人工知能学会第二種研究会資料
AIチャレンジ研究会
SIG-Challenge-058-09

Detecting earthquakes: a novel deep learning-based approach for effective disaster response

Muhammad Shakeel[*1] Katsutoshi Itoyama[*1] Kenji Nishida[*1] Kazuhiro Nakadai[*1*2]

[*1] Tokyo Institute of Technology [*2] Honda Research Institute Japan, Co., Ltd.

This research provides an efficient earthquake event classifier that aims to aid robots in automating the conventional disaster response process. Additional sensors and automation are constantly required to react efficiently to a crisis scenario. Deep learning has shown effectiveness in a wide range of applications having a low signal-to-noise ratio, which encouraged us to present a unique 3-dimensional convolutional recurrent network-based earthquake detection method to demonstrate its efficacy in real-time implementation. We train the network using a publicly available earthquake dataset and perform ablations on real-time collected event samples. We preprocess the raw earthquake signals using Log-Mel-based features extraction to retrieve spatial and temporal information. The model extracts the feature information from the low-frequency seismic signals. Furthermore, we propose implementing the model in real-time to distinguish major and minor tremors from seismic signals with an accuracy, sensitivity, and specificity of 98%, 97.7%, and 99.79%, respectively, and a probability threshold of 0.7. Additionally, we develop and validate the model using a two-month continuous data stream from a laboratory-based personal seismometer. The method reliably detects all 63 strong earthquakes recorded by the Meteorological department in Japan from November to December 2019.

1. INTRODUCTION

Over the past decade, tremendous progress has been made in Search, Rescue, and Disaster Robotics[1], and several revolutionary technologies have been developed to enable efficient disaster response. However, considerable effort has to be made in this field to mitigate the effect and destruction caused by natural catastrophes that go beyond human perception. Earthquake recognition continues to be a key and significant aspect for successful crisis response, and the proposed study is an important step forward in the area of "Disaster Robotics".

Identifying earthquakes is a challenging research subject, and a reliable classification method is required to distinguish earthquake waveform from seismological noise. There is a strong potential in using deep learning models in earth observation to describe and identify earthquakes (e.g. [2, 3]) effectively. Effective implementation of these methods is relatively dependent on the availability of high-quality datasets. To address this issue and expedite exploration in this discipline, a worldwide collection of seismic data for machine learning applications has been provided recently. Forming a new dataset, such as STEAD[4], is used in various ways to assist in the development of technologies for the seismological industry. It can also benefit the robotics field in the coming future for efficient disaster mitigation.

Deep learning emerged as a noteworthy area in the field of artificial intelligence and has evolved in various disciplines, including speech recognition, computer vision, and computational linguistics. Availability of enormous processing capabilities[5, 6, 7, 8, 9, 10, 11], deep learning models, implemented as convolutional neural networks (CNN), have demonstrated considerable advances in various classification-related tasks and obtained promising outcomes in a variety of tasks. Thus, an effective learning algorithm

Contact: Muhammad Shakeel, Tokyo Institute of Technology, 2-12-1-W8-30 Ookayama, Meguro-ku, Tokyo, 152-8552, JAPAN, E-mail: shakeel@ra.sc.e.titech.ac.jp

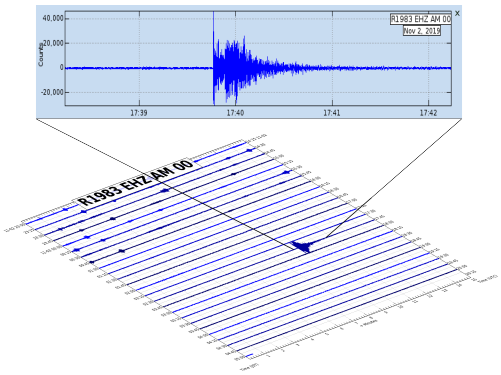

Figure 1: Earthquake detected, by a model learned using 3D-CNN-RNN architecture, within a stream of real-time data available from personal seismometer. The earthquake was reported having a Magnitude of 3.9 in the Japan Meteorological Agency database on November 02, 2019 17:39:29 UTC

is intended to recognize poor signal-to-noise ratio occurrences in a seismograph compared to traditional techniques to automate and enhance the earthquake identification method. With a significant fraction of historical raw earthquake waveforms availability and its promising application area in terms of "Disaster Robotics," we decided to employ deep learning-based algorithms such as convolutional and recurrent neural networks to identify and classify earthquakes. It is possible to convert the seismic signal into a Log-Mel spectrogram. These time-frequency transitions are used as a two-dimensional input to the CNN to extract suitable features from the data.

The three major components of our suggested technique are as follows: Firstly, we generate our custom dataset using the 1.2 million waveforms currently accessible in STEAD. Secondly, we train a model to identify seismic events using a 3-dimensional convolutional recurrent architecture to enhance its accuracy further. Thirdly, we implement and test the system on real-time data trans-

mitted via a personal seismometer to analyze the performance of our classification network. This approach will result in the first implementation of an earthquake monitoring system powered by artificial intelligence for emergency preparedness in the robotics domain. The primary benefit of this technology is that it may be deployed in an environment where robots can be engaged in real-time in the event of a potentially major seismic event. In summary, we assert that our technique has made the following contributions:

1. Propose a three-dimensional (3D) convolution framework for the identification of seismic signals. In a conventional convolutional recurrent network, feature maps are layered to capture the optimum amount of information from the input data; however, we combine separate RNNs on every filter of the last convolutional layer in this study;

2. This is the first study to use a feature extraction technique based on Log-Mel spectrograms to seismic waveforms to the best of our knowledge;

3. Performance evaluation of the system using triangular-shaped filters set to 60 in Log-Mel spectrograms;

4. Extensive analysis of real-time data obtained through a personal seismometer: an earthquake detecting gadget[12].

In this section, we briefly review closely related approaches.

Conventional Methods. Due to their simplicity, STA/LTA[13, 14] and template matching[15, 16] are often used approaches for event identification in seismology. STA/LTA detects earthquakes by comparing short-term average energy to long-term average energy. However, in difficult circumstances with a poor signal-to-noise ratio and time-varying background noise, it generates many false detections. Template matching is a technique for detecting anomalies in candidate waveform data that needs previous knowledge of the candidate waveform data. Cross-correlation algorithms employed in template matching are inefficient and lack generality to handle data in real-time. Both of these systems have a low signal-to-noise ratio and a high false-positive rate, making them impractical for real-time applications, particularly disaster response, where high accuracy is the primary goal.

ConvNetQuake[17]. Convolutional neural networks (CNNs) for seismic data have lately gained popularity as a means of overcoming the limits of traditional techniques. Even though ConvNetQuake is built on the deep topology of a convolutional neural network[9], it is learned on unprocessed waveforms. It does not include feature engineering, which is critical for extracting spatiotemporal information from seismic data. The 2D-CNN framework[9] used in ConvNetQuake serves as a feature extractor in various classification-related tasks, and it is widely recognized as being superior to hand-crafted features. Because of this, a spectrogram of a seismic signal (an intermediary representation of the seismic signal) is required as a two-dimensional input in leveraging the high-dimensional information. In this study, dense CNN models demonstrate good detection accuracy since simulated noisy data was combined with actual data to improve accuracy; however, we demonstrate that state-of-the-art performance may also be obtained if convolutional networks are utilized sensibly alongside recurrent networks on entire real data.

CRED[18]. In this latest research, earthquake detection is treated as a sequence-to-sequence learning problem[19], and two-dimensional convolutional layers are organized in residual blocks, as described in [6], to optimize feature extraction. The authors conducted a comprehensive review of CRED and compared it to established techniques such as STA/LTA and template matching. Their method recognized three orders of magnitude more events than STA/LTA and reduced the false positive rate, demonstrating the deep learning architecture's efficacy and reliability. However, in this research, two-dimensional convolutional neural networks are layered with RNN(Bi-LSTM) layers to extract local features and model hidden temporal relationships, respectively. In general, super deep CNN architectures[6],[9] outperform regular CNN models. However, expanding the receptive area of a 3-dimensional convolutional recurrent network by extracting spectral and temporal feature maps may also give state-of-the-art results.

2. PROPOSED METHOD

We propose a three-dimensional (3D) convolutional architecture for earthquake detection and expand the usage of Log-Mel spectrograms to extract features from seismic signals to attain greater temporal and spatial resolution. The 3D-CNN architecture is employed in various fields, including human action detection[20] for video processing applications, audio-visual recognition[21], and, more lately, speaker verification tasks that do not need text[22]. We introduce a 3D-CNN-RNN framework in this study to leverage the temporal and spatial characteristics of seismic waveforms. In comparison, a traditional CNN-RNN architecture stacks feature maps together. In contrast, we implement distinct RNNs to every filter of the final convolution operation to capture most temporal information from the seismic waves. Similar to ConvNetQuake and CRED, we use Log-Mel energies to balance frequency and temporal characteristics.

2.1 3D CONVOLUTIONAL NEURAL NETWORKS

In principle, the 3D-CNN is the expansion of 2D-CNN. When 2D-CNNs are employed on 2D feature maps we can only extract the information in spatial domain. In the case of seismic events, if two events occur at the same time it is most desirable to extract the temporal information related to actual earthquake signal. The said information can only be inferred in temporal domain to capture the changing behaviour of the signal. To tackle the aforementioned issue, we propose to perform 3D convolutions in convolution stages which is desirable in computing the features from spatial and temporal perspective. In theory, 3D convolution is performed by convolving a 3D kernel (filter) and stacking multiple adjacent frames in a form of cube. In this topology, stacked frames in the previous layer are connected to feature maps in the convolution layer, thereby capturing temporal information. In formulation, the value of any unit at position (x,y,z) in the jth feature map in the ith layer, denoted as $u_{ij}^{x,y,z}$, is given by

$$u_{ij}^{x,y,z} = g\left(b_{ij} + \sum_{m}\sum_{p=0}^{P_i-1}\sum_{q=0}^{Q_i-1}\sum_{r=0}^{R_i-1} w_{ijm}^{p,q,r}\, u_{(i-1)m}^{(x+p)(y+q)(z+r)}\right), \quad (1)$$

where g is the activation function, b_{ij} is the bias for the feature map, R_i is the size of the 3D kernel along the time axis, $w_{ijm}^{p,q,r}$ is the (p,q,r)th value of the kernel connected to mth feature in the previous layer.

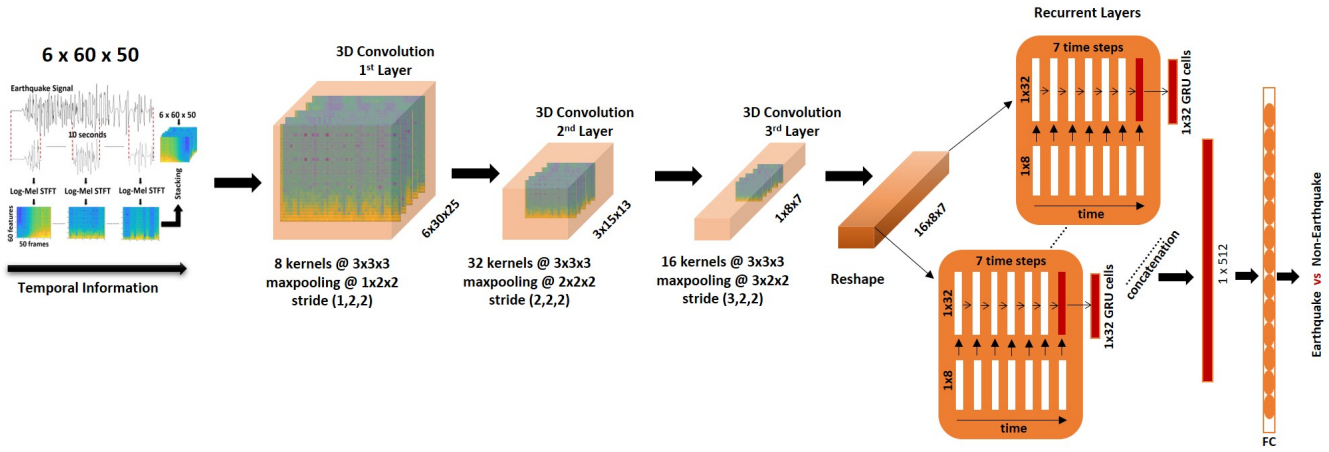

Figure 2: 3D-CNN-RNN combination for earthquake detection. Combination of three convolutional layers and sixteen separate GRUs for each filter in the final convolutional layers is used in above architecture. Each feature map of the last layer is fed to 32 GRU cells in the sixteen recurrent layers. Softmax output layer acts as a fully connected (FC) last layer to classify the events in to earthquake and not-earthquake. Input is a stack of 10-second ground motion clips.

2.2 RECURRENT NEURAL NETWORK (RNN)

The RNNs are important in sequence-to-sequence learning tasks as they retain relations among inputs while training. In practical applications[23] *gated* RNNs, also known as gated recurrent units or GRUs, are used most effectively because they have the derivatives that neither vanish nor explode while creating paths through time. In many sequential tasks GRUs are used because they have the capability of simultaneously controlling the forgetting factor and the decision to update the state unit with a single gating unit. The update equations[24] for GRUs are as follows:

$$h_i^{(t)} = u_i^{(t-1)} h_i^{(t-1)} + (1-u_i^{(t-1)})\sigma\left(b_i + \sum_j U_{i,j} x_j^{(t-1)} + \sum_j W_{i,j} r_j^{(t-1)} h_j^{(t-1)}\right), \quad (2)$$

where u stands for the update gate and r for the "reset" gate. Their value is separately defined as:

$$u_i^{(t)} = \sigma\left(b_i^u + \sum_j U_{i,j}^u x_j^{(t)} + \sum_j W_{i,j}^u h_j^{(t)}\right), \quad (3)$$

$$r_i^{(t)} = \sigma\left(b_i^r + \sum_j U_{i,j}^r x_j^{(t)} + \sum_j W_{i,j}^r h_j^{(t)}\right), \quad (4)$$

In GRUs reset and update gates can independently "ignore" some parts of the state vector making it dynamically control the time scale and forgetting behaviour of different units.

2.3 3D CNN-RNN ARCHITECTURE

A range of CNN-RNN topologies may be designed using the 3D convolution and Recurrent Neural Network methods described earlier. We explain a 3D-CNN-RNN framework that we constructed for an earthquake classification task in the next section. Three convolutional layers are used in this design, as seen in Fig.2. We suggest a $3 \times 3 \times 3$ field of view (3×3 in the space, 3 in the time axis). Furthermore, we apply max pooling operation in each convolutional layer, such as: $(1 \times 2 \times 2)$ for the first, $(2 \times 2 \times 2)$ for the second, and $(3 \times 2 \times 2)$ for the final layer. Strided convolutions combined with a max-pooling layer operation enable us to downsample the signals through each dimension, lowering the computational complexity while rapidly increasing the field of view across the original signal. To retrieve temporal and spatial details, indirect connections are employed. Moreover, we add the ReLU (rectified linear unit) activation function $g(.) = max(0,.)$ to each convolutional layer, using its backpropagation rule to cancel out any gradient elements that are smaller than zero. To optimize the learning rate, batch normalization[25] is performed individually in each layer. To prevent overfitting, a dropout rate of 0.5 is employed. The Xavier initialization[26] technique is adopted for randomly initializing the training weights. We apply several GRUs to the final convolutional layer's feature maps to extract temporal information from seismic waves. Since the final layer has sixteen filters (kernels), each filter is represented by a distinct GRU, resulting in sixteen GRUs. We built seven recurrent layers among each feature map, where seven is the number of time steps mapped from the 50 timestamps in the original spectrogram. The recurrent network comprises 32 GRU cells per layer. Each recurrent layer employs a many-to-one structure, and the output of all layers is concatenated and then fed into a fully linked layer. Using the backpropagation technique, optimization is performed simultaneously on 3D-CNN and RNN architectures. As a final layer, a fully connected softmax layer comprising two nodes is implemented to categorize occurrences as earthquakes or non-earthquake.

3. DATA AND METHODS

3.1 PROPERTIES OF DATASET

Earthquakes occur when rapid movements across active faults release stored elastic energy in the rocks, generating shock waves that flow through the ground. Each day, thousands of earthquake events occur worldwide, of which fifty are intense enough to be experienced (magnitude > 2.5)[27]. These seismic signals are recorded at local seismic stations, and there was a need for a single

universal database. To anticipate the potential challenge, Stanford researchers have released a database featuring seismic waves from throughout the world from January 1984 to August 2018. STanford Earthquake Dataset (STEAD)[4] is a publicly accessible online database for research purposes. It is classified into two types: localized earthquakes and seismic noise (free of earthquake signals). It comprises *sim* 1,050,000 three-component seismograms, with 6000 samples per waveform in the east-west, north-south, and vertical directions. However, we selected the vertical component of the waveform since our model would be validated using a real-time personal seismometer that can only monitor the vertical component. Each earthquake event contains 32 properties, one of which is 'source magnitude,' which is critical for balancing the dataset. The majority of earthquake waveforms presented in the database have a magnitude of < 2.5. We utilized only manually selected waveforms, i.e., those provided by seismic stations, and ignored any waveforms determined by computerized algorithms. We picked distinct waveforms for the training and test sets. All waveforms (earthquake and non-earthquake) are classified according to their stated properties. The waveforms in the given dataset have been detrended (i.e., the mean has been removed), resampled at 100 Hz, and then filtered using a 1-45 Hz bandpass filter. To address data disparity and increase generalization, we omitted specific waveforms and created our dataset. There are 108,680 and 46,561 waveforms used to train and test the model, respectively. The dataset for training and test set makes up a composition of 70% and 30% independently. A one-hot encoding of the training and test set is performed, and each waveform is labeled with 1 if an earthquake event is present and 0 if no earthquake event (seismic noise). The statistics for the development and evaluation sets are presented in Table 1 and 2.

Table 1: Dataset orientation for Earthquake Waveforms.

Earthquake Magnitudes	**Earthquake Waveforms** (Training Set)	**Earthquake Waveforms** (Test Set)
> 0	10868	4656
> 1	10868	4656
> 2	10868	4656
> 3	10868	4656
> 4	9923	4252
> 5	883	378
> 6	61	26
> 7	1	0
Total	54340	23280

3.2 DATA REPRESENTATION: FEATURE EXTRACTION

We propose that Mel Spectrograms be employed as a data representation of seismic waveforms at the frame level. Mel-spectrograms are constructed by incorporating linearly spaced triangular-shape filters in the Mel scale. Further, we obtain the log-energies in the Mel scale. The method is very similar to MFCCs, except that the Discrete Continuous Transform (DCT) is not used in this case. We apply this approach to seismic data to extract frequency components by applying triangular-shape filters while maintaining the maximum temporal information. We divided 60-second ground motion data into six 10-second segments. Overlapping windowed signals represent the temporal features. Using the sequence of these window signals, a single spectrogram of a ten-second clip is generated. The signal is framed using a 400ms window length. Using a Fast Fourier Transform (FFT) with 64 bins (zero padded) and a hamming window with a 50% signal overlap, we can calculate a Short-Time Fourier Transform (STFT) with zero padding. The complex spectrum of a seismic signal $s(t)$ may be expressed as follows:

Table 2: Dataset orientation for seismic noise waveforms.

Non-Earthquake Waveforms (Training Set)	**Non-Earthquake Waveforms** (Test Set)
54340	23281

$$S(n,f) = |S(n,f)|e^{j\theta(n,f)} \tag{5}$$

where $|S(n,f)|$ is the magnitude and $\theta(n,f)$ as the phase spectrum for frequency f in frame n.

Mel-scale is extensively used in speech recognition tasks as one of the feature extraction method. However, we propose this scale can also be utilized for seismic signals. Several analytical expressions exist to convert Hertz-scale frequencies to Mel-scale and one of the common relation as given by D. O'Shaughnessy is used in our study to extract features for network training.

$$m = 2595\log_{10}(1 + f/700) \tag{6}$$

and filter bandwidths computed using,

$$f = 700(10^{m/2595} - 1) \tag{7}$$

Linearly spaced triangular-shape filters in Mel-scale are constructed using the aforementioned equation. The number of filters are set to 60, that act as spectral features for our problem. Finally the magnitude values are then converted into log magnitudes and were normalized as input to the network.

$$S(n,f) = \log(|S(n,f)|) \tag{8}$$

Each feature map has the dimensionality of $\delta \times 60 \times 50$. δ is the number of seismic signal clips, 60 are the number of filters in Mel-scale and 50 is the window frames used to calculate the STFT. Finally the input feature for 3D-CNN RNN architecture is $6 \times 60 \times 50$ and feature extraction process as employed on raw seismic waveform is shown in Fig.3.

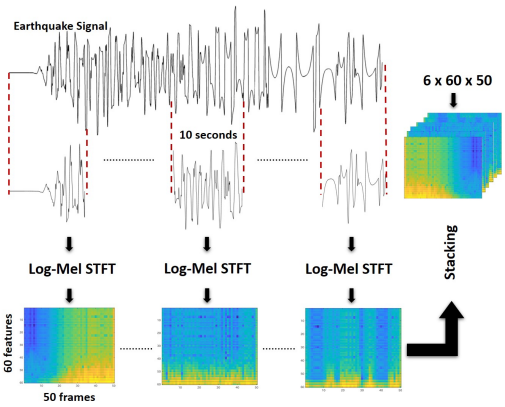

Figure 3: Data input process for 3D-CNN RNN network: Feature extraction using Log-Mel spectrograms

4. EXPERIMENTS

4.1 EVALUATION METRICS

We report the performance evaluation metric of our earthquake detector classifier in terms of Precision (sensitivity), Recall (specificity) and F-score (accuracy) using (9), (10) and (11) respectively. Sensitivity is defined as the number of earthquakes predictions that are accurate, Specificity is defined as the number of instances that are accurately predicted, and the F-score is the harmonic mean of sensitivity and specificity. We calculate these scores using a decision threshold value (thresh) for output probabilities.

$$Precision = TP(thresh)/(TP(thresh)+FP(thresh)) \quad (9)$$

$$Recall = TP(thresh)/(TP(thresh)+FN(thresh)) \quad (10)$$

$$Fscore = 2 \times Precision \times Recall/(Precision+Recall) \quad (11)$$

where, TP denotes true positives, FP denotes false positives, and FN are false negatives. TP=1 and FP=0 in a perfect classifier.

4.2 TRAINING

We used 3D-CNN architecture in all it's essence i.e. 3-dimensional convolution and three CNN layers. We trained the model using a drop-out rate of 0.5. In binary classification problem, as in our study, we employed binary cross-entropy loss on the softmax function. We employed RMSProp optimizer[24] with initial learning rate of 10^{-3} and a momentum decay of 0.9 to avoid the gradient vanishing/exploding issues, and to increase the learning rate. We use batch size of 64 i.e. 64 training examples to train our models. Furthermore, as a training policy, we split the training set into 97% of the total training examples whereas the remainig 3% of the examples are used as a validation set to monitor the validation loss and observe the training process. The data in the training set is shuffled randomly and to overcome the overfitting problem and maintain generalization we stopped the training after 100 epochs. Data augmentation strategy is not applied because of the availability of large amount of data. During testing, we selected the best model having highest accuracy on the validation set and calculated the predictions. Tensorflow is used to implement the model. We train our networks on a single NVIDIA V100 GPU. The learning time for our proposed architecture is 24 hours that includes feature extraction, training, testing and predicting the probabilities.

4.3 DETECTION ACCURACY

The detection accuracy of the algorithm is the percentage of waveforms correctly classified as earthquake or seismic noise. We selected the best model based on the tuned hyperparameters to detect an earthquake event. Regardless of the threshold choice, our earthquake detector successfully detects 22380 earthquake events as catalogued in STEAD and misclassifies 23 of the earthquake waveforms as seismic noise, whereas it correctly classifies 23258 noise events and misclassifies 900 as earthquakes. In summary (see Fig.4), our algorithm predicts 22380 true positives, 23 false negatives, 900 false positives, and 23258 true negatives. Therefore, the sensitivity (fraction of earthquake events that are true events) is 96%, and specificity (fraction of true events correctly detected) is 99.99%. However, with a probability threshold of 0.7, sensitivity of the network is increased to 97.70% and specificity decreased to 99.79%.

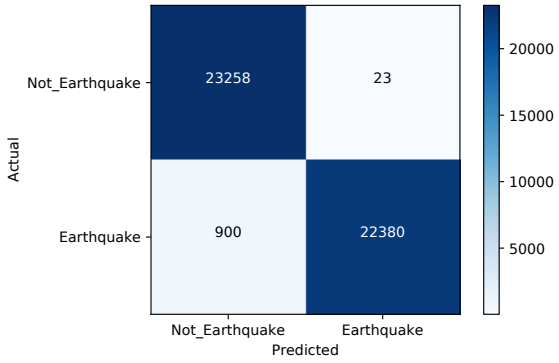

Figure 4: Confusion Matrix Regardless of Threshold Value

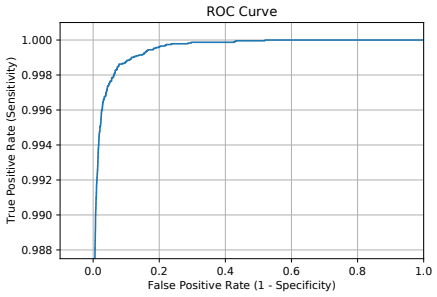

Figure 5: Receiver Operating Characteristics (ROC)

Building a robust deep learning model typically requires a large amount of labelled training data as discussed in **ConvNetQuake**[17] and **CRED**[18]. In general our classifier has superior performance as compared to other two methods. In **ConvNetQuake** authors report the classifier has a precision of 94% and recall of 100% while in **CRED** the model has a precision of 96% and recall of 99%. In **CRED** model is trained using 500000 seismographs (250000 as earthquake events and 250000 as noise waveforms) and with a denser network, whereas in **ConvNetQuake** the network was trained using 702748 waveforms (2709 events and 700,039 noise windows) whereas noise windows were synthetically generated. However, we have demonstrated that state-of-the art results can also be achieved by using a smaller dataset and with a less denser network i.e. only 3 convolutional

layers. Our model is learned using a real and smaller dataset, making it to generalize better in real-time scenarios. The superior performance of our method is due to its reliance on both spectral and temporal feature extraction of the signal rather than the waveform and spectral features only. Hence, denoising the signal as input to the learned model can help reduce the false positive rate as large amount of seismic noise is present in the real-time environment. Furthermore, for fair comparison in general, the datasets for the baseline and proposed methods should have been same but due to non-availibility of datasets used by ConvNetQuake and CRED; we had to train our model using an efficient dataset i.e. STEAD and is publicly available for an acceptable comparison in future studies.

CONCLUSION

We presented a unique autonomous system capable of detecting earthquakes using a learned system based on deep neural networks and a personal seismometer. This breakthrough significantly expands the application fields for artificial intelligence systems, including seismology and disaster robotics. The described method is not dependent on an alert center and may function effectively as an earthquake detection tool in metropolitan areas on its own. While the current study concentrated on detection, future work will examine how artificial intelligence-based algorithms may be utilized to improve the reaction time of an earthquake warning system.

References

[1] S. Tadokoro, Ed., *Disaster Robotics*. Springer International Publishing, 2019. [Online]. Available: https://doi.org/10.1007/978-3-030-05321-5

[2] W. Zhu and G. C. Beroza, "PhaseNet: a deep-neural-network-based seismic arrival-time picking method," *Geophysical Journal International*, vol. 216, no. 1, pp. 261–273, 10 2018. [Online]. Available: https://doi.org/10.1093/gji/ggy423

[3] S. Qu, Z. Guan, E. Verschuur, and Y. Chen, "Automatic high-resolution microseismic event detection via supervised machine learning," *Geophysical Journal International*, vol. 218, no. 3, pp. 2106–2121, 06 2019. [Online]. Available: https://doi.org/10.1093/gji/ggz273

[4] S. M. Mousavi, Y. Sheng, W. Zhu, and G. C. Beroza, "Stanford earthquake dataset (stead): A global data set of seismic signals for ai," *IEEE Access*, vol. 7, pp. 179 464–179 476, 2019.

[5] G. Huang, Z. Liu, and K. Q. Weinberger, "Densely connected convolutional networks," *CoRR*, vol. abs/1608.06993, 2016. [Online]. Available: http://arxiv.org/abs/1608.06993

[6] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *CoRR*, vol. abs/1512.03385, 2015. [Online]. Available: http://arxiv.org/abs/1512.03385

[7] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Computer Vision and Pattern Recognition (CVPR)*, 2015. [Online]. Available: http://arxiv.org/abs/1409.4842

[8] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *CoRR*, vol. abs/1409.1556, 2014.

[9] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *NIPS*, 2012.

[10] S. Ren, K. He, R. B. Girshick, and J. Sun, "Faster R-CNN: towards real-time object detection with region proposal networks," *CoRR*, vol. abs/1506.01497, 2015. [Online]. Available: http://arxiv.org/abs/1506.01497

[11] R. B. Girshick, "Fast R-CNN," *CoRR*, vol. abs/1504.08083, 2015. [Online]. Available: http://arxiv.org/abs/1504.08083

[12] J. Diaz, M. Schimmel, M. Ruiz, and R. Carbonell, "Seismometers within cities: A tool to connect earth sciences and society," *Frontiers in Earth Science*, vol. 8, Feb. 2020. [Online]. Available: https://doi.org/10.3389/feart.2020.00009

[13] R. Allen, "Automatic phase pickers: Their present use and future prospects," *Bulletin of the Seismological Society of America*, vol. 72, no. 6B, pp. S225–S242, 12 1982.

[14] M. Withers, R. Aster, C. Young, J. Beiriger, M. Harris, S. Moore, and J. Trujillo, "A comparison of select trigger algorithms for automated global seismic phase and event detection," *Bulletin of the Seismological Society of America*, vol. 88, no. 1, pp. 95–106, 02 1998.

[15] S. J. Gibbons and F. Ringdal, "The detection of low magnitude seismic events using array-based waveform correlation," *Geophysical Journal International*, vol. 165, no. 1, pp. 149–166, 04 2006. [Online]. Available: https://doi.org/10.1111/j.1365-246X.2006.02865.x

[16] D. R. Shelly, G. C. Beroza, and S. Ide, "Non-volcanic tremor and low-frequency earthquake swarms," *Nature*, vol. 446, no. 7133, pp. 305–307, 2007. [Online]. Available: https://doi.org/10.1038/nature05666

[17] T. Perol, M. Gharbi, and M. Denolle, "Convolutional neural network for earthquake detection and location," *Science Advances*, vol. 4, no. 2, 2018. [Online]. Available: https://advances.sciencemag.org/content/4/2/e1700578

[18] S. M. Mousavi, W. Zhu, Y. Sheng, and G. C. Beroza, "Cred: A deep residual network of convolutional and recurrent units for earthquake signal detection," *Scientific reports*, vol. 9, no. 1, pp. 10 267–10 267, Jul 2019, 31311942[pmid]. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/31311942

[19] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," *CoRR*, vol. abs/1409.3215, 2014. [Online]. Available: http://arxiv.org/abs/1409.3215

[20] S. Ji, W. Xu, M. Yang, and K. Yu, "3d convolutional neural networks for human action recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 1, pp. 221–231, Jan 2013.

[21] A. Torfi, S. M. Iranmanesh, N. M. Nasrabadi, and J. M. Dawson, "Coupled 3d convolutional neural networks for audio-visual recognition," *CoRR*, vol. abs/1706.05739, 2017. [Online]. Available: http://arxiv.org/abs/1706.05739

[22] A. Torfi, N. M. Nasrabadi, and J. M. Dawson, "Text-independent speaker verification using 3d convolutional neural networks," *CoRR*, vol. abs/1705.09422, 2017. [Online]. Available: http://arxiv.org/abs/1705.09422

[23] K. Jarrett, K. Kavukcuoglu, M. Ranzato, and Y. LeCun, "What is the best multi-stage architecture for object recognition?" in *2009 IEEE 12th International Conference on Computer Vision*, Sep. 2009, pp. 2146–2153.

[24] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. The MIT Press, 2016.

[25] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," *CoRR*, vol. abs/1502.03167, 2015. [Online]. Available: http://arxiv.org/abs/1502.03167

[26] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," ser. Proceedings of Machine Learning Research, Y. W. Teh and M. Titterington, Eds., vol. 9. Chia Laguna Resort, Sardinia, Italy: PMLR, 13–15 May 2010, pp. 249–256. [Online]. Available: http://proceedings.mlr.press/v9/glorot10a.html

[27] P. M. Shearer, *Introduction to Seismology*. Cambridge Univ. Press, 2009.