{a},\ \mathrm{b}) = \sigma\left(\frac{\langle \boldsymbol{a}_{h,w},\ \boldsymbol{b}_{h,w}\rangle}{\|\boldsymbol{a}_{h,w}\|_2 \|\boldsymbol{b}_{h,w}\|_2}\right)_{(h,w)\in\{1,\ldots,H\}\times\{1,\ldots,W\}}$$

Proposed Architectural Unit: Somnial Unit

Algorithm SomnialUnit(x_t)

---

if training:

  $\mathcal{M}_t \leftarrow \mathcal{M}_t \cup \{\mathrm{x}_t\}$

  $\mathrm{x}_s \leftarrow \mathrm{rand}(\mathcal{M}_t)$

else:

  $\mathrm{x}_s \leftarrow \mathrm{x}_t$

$\hat{\mathrm{x}}_s \leftarrow \rho(\mathrm{x}_s)$

$\mathrm{m} \leftarrow \mu(\hat{\mathrm{x}}_s, \mathrm{x}_t)$

return $\mathrm{m} \odot \hat{\mathrm{x}}_s + (1 - \mathrm{m}) \odot \mathrm{x}_t$

The unit learns how to recollect a previously seen feature map, and blend a recollected feature map with a current feature map.

Training

Table: Training hyperparameters for models.

Batch size	32
Optimiser, learning rate	AdamW, 0.001
Number of epochs	$\{50, 100\}$
Patience	$\{7, 10\}$
S-: memory buffer size	10
CNN: padding, kernel size	1, 3×3
CNN: pooling type, size, stride	Max, 2×2, 2
CNN: dropout	$\{0.25, 0.37, 0.5\}$
RNN: memory vector size	$\{64, 128, 256\}$
RNN: number of recurrent layers	$\{1, 2, 3, 4\}$
RNN: number of linear layers	$\{2, 3\}$
RNN: dropout	$\{0.2, 0.5\}$

Training

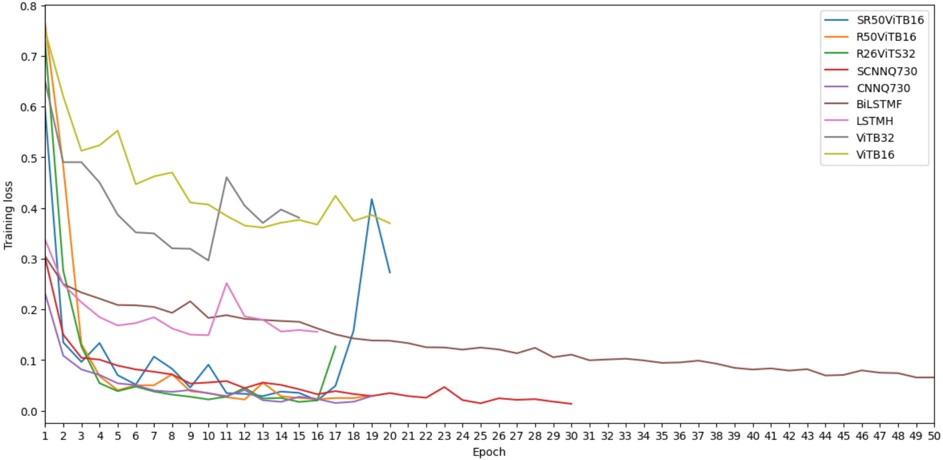


Figure: Training losses for models.

Evaluation

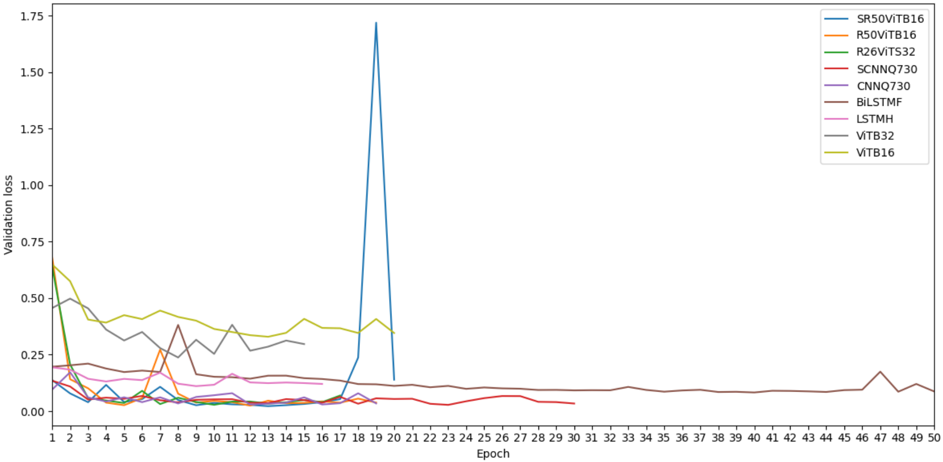


Figure: Validation losses for models.

Evaluation

Table: Evaluation results for models.

ID	Family	Macro F1	Recall
SR50ViTB16	Hybrid	0.99600000	1.00000000
R50ViTB16	Hybrid	0.99499999	0.99600000
R26ViTS32	Hybrid	0.99299994	0.99600000
SCNNQ730	CNN	0.99199997	0.99000000
CNNQ730	CNN	0.98899972	0.98400000
BiLSTMF	RNN	0.96399770	0.97200000
LSTMH	RNN	0.95399540	0.94400000
ViTB32	ViT	0.90175439	0.95200000
ViTB16	ViT	0.85554700	0.91200000

Observations: Model Performance

Adding a somnial unit slightly improved R50ViTB16 and CNNQ730.

SR50ViTB16 achieved a perfect (state-of-the-art) recall!

The best CNN by Rose et. al had a recall around 0.9883 on a different STEAD subset.

Observations: Model Performance

Bidirectional LSTM outperformed unidirectional LSTM.

However, for real-time detection, bidirectional LSTM may not be feasible as it may not have access to the end state earthquake signal that is still in progress.

Observations: Model Performance

Hybrid > CNN, RNN > ViT?

We think the inductive bias introduced by convolution was useful for the task.

Observations: Model Predictions

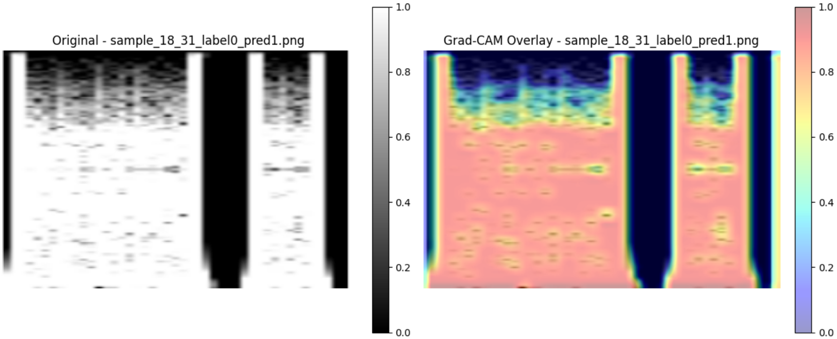


Figure: Grad-CAM visualisation of a false positive predicted by SCNNQ730.

The model seems to focus a lot on certain vertical, high-intensity structures which resemble earthquake phase arrivals.

Observations: Model Predictions

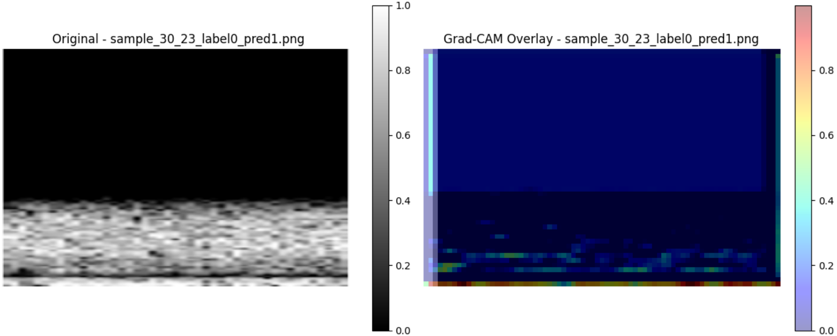


Figure: Grad-CAM visualisation of a false negative predicted by SCNNQ730.

The earthquake manifests as a continuous, horizontal energy band.

The model could have been looking for more pronounced P-wave and S-wave patterns.

Foregone Explorations: Somnial Unit

- Fourier neural operator (FNO).
- Kolmogorov-Arnold layers.
- Cyclic topologies.
- Fourier feature mapping.
- Adversarial boundary refinement.

Suggested Explorations: Somnial Unit

- For a classification task, define a memory buffer for each class.
- Use a different sampling function.
- Noise the generator to introduce regularisation.
- Use a moving average for modulation.

Suggested Explorations: Models

- Online supervised learning.
- Integration into a larger system of real-time predictors.