

2019-08-19

Comparison of convolutional neural networks with matched-filtering for detection of induced seismicity

Vragov, Volodymyr

Vragov, V. (2019). Comparison of convolutional neural networks with matched-filtering for detection of induced seismicity (Master's thesis, University of Calgary, Calgary, Canada). Retrieved from <https://prism.ucalgary.ca>.

<http://hdl.handle.net/1880/110751>

Downloaded from PRISM Repository, University of Calgary

UNIVERSITY OF CALGARY

Comparison of convolutional neural networks with matched-filtering for detection of induced
seismicity

by

Volodymyr Vragov

A THESIS
SUBMITTED TO THE FACULTY OF GRADUATE STUDIES
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE
DEGREE OF MASTER OF SCIENCE

GRADUATE PROGRAM IN GEOSCIENCE

CALGARY, ALBERTA

AUGUST, 2019

© Volodymyr Vragov 2019

UNIVERSITY OF CALGARY

FACULTY OF GRADUATE STUDIES

The undersigned certify that they have read, and recommend to the Faculty of Graduate Studies for acceptance, a thesis entitled "Comparison of convolutional neural networks with matched-filtering for detection of induced seismicity" submitted by Volodymyr Vragov in partial fulfillment of the requirements for the degree of Masters of Science.

Supervisor: Dr. David William
Eaton
Department of Geoscience

Dr. Daniel Osvaldo Trad
Department of Geoscience

Dr. Edward Stephen Krebes
Department of Geoscience

Date

Abstract

The problem of extracting weak earthquake signals from continuous waveforms data recorded by networks of seismic sensors, referred to as earthquake detection, is a challenging and critical task in seismology. Waveform cross-correlation (matched-filtering) is a widely used method to detect weak earthquake signals with waveforms similar to those of known events. Even using this method, earthquake catalogs are often limited and incomplete, so there is a need for a more general detector. Because earthquakes occur infrequently, detection algorithms must be capable of quick processing of months to years of continuous data dominated by noise. To address these challenges, we propose to use convolutional neural networks, a new detection method that utilizes large seismic datasets to perform computationally efficient search to identify events in the continuous data. We set up an algorithm as a supervised machine learning problem and test its performance against matched-filtering based detector on synthetic and Kaybob-Duvernay region data.

In this work, I introduce a DuverNet, which is an optimized convolutional neural network for the detection of induced seismicity in the Kaybob-Duvernay production region. First, I test two different convolutional neural network architectures: ConvNetQuake and VGG-Junior. Second, I test two different loss functions: cross-entropy loss and focal loss. My thesis is the first time focal loss is used to tackle class imbalance problem in earthquake detection. Focal loss helps tackling the problem of class imbalance and allows to achieve higher accuracy for convolutional neural network detectors. Third, I compare performance of matched-filtering and DuverNet. After allowing time for appropriate training, DuverNet performs best in computational runtime and memory use. Moreover it delivers superior detection performance compared to the matched-filtering detector. Synthetic data testing illustrates that DuverNet better generalizes to previously unseen events and is found to be better at detecting more events at lower signal-to-noise ratio compared to the matched-filtering method. Finally, my thesis introduces a novel dataset collected by the 6 UC/DSA array stations installed by Nanometrics for the University of Calgary.

Acknowledgements

The journey through this thesis over the past 3 years would not have been possible with the consistent guidance, support, and understanding from numerous individuals across the globe. First of all, I would like to thank my advisor Dr. David Eaton, who gave me the freedom and opportunity to pursue my interests in the domains formerly unimaginable. I have learned a lot from working with Professor Eaton, both about microseismic, and about being an effective communicator. One of the things I appreciate most about David is that not only is he a dedicated advisor, but he is also acts as a champion and an advocate for his students, always encouraging us to present and share our work.

I would like to thank my research partners and officemates, without whom my time at the University would be less productive and significantly less entertaining. Thank you to Anton Biryukov for guiding my technical development to machine and deep learning, for endless conversations about investing and trading, and most importantly being an amazing friend. Thank you to Alana Kent, for being a great skip in curling, for being an honorary member of the "Eastern Block", and for being a person I could always rely on in the time of struggle or procrastination. I also would like to thank the rest of our MIC research group for making my research experience productive and enjoyable. I thank Nadine Igonin, Eneanwan Ekpo, Atila da silva Paes, German Rodriguez, Hongliang Zhang, Thomas Eyre, Andrew Poulin, Megan Zecevic, Marko Venieri, Scott McKean, Paulina Wozniakowska and Lydia DiCaprio, for always being there for me. Thanks to my friends Stepan Lavrinenko and Stephan Prisacari, who supported and intellectually challenged me for the last 8 years. This thesis may have never come about if I never ran into them in my first year at University of Alberta and they convinced me to take a pause with the "rocket science stuff" and try geophysics instead.

I am further humbled by the countless friendships I made at UofC along the way. Specifically to Adriana Gordon, for the perpetual energy, great spirit and never ending coffee and food runs.

You bring joy to all who know you.

I am grateful for the many supportive graduate students, postdocs, and industry representatives I have met through SEGA/CSEG student group and thank them for sharing their wisdom, experience and encouragement over the years.

Finally, I thank my parents, my grandmother, and my extended family. You instilled in me a relentless drive to succeed and a "never give up" attitude. Thank you for the 27 years of support.

Table of Contents

Abstract	iii
Acknowledgements	iv
Table of Contents	vi
List of Tables	viii
List of Figures and Illustrations	ix
1 INTRODUCTION	1
1.1 Statement of Problem	2
1.1.1 World Energy Demand	2
1.1.2 Duvernay Formation	5
1.2 Brief Overview of Seismic Event Detection	8
1.2.1 Value of Complete Catalogs	11
1.2.2 Challenges of Event Detections	12
1.3 Seismic Event Detection Theory	15
1.3.1 Statistical signal processing	15
1.3.2 Waveform similarity	16
1.3.3 Machine Learning	18
1.4 Thesis Outline	23
2 OPTIMIZATION OF THE MATCHED FILTERING ALGORITHM FOR THE DETECTION OF INDUCED SEISMICITY	25
2.1 Introduction to the Problem	25
2.1.1 Theory of Matched-filtering	26
2.2 Methodology, Data and Tools	27
2.2.1 Data	28
2.2.2 Tools	31
2.2.3 ObsPy and EQCorrscan	32
2.2.4 Workflow	33
2.2.5 Parameter Optimization	34
2.3 Results	36
2.3.1 Length of the Template	37
2.3.2 Threshold Value	38
2.3.3 Local vs Regional Networks	40
2.4 Discussion	40
2.4.1 The Role of Parent Events and MAD Threshold Value	41
2.4.2 Implications on the Use of Regional Stations in the Presence of Local Arrays	41
2.5 Conclusions	42
3 TRAINING AND OPTIMIZATION OF CONVOLUTIONAL NEURAL NETWORK FOR DETECTION OF INDUCED SEISMICITY	44
3.1 Introduction	44
3.1.1 Basics of Neural Networks	44
3.1.2 Why Use Neural Networks?	45
3.1.3 Convolutional Neural Networks	46

3.2	Data, Methodology and Tools	47
3.2.1	Data	47
3.2.2	Class Imbalance	49
3.2.3	Convolution Layer	49
3.2.4	Max-pooling Layer	49
3.2.5	Activation Layer	50
3.2.6	Batch Normalization	51
3.2.7	Convolution Block	51
3.2.8	Loss Function	51
3.2.9	Tools	53
3.3	Investigating Different CNN Architectures For Event Detection	53
3.3.1	ConvNetQuake	54
3.3.2	Variation of Shallow VGG Network	58
3.4	Methodology and Workflow	59
3.4.1	Dataset Augmentation	59
3.4.2	Workflow	60
3.4.3	Methodology	61
3.5	Results	62
3.5.1	Detection Accuracy	62
3.6	Discussion	64
3.6.1	Detectability of Different Models	64
3.6.2	Avoiding Overfitting	64
3.7	Conclusions	65
4	COMPARISON BETWEEN MATCHED FILTERING AND CONVOLUTIONAL NEURAL NETWORK BASED DETECTION METHODS	66
4.1	Introduction	66
4.2	Data and Methodology	67
4.2.1	Synthetic data	67
4.2.2	Kaybob-Duvernay data	70
4.3	Results and Discussion	71
4.3.1	Synthetic data	71
4.3.2	Kaybob-Duvernay data	73
4.4	Conclusions	76
5	Conclusions and Future Work	77
5.1	Conclusions	77
5.2	Future work	79
A	UC/DSA array instrumentation	91
B	Distribution of the detection for each of the available seismic station networks with varying threshold values for the period of January 5-19, 2016.	95
C	Accuracy and Loss Plots for the ConvNetQuake and VGG-Junior Architectures with Cross-Entropy and Focal losses	98

List of Tables

2.1	Stations available for this study in the Crooked Lake area	31
3.1	ConvNetQuake architecture. The input is a waveform of 1000 points (100Hz, 10s) on 3 channels. Each layer consists of 32 filters that downsample the data by a factor of 2. After the eighth convolutional layer, the features are flattened into a 1D vector of 128 features.	55
3.2	VGG - junior architecture. The input is a waveform of 1000 points (100Hz, 10s) on 3 channels. After the 4 convolutional layers, the features are flattened into a 1D vector of 256 features	59
3.3	The classification results and implementation parameters given by proposed CNN methods.	62
3.4	The number of network parameters of two CNN architectures based on ConvNetQuake (Perol et al., 2018) and VGG-Junior.	63
4.1	Performance of the Convolutional Neural Network vs matched filtering method. Processing time here is referred to processing one month of continuous waveform data.	73

List of Figures and Illustrations

1.1	Projected growth in energy demand around the world: 2017-2040. MTOE stands for Million Tonnes of Oil Equivalent.	2
1.2	On the left: North American shale basins and major plays, along with the study areas referenced in this thesis shown by red circle. On the right: Stratigraphic column showing relative ages of shale plays.	4
1.3	Map of the Duvernay Formation organic-rich mudstones. Data from Switzer et al., 1994.	6
1.4	Late Devonian stratigraphic chart from Central Alberta. Modified from Switzer et al., 1994.	8
1.5	Seismic waveforms of a seismic event recorded with one borehole 3C monitoring station, with picked P- and S-wave arrivals. Modified from Rodriguez and Eaton, 2018.	9
1.6	Diagram illustrating the principle behind waveform similarity for earthquake detection. Two events with similar sources and similar raypaths will produce similar waveforms at the same seismometer.	17
2.1	Basic 1D Matched Filter	26
2.2	Visual representation of the cross-correlation process. Modified from http://eqcorrscan.readthedocs.io/en/develop/tutorials/matched-filter.html	27
2.3	The location of seismic stations and M_W 4.1 January 12, 2016 event. Purple boxes containing the University of Calgary (DSA) stations. The orange boxes contain RAVEN stations, yellow boxes contain industry installed stations. The red dot shows the M_W 4.1 January 12, 2016 event.	28
2.4	DSA waveforms from 6 stations for M_W 4.1 January 12, 2016 event. Z-orientation is red, N-orientation is green, E-orientation is blue.	30
2.5	WSK waveforms from 4 stations for M_W 4.1 January 12, 2016 event. Z-orientation is red, N-orientation is green, E-orientation is blue. These stations are located significantly closer to the epicenter than DSA stations (Figure 2.4)	30
2.6	Workflow for the application of the MFA algorithm	33
2.7	Number of detections (January 5-19, 2016) versus the length of the template used in the MFA applied to each seismic network independently.	37
2.8	Number of detections (January 5-19, 2016) versus the threshold value used in the MFA applied to each seismic network independently.	38
2.9	Distribution of the detections for each of the available seismic station networks with MAD threshold value of 6 and template length of 10s for the the period of January 5-19, 2016.	39
3.1	Location of seismic stations and events used for training of the convolutional neural network. UC/DSA stations (start with DS) are the University of Calgary DS network stations. Other stations were installed by the AER (RAVEN network).	48
3.2	Visualization of sigmoid and ReLU activation functions.	50
3.3	Comparison between standard cross-entropy loss (blue line) and focal loss as a function of probability of ground truth class. Modified from Lin et al., 2018.	52

3.4	ConvNetQuake architecture. The input is a waveform of 1000 points (100Hz, 10s) on 3 channels. Each layer consists of 32 filters that downsample the data by a factor of 2. After the eighth convolutional layer, the features are flattened into a 1D vector of 128 features. Modified from Perol et al. (2018).	54
3.5	VGG - junior architecture. The input is a waveform of 1000 points (100Hz, 10s) on 3 channels. After the 4 convolutional layers, the features are flattened into a 1D vector of 256 features	58
3.6	Workflow used in this study for the convolutional neural network method.	60
4.1	Day-long synthetic seismic record. The average SNR of this synthetic seismic record is 3 dB.	68
4.2	Templates used to generate synthetic data. Scaled copies of these two templates were inserted at random times over a Gaussian noise background. Figure on the left: template W_1 used to build both the synthetic training set and the first synthetic test set. Figure on the right: template W_2 used to create the second synthetic test set.	69
4.3	Detection accuracy between DuverNet (CNN based) and MFA for the similar events.	72
4.4	Detection accuracy between DuverNet (CNN based) and MFA for the dissimilar events.	72
4.5	Runtime of the two methods: DuverNet (CNN based) and MFA.	74
4.6	Memory use of the two methods: DuverNet (CNN based) and MFA.	75
4.7	Seismogram of an event that was detected by DuverNet and MFA detector.	75
4.8	Seismogram of an event that was detected by DuverNet, but was missed by MFA detector.	76
A.1	Self-noise graph for Trillium broadband seismometer. Modified from https://www.nanometrics.ca/sites/default/files/2019-03/trillium_compact_datasheet.pdf	91

Chapter 1

INTRODUCTION

Induced seismicity in the Fox Creek/Crooked Lake region of central Alberta has become a more pressing issue within the last five years because of heightened interest in the region by operators. This increase of seismic activity in the Fox Creek/Crooked Lake area has prompted the installation of the Regional Alberta Seismic Observatory of Earthquake Studies Network, University of Calgary DSA array, and multiple other local arrays installed by industry operators in the area. Increasing need for near real-time analysis especially for urgent traffic light regulatory monitoring (Kao et al., 2016).

Even though the coverage of the area by broadband stations has improved significantly, open access event catalogs remain sparse and have a limited number of detected events. Moreover, processing and interpretation of induced seismicity activity can take a long time. For example, the Incorporated Research Institutions for Seismology (IRIS) catalog (collection of first-run bulletins from the International Seismological Centre (ISC) and United States Geological Survey (NEIC) catalogs) contains only 1 event between January and March 2016 and it does not contain the crucial arrival information. The TransAlta catalog, maintained by Nanometrics, has 36 events (19 events with local magnitude above 2.0) for the same period of time. This is insufficient for informative interpretation, so we are motivated to seek methods to increase the catalog completeness and speed up the process of interpretation. Machine-learning methods are well suited for this application, due to the ability of these algorithms to scale and to quickly execute complicated classifiers.

This chapter provides an overview of this thesis and relevant scientific literature. It explains why additional work is required in the event-detection space and how recent developments in

machine-learning perform relative to more established methods for seismic event-detection. This chapter also reviews unconventional reservoirs, focusing on North America and Duvernay Formation, induced human-caused seismicity. Ways to detect induced seismicity are also discussed. A brief history of seismic event-detection and machine-learning techniques that help with the clustering and detection are introduced and discussed.

It is important to note that I am using machine learning definitions throughout this thesis, which might not always be the same as exploration geophysics or digital signal processing definitions.

1.1 Statement of Problem

1.1.1 World Energy Demand

According to a recent report by the International Energy Agency (IEA) (IEA, 2018), the global demand for energy will grow by 27% by 2040. This includes growth in both natural gas and oil demand, with oil and natural gas share in the mix continuing to meet 53 percent of the world's energy needs in 2040. Improvements in energy efficiency and the use of lower-carbon energy sources will help moderate energy demand in developed nations. However, population growth and improved standards of living will drive increased energy demand, especially on the continents of Asia and Africa (Figure 1.1).



Figure 1.1: Projected growth in energy demand around the world: 2017-2040. MTOE stands for Million Tonnes of Oil Equivalent.

Meeting this growing demand is a significant energy industry challenge, since many easily accessible conventional sources of energy are depleting (Smith and Montgomery, 2015). Unconventional resources are abundant (Figure 1.2), but require secondary recovery techniques. These techniques require more technological advancement. The shift from conventional to unconventional hydrocarbon resources production is happening in unison with the accelerating transition to the renewable energy sources. Increasing risks of global warming and growing opposition to the oil and gas development creates extra incentives for energy industry to produce lower-carbon fuels in a safer and more sustainable manner. The most sustainable hydrocarbon source of energy that is promoted as a replacement for burning wood, oil, and coal, is natural gas. It generates fewer CO₂ emissions and other post-combustion pollutants when burned for energy (Pacala and Socolow, 2004). Most of the onshore conventional gas reserves are depleted and are declining. Therefore, the majority of new gas production will be derived from unconventional oil and gas plays. By 2040 unconventional plays are expected to provide over 60% of produced natural gas (IEA, 2018). For example, Alberta's unconventional reserves are estimated to be 96 trillion cubic meters of natural gas, 420 billion barrels of oil, and 59 billion barrels of condensate (Rokosh et al., 2012), which would be enough to power the United States for the next century.

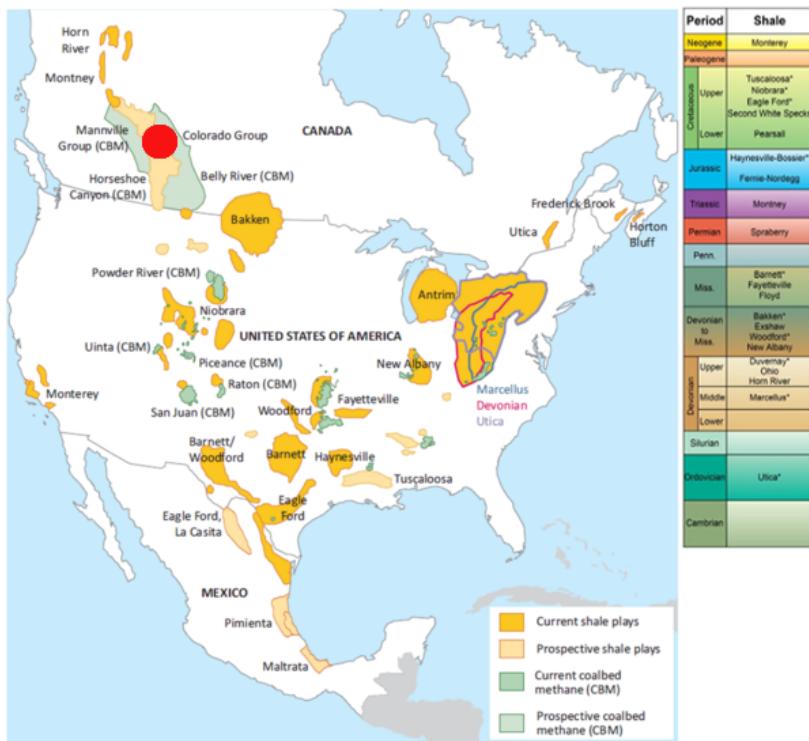


Figure 1.2: On the left: North American shale basins and major plays, along with the study areas referenced in this thesis shown by red circle. On the right: Stratigraphic column showing relative ages of shale plays.

Economic production from shales and other low permeability rocks requires multistage hydraulic fracturing (King, 2012) to be economic. Advancements in geomechanics tend to help with improved understanding and more efficient hydraulic fracturing approaches (Montgomery and Smith, 2010). However, poorly designed completions programs occur often, which results in lower recovery factors, lost stages, and increased emissions through inefficiencies and leaks (Howarth, 2011). Moreover, hydraulic fracturing and wastewater injections can cause induced seismicity (i.e. seismic events triggered by human activities in the subsurface). Examples of subsurface activities that can trigger seismic activities include mining, geothermal, and fluid injection or withdrawal (Eaton, 2018).

Induced seismicity due to wastewater injection or enhanced geothermal has caused 5.8-magnitude earthquake in Oklahoma (USA) (Chen and Nakata, 2017), and 5.5-magnitude earthquake in Pohang (South Korea) (Grigoli et al., 2017). Hydraulic-fracturing induced seismicity is also occurring in the Western Canadian Sedimentary Basin (WCSB), which threatens local producing companies' regulatory licenses or social licenses to operate (Atkinson et al., 2016; Bao and Eaton, 2016; Wang et al., 2016). Geoscientists, engineers, and researchers need to work on better understanding of the hydraulic fracturing and how it changes the subsurface conditions. Induced seismicity in the Fox Creek/Crooked Lake region of central Alberta has become a more pressing issue within the last five years because of heightened interest in the region by operators.

1.1.2 Duvernay Formation

Duvernay formation sediments were deposited during the Late Devonian in Western Canada (see Figures 1.3,1.4), roughly the same time as several other major black shale deposits across North America. During the Middle to Late Devonian, global sea level was higher than the present day (Savoy and Mountjoy, 1995; Haq and Schutter, 2008), resulting in the widespread flooding of continental margins. It is within these settings that many organic-rich Devonian successions were deposited, such as Duvernay, Horn River, Bakken, Marcellus, Chattanooga, and Woodford shales.

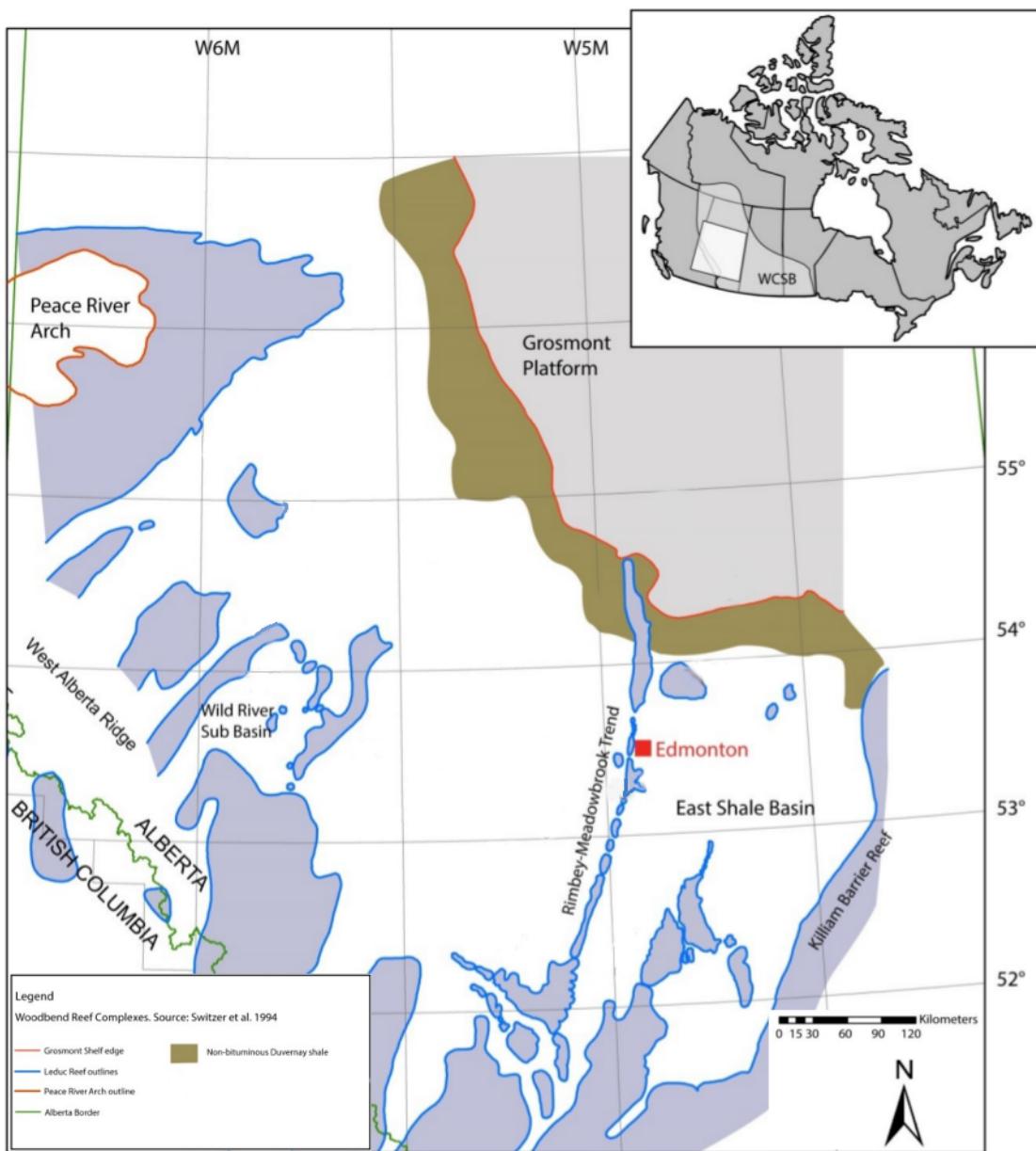


Figure 1.3: Map of the Duvernay Formation organic-rich mudstones. Data from Switzer et al., 1994.

The WCSB was a passive margin at the western edge of North America during the Late Devonian. The northwest margin of North America was mainly dominated by open marine shales in British Columbia and the Northwest Territories, transitioning to shallow-water carbonates in Alberta (Ziegler, 1967). During deposition of the Woodbend Group (see Figure 1.4), the rate

of accumulation and preservation of sediment increased dramatically. In Alberta, the Woodbend Group section is characterized by thick accumulations of basin-filling shales, including hydrocarbon source rock, and extensive reef complexes (Switzer et al., 1994).

Tectonic features that influenced sedimentation during deposition of Duvernay Formation include the Peace River Arch, West Alberta Ridge, Rimbey Arc, and Meadow Lake Escarpment (Figure 1.3). The Peace River Arch was an emergent landmass on the west side of the basin, fringed by Leduc Formation reefs (Dix, 1990, Eaton et al., 1999). The west Alberta ridge was flooded, but formed a base for extensive Leduc reef complexes (Switzer et al., 1994). During the deposition of Duvernay Formation sediments, the lineament was marked by a chain of Leduc Formation reefs called the Rimbey-Meadowbrook trend. This reef chain divides the basin into a West Shale and East Shale Basins (Eaton at al., 1995). There was more accommodation space available to the West Shale Basin, in comparison to East Shale Basin, due to differences in tectonic subsidence on either side of Rimbey Arc. Differential compaction amplified this effect, because the West Shale Basin was a major site of shale deposition, while the East Shale Basin was underlain by the Cooking Lake Carbonate Platform (Switzer et al., 1994). The Meadow Lake Escarpment is a pre-Devonian erosional feature. During the deposition of Duvernay Formation, the Killiam Barrier Reef roughly coincided with the underlying Meadow Lake Ecarpment, and established the furthest eastward extent of Duvernay Formation shales (Oldale and Munday, 1994).

The Duvernay Formation overlies the Majeau Lake Formation throughout most of the basin (Figure 1.4). In the south and west, Majeau Lake Formation was not deposited and Duvernay strata onlaps Swan Hills Formation platform carbonates and older Devonian strata. The Ireton Formation overlies the Duvernay Formation across much of the basin. Near reef complexes and the Grosmont Platform, Leduc Formation reef-margin strata and Grosmont Formation platform carbonates, respectively, overlie the Duvernay Formation. The Duvernay Formation has been

subdivided into lower, middle, and upper members (Andrichuk, 1961).

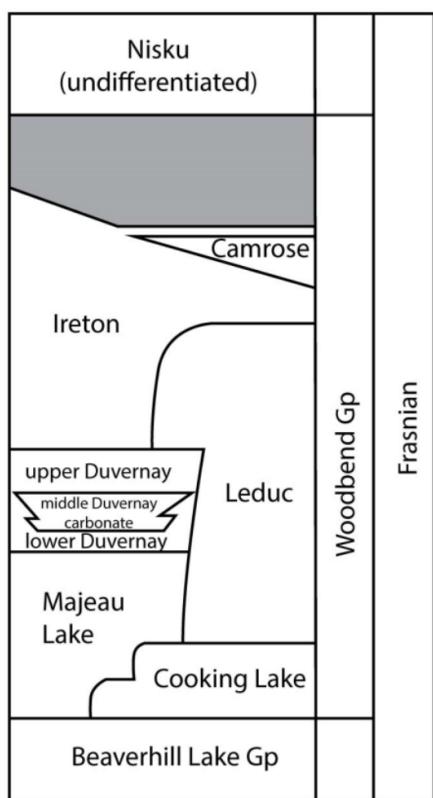


Figure 1.4: Late Devonian stratigraphic chart from Central Alberta. Modified from Switzer et al., 1994.

The Duvernay Formation is predominantly composed of organic-rich, siliceous-calcareous mudstones. West Shale Basin sediments are notably less calcareous than East Shale Basin deposits (Rokosh et al., 2012). Limestones and dolostones are observed proximal to reef complexes. Finally, organic-rich siliceous-calcareous mudstones are common within the upper and lower Duvernay, while less organic-rich limestones dominate the middle Duvernay (Andrichuk, 1961).

1.2 Brief Overview of Seismic Event Detection

Earthquake seismology is an observational science. It uses data collected from seismic sensors to study processes within the earth. Automated earthquake detection is a fundamental step of the earthquake seismology and is based on the use of algorithms to identify seismic signals in continuous ground motion measurements. Acc-

curate and complete earthquake catalogs are critical for studying seismic hazards and earthquake processes. Currently, there are tens of thousands of the seismic sensors recording data continuously that are installed worldwide. As the volume of recorded data grows, there is increased need to adopt state-of-the-art algorithms and data-intensive computing techniques to process increasingly large seismic datasets.

The data used for the detection of seismic events are continuously recorded ground motion. Sensors that measure the motion of the ground over a large frequency range are called broadband seismometers. Sensors can record only the vertical component or three-component (vertical, north-south, east-west) data, and typically sample the continuous signal at between 20 and 200 Hz. Seismologists refer to the full record of all waveform data as continuous waveform data. An example of the seismogram that contains event with P-wave arrival, S-wave arrival and coda can be seen on figure 1.5. I refer to surface waves and coda (multiple reflections scatter) as S-wave coda or the signal that comes after initial S-wave arrival.

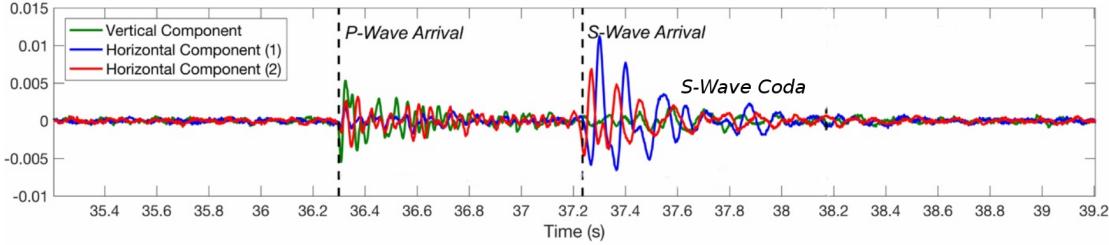


Figure 1.5: Seismic waveforms of a seismic event recorded with one borehole 3C monitoring station, with picked P- and S-wave arrivals. Modified from Rodriguez and Eaton, 2018.

A standard earthquake monitoring workflow typically involves multiple processing steps including automated detection, location and characterization of earthquakes. Automated earthquake detection is usually applied separately to the continuous data from each station, and is often divided into an event detection (or triggering) step, which identifies approximate time intervals that correspond to seismic events, and a phase-picking step, which determines the precise onset times of the P and(or) S waves (Akram and Eaton, 2014). Firstly detecting and then validating weak earthquake signals from single station data can be difficult, and the accuracy of the locations and derivation of earthquake sources can be improved with information from multiple stations. Therefore, earthquake detection algorithms, when available, use a combination of data from multiple stations in a seismic network. The process of picking arrivals from different stations is referred to as association.

There are two broad classes of event detection methods: 1) batch processing algorithms for small-to-large data volumes of previously collected data, and 2) near real-time algorithms for streaming data. In near real-time systems, events are detected as they occur, with the detector deciding seconds after the data become available. Public earthquake catalogs are typically produced from real-time algorithms, but it is also common for seismologists to reprocess archived data in areas of interest to identify lower magnitude events to produce more complete catalogs. In this thesis, I concentrate on the algorithm for batch processing in long-duration data sets toward the goal of detecting events that are missed by standard detectors.

There is a wide variety of automated detection algorithms available to be applied to seismic data sets (Sharma et al., 2010). Each method provides its own measure of waveform characterization or similarity to a set of reference waveforms. Some methods produce a set of arrival times, or phase picks, based on the triggering of a characteristic function. A widely known and used characteristic function, short-term/long-term averaging (Allen, 1978), is entirely dependent on the signal-to-noise ratio and is forced to set trigger thresholds that fail to flag true arrivals in a low signal-to-noise ratio signal. Advanced characteristic functions, which are based on statistical measures such as kurtosis or skewness (Saragiotis et al., 2002; Baillard et al. 2014), are capable of producing phase picks at close to unity signal-to-noise ratio (Galiana-Merino et al. 2008). However, these measures fail to utilize the relative amplitudes of real seismic events at different frequency bands. Relative frequency content is accounted for in autocorrelation (Brown et al. 2008; Bostock et al. 2012) and cross-correlation (Gibbons and Ringdal 2006; Shelly et al. 2007) methods, as they are based on waveform similarity to a set of pre-detected and located event templates. These correlation methods will report picks only if the event's waveform has a similar shape to the template in the existing database, allowing unique waveforms to go undetected and limiting the utility of these methods. This constraint is reduced with the application of a more

recent algorithm, called FAST (Yoon et al. 2015). It has shown the capability to match generalized features between waveforms as opposed to the specific waveform shape. Other improvements have been made to negate the effect of noise in the desired frequency range (e.g. Madureira and Ruano 2009; Lomax et al., 2012; Poiata et al., 2016), by essentially splitting the waveforms into a number of frequency bands and reducing the likelihood that noise overlaps with the signal.

Machine-learning techniques have been previously applied for event detection (e.g. Beyreuther et al. 2012; Riggelsen & Ohrnberger 2012; Perol et al., 2018), phase picking (e.g. Gentili & Micheletti 2006), event classification (e.g. Vallejos & McKinnon 2013; Mousavi et al. 2016), making use of a wide range and combinations of waveform attributes. However, none of these studies utilize these methods on a network scale of event detection. Sets of picks are grouped based on their goodness-of-fit to a reference velocity model. These sets lack a unified sense of the probability of event occurrence, as various statistics (such as the number of observations, pick quality and residuals) provide their measure of the event's quality. The lack of a single value representing an event's quality has been overcome by methods that make use of observations by all stations simultaneously (e.g. Baker et al., 2005; Poiata et al., 2016). These network-based detections provide an estimate on the location of the event, as well as a single statistic relating to event likelihood.

1.2.1 Value of Complete Catalogs

Earthquake catalogs, produced by global or regional seismic networks, can typically detect moderate to large magnitude events ($>M1.8$), but are known to be incomplete at lower magnitude (Visser et al., 2017). Even though these smaller magnitude events are too weak to be felt on the surface or to cause infrastructure damage or loss of life, they can nevertheless provide valuable insights into earthquake processes and potential hazards (Eaton, 2018). Combination of advanced algorithms with use of more sensitive detectors can increase the number of events detected by one to two orders of magnitude (Schaff and Waldhauser, 2010; Huang and Beroza, 2015). Having more complete catalogs allows seismologists to have a more complete picture of the seismic

activity in a particular area of interest, which allows scientists to discover faults and patterns not seen previously using more limited catalogs. For example, additional events detected using waveform similarity has enabled to demonstrate a link to the relationship between slow earthquakes and tectonic tremor (Shelly et al., 2007).

1.2.2 Challenges of Event Detections

Automatic earthquake detection is an essential task in seismology, but it remains a challenging problem with a room for improvement and innovation. Associated challenges and requirements for the automatic detection algorithm include the following:

- **Lack of the absolute(ground) truth.** Absolute (ground) truth is an objective and necessary standard for evaluating the performance of an algorithm. In our case, ground truth would be represented by the complete list of every earthquake in the waveform data. In practice, the closest researchers can get to the ground truth information is simply to treat a catalog as a reference dataset. However, earthquake catalogs are known to be incomplete. The completeness and quality of existing catalogs vary significantly, both temporally and spatially (Schorlemmer and Woessner, 2018). There are multiple reasons for this. Firstly, past records may be incomplete in some regions where sensor coverage is limited. Secondly, past records may not be representative when there are changes in patterns of seismic activity, such as changes in regions experiencing increasing rates of induced seismicity (Ellsworth, 2013). Even if a region is well covered with seismic stations and has data recording stretching back over a decade, catalogs are missing many events with the lower magnitudes due to the detection sensitivity limits of the equipment and the algorithms used to generate the catalogs.

Incomplete and skewed catalogs are a challenge for developing earthquake detection algorithms. The lack of absolute (ground) truth data leads to difficulties with the validation of the performance of new algorithms. The goal of developing new earthquake detectors is to detect more events than existing methods, however validating the performance of the new method is challenging because the most complete catalogs (used for benchmarking) are typically produced with existing methods. Such incomplete or limited earthquake catalogs result in a lack of absolute (ground) truth data, meaning that there is a biased or limited training data available for training in supervised machine learning/deep learning algorithms (see Section 1.3.3).

- **Complex event sequences.** Earthquakes that occur during relatively active periods of seismic activity are of particular interest to researchers. In an ideal case, a detector should be able to identify events that occur during complex event sequences, such as slow-slip events, volcano-tectonic swarms, aftershock sequences, and fore-shock sequences. Unfortunately, detecting events in these complex event sequences is challenging, because there may be events that occur close together in time.
- **Local and other major sources of noise.** Broadband seismometers record the motion of the ground, however not all ground motion is due to seismic sources. In addition to energy coming from earthquakes, seismometers record local noises, including both human activity related noise such as vehicles and transportation systems or crowd noises (Vidale, 2011; Diaz et al., 2017), and natural sources such as wind, tidal waves and storms (Bromirski, 2001). Presence of these noise sources can produce false detections or false picks. Moreover, energy from other significant events such as landslides (Weichert et al., 1994) and quarry blasts (Wiemer and Baer, 2000) can also interfere with detection of the desired events as well as lead to false detections on both a regional or global seismic networks. Ideally, an earthquake detector should be able to detect seismic events without incurring excessive

false detections due to other sources.

- **Imbalanced data sets.** There is a considerable noise in continuous seismological data sets. Earthquake occurrence is relatively infrequent and represents a small percentage of the signals in the continuous records. Therefore, a useful earthquake detection algorithm should be able to detect the maximum number of earthquakes (high detection rate) while also keeping a false-detection rate to the minimum. In machine-learning terms, a practical earthquake detection algorithm should have high precision and high recall, where precision is the percentage of true earthquake among all of the detections, while recall is the percentage of earthquakes that are detected by the algorithm.

To illustrate these terms, let us consider the following example. A hundred earthquakes occurred in a one-week time period (14.3 earthquakes per day). If a particular detection algorithm correctly detects 95 of these events, then its recall is 95%. If on top of that detector returns an average of two false detections per day, then the algorithm will identify 114 detections for a precision of $95/(95 + 14) = 87\%$, which means that 87% of the detections identified by the algorithm are real earthquakes. The work in this thesis is concerned with detecting weak signals from lower magnitude events, achieving high recall while minimizing the false detection rate to achieve a reasonable precision.

- **Poor quality or missing data.** Earthquake detectors are meant to run on real sensor-recorded data. Recording sensors are subject to failure, malfunctions and faults. Therefore, continuous data records may include periods that lack events or overly noisy data due to sensor shutdown, transmission or maintenance failures. Earthquake detectors should thus be robust enough to tolerate poor quality or missing data at one or more stations in the seismic networks.

- **Large data volumes.** The amount of available seismic waveform data is growing rapidly. There are two ways in which one can characterize size of large datasets: large- N problems and large- T problems. Large- N problems refer to a situation where there are larger seismic networks with hundreds to thousands of sensors. Large- T problems refers to the situation where there are long-duration data sets that contain continuous records stretching back over ten years (Li et al., 2018). In this thesis, I focus on local regional stations data which is a large- T type of the problem. The availability of large sets in earthquake seismology presents both an opportunity and a challenge.

1.3 Seismic Event Detection Theory

Seismic waveform analysis in the form of earthquake detection and phase-picking has been extensively researched for decades. In this section, I review the relevant literature on earthquake detection.

1.3.1 Statistical signal processing

Energy-based methods, such as short-term average/long-term average (STA/LTA) algorithm (Allen, 1982; Akram and Eaton, 2016), are efficient earthquake detectors used to identify the onset of seismic waves based on the increase in signal energy associated with the wave arrivals. Energy-based detectors take into account only basic information between the strength of the signal and background noise, without using waveform signatures or characteristics. However, the simplicity of this detector makes it very computationally efficient. Therefore, STA/LTA related methods are the basis and the first step for many real-time monitoring systems (Lomax et al., 2012). Energy-based detectors are good at detecting events with strong impulsive arrivals, but they tend to miss events that are weaker, have emerging arrivals, or events that are overlapping or closely spaced in time. They also may produce a large number of false picks.

It is common for event detectors, including energy-based detectors, to identify the seismic wave arrivals by applying a certain threshold to a characteristic function. Another class of detection algorithms uses kurtosis or other higher-order-statistics (Saragiotis et al., 2002; Kuperkoch et al., 2010) of the signal in the characteristic function. Some other techniques proposed for event detection include wavelet decomposition (Anant and Dowla, 1997) and polarization analysis (Jurkevics, 1988).

1.3.2 Waveform similarity

The difficulty of detecting non-impulsive arrivals and low signal-to-noise events encouraged the development of sensitive detectors based on waveforms similarity. Waveform similarity detection methods are powerful techniques for identifying events that are not detected by the standard energy-based detectors.

Waveform cross-correlation, also referred to as matched-filtering or template matching, is a highly sensitive detector based on waveform similarity that has been successfully applied to a range of detection problems (Gibbons and Ringdal, 2006; Shelly et al., 2007; Skoumal et al., 2014). Waveform cross-correlation as an earthquake detection method is based on the assumption that earthquake signature can be identified by their similarity to the waveforms of previously detected events (see Figure 1.6). The similarity of the waveforms is measured by the normalized cross-correlation. If two events are highly similar, it indicates that they share similar source mechanisms and locations (Poupinet et al., 1984).

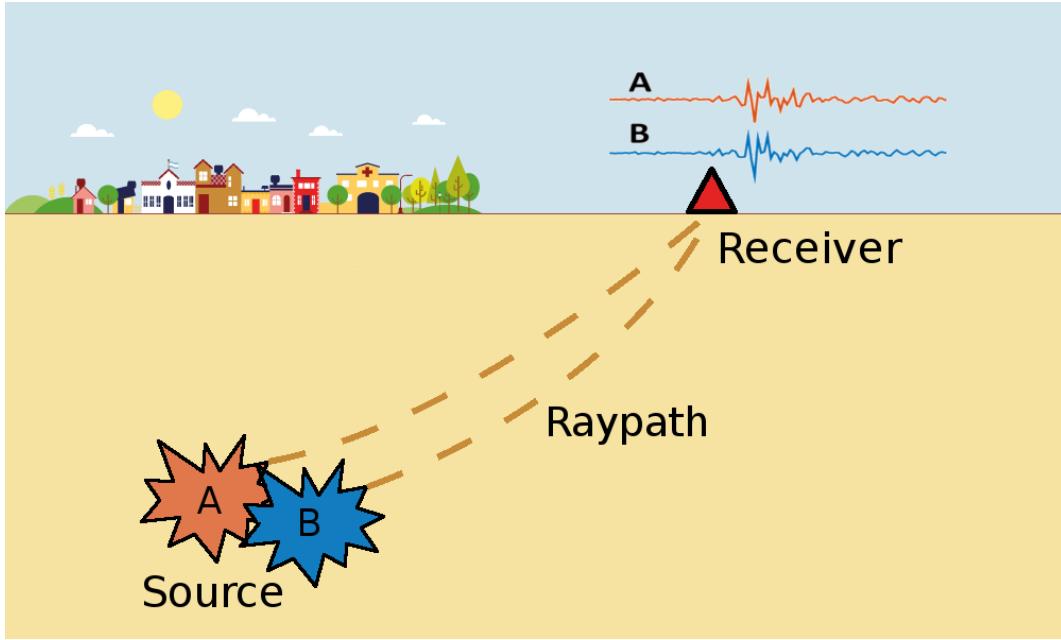


Figure 1.6: Diagram illustrating the principle behind waveform similarity for earthquake detection. Two events with similar sources and similar raypaths will produce similar waveforms at the same seismometer.

Template matching requires a library of template waveforms, which were previously detected by other methods. This makes it highly sensitive, at the cost of limitation to detecting events only in the areas with an existing seismicity catalogs and known sources. Subspace detectors, which belong to the related class of generalized correlation-detectors, can identify non-repeating sources (Harris, 2006; Eaton, 2018). However, they still require prior knowledge of waveform signatures. The match-and-locate method (Zhang and Wen, 2015) is an example of more general correlation-detector that allows for larger variability in source locations, while still requiring template waveforms.

Not all waveform-similarity-based detectors require prior knowledge of waveform signatures. The autocorrelation detector proposed by Brown et al. (2008) is a variation of waveform cross-correlation algorithms that identifies potential earthquake signals using a brute-force search for similar waveforms in continuous data. Even though autocorrelation is capable of detecting events

with previously unknown waveforms, it has poor scalability that limits its use to short-duration data sets (Aguiar et al., 2017).

1.3.3 Machine Learning

Machine learning has been used for earthquake detection on a limited basis for decades. Two recent technological development created exciting new opportunities for data-driven methods in seismology. First is the increase in computational capabilities that allows for processing of the larger data sets faster. Second is the increasing availability of massive seismic data sets to process. As a result, in the last few years there has been growing interest in the research community to introduce modern machine-learning techniques into earthquake detection workflows.

Supervised learning

Supervised machine learning algorithms create a model through mapping a data vector to a corresponding label or target value, using **training data** (data for which both the input and corresponding label/target are available to the algorithm) to optimize the model. A properly designed and well trained model should be able to generalize to make accurate predictions of the labels for the previously unseen input data.

Supervised machine-learning uses examples of known earthquake waveforms, as training data, to learn a general model for detecting earthquake waveforms in the continuous recordings. Specifically, the supervised learning algorithm seeks a function that maps each waveform, or set of features extracted from the waveform, into a label indicating whether that represents an event or noise. The advantage of this approach over traditional signal processing techniques, like template matching, is the ability to generalize to new type of events that were not present in the training set. Therefore earthquake detection by supervised learning algorithms has the potential to identify earthquakes from previously unknown sources.

Template matching is more accurately categorized as a signal processing technique rather than a supervised learning. However, template matching can be thought of as a **supervised** signal processing method, in the sense that it relies on a set of labeled examples to classify waveforms as earthquakes or noise. In contrast with typical supervised learning algorithms, template matching only uses examples from one class (earthquakes) in its detection model, and it has limited ability to generalize to new events. Template matching can only identify events that have waveforms with high similarity to the template waveforms.

There have been a number of research efforts aimed at using supervised machine learning techniques for earthquake detection, phase-picking and signal classification tasks in earthquake seismology. Pattern recognition system developed by Joswig (1990), called Sonogram-detector, was capable of detecting earthquakes by correlating spectrogram features from the continuous waveform data with those in a training set, called 'knowledge base', of known earthquakes and noise signals.

A number of studies have used artificial neural networks (ANNs) for event detection and phase-picking (Zhao and Takano, 1999; Gentili and Michelini, 2006). ANNs have also been used for signal classification to differentiate earthquakes from other types of events, including quarry blasts (Scarpetta et al., 2005), landslides (Esposito et al., 2006), underground nuclear tests (Dowla et al., 1990), underwater explosion (Del Pezzo et al., 2003), and chemical explosions (Dysart and Pulli, 1990). The majority of studies concentrated on applying ANNs to phase-picking or signal classification tasks provide results using pre-triggered data rather than continuous data. The aforementioned studies use "shallow" neural networks with single hidden layer and relatively few parameters. Therefore, they are generally trained using hand-engineered features calculated in the time and/or spectral domains based on previous domain expertise using small training sets of tens to hundreds of labeled waveforms.

Another approach to earthquake detection problem is to use Hidden Markov Models (HMMs). Collectively, HMMs are an approach for modeling sequential data that have originally been used in speech recognition (Rabiner, 1989). These techniques have been applied to continuous seismic data for the detection and classification of volcanic signals (Beyreuther et al., 2008), regional earthquakes (Beyreuther and Wassermann, 2008) and induced earthquakes (Beyreuther et al., 2012). Dynamic Bayesian networks (DBNs), another type of graphical model that generalizes HMMs, have also been used for earthquake detection (Riggelsen et al., 2007). Three-component DBN-based algorithm was proposed for real-time detection in continuous data, but lack of absolute (ground) truth has been cited as a challenge for evaluating performance (Riggelsen and Ohrnberger, 2014).

The majority of applications of supervised machine-learning to earthquake detection, including research presented in this thesis, has been designed to operate on data from a single channel or station. Two recent studies introduced network approach to earthquake detection. Reynen and Audet (2017) proposed an algorithm for event detection and classification that utilizes a logistic regression classifier, combined with a custom association algorithm that accounts for class probabilities from each station in the seismic network. In 2018, Draelos et al. introduced a reinforcement learning algorithm for dynamic selection of thresholds for single-station detectors based on the observations at neighboring stations. This approach is a general method for automated parameter tuning that can be used to improve the sensitivity of detection on the single station by using information from the seismic network.

In recent years there was increased adoption of sophisticated neural network architectures for earthquake detection and phase-picking. The first application of recurrent neural network (type of neural network used for the analysis of sequential data) to earthquake detection was published

by Wiszniowski et al. in 2014. Several more recent studies have applied deep neural networks (LeCun et al., 2015) to the problems of event-detection (Wu et al., 2017; Perol et al., 2018) and phase-picking (Zhu and Beroza, 2018; Ross et al., 2018). Deep neural networks typically have more than one hidden layer and sometimes are also referred to as deep learning methods. Deep neural networks generally require a very large number of labeled examples for training, which in earthquakes seismology can be a challenge. In general, this algorithm require a vast amounts of data (thousands to hundred of thousands of labeled template waveforms), which in turn allows them to perform feature extraction and prediction automatically. This means that deep learning algorithms can be applied to data with minimal preprocessing, rather than relying on the use of hand-engineered features, as in earlier work using artificial neural networks, but requires larger labeled data sets.

One of the major limiting factors to the performance of these methods is the quality and quantity of the data set used. Machine-learning models tend to adopt any biases present in the training data, so mislabeled or incomplete training data (e.g. events incorrectly labeled as noise or weak earthquakes left out of training data set) can hurt performance of the model. Therefore, supervised learning methods have a similar weakness as template matching: the performance is limited by the quality and diversity of the template event waveforms used for the training. Moreover, continuous waveform data sets in which earthquake signals represent a small percentage of the data (imbalanced data set), also likely to present a challenge to learning algorithms (Krawczyk, 2016), often resulting in weak performance with respect to the underrepresented class.

Unsupervised learning

Methods that learn patterns or structures in data sets without relying on labeled information are referred to as unsupervised learning techniques. Unsupervised learning areas include dimensionality reduction, clustering, and feature learning. This approach is often required in cases in

which few labeled data are available for given data set.

For analysis of the seismic signal, the majority of applications of unsupervised learning techniques have been used for exploratory analysis. Examples of such applications include discrimination between different types of signals, analyzing the relationship between event waveforms, or selecting a feature representation. Valentine and Trampert (2012) used an autoencoder network (type of unsupervised neural network designed to learn efficient encodings for data) to learn a compact feature representation for earthquake waveforms. Another type of unsupervised machine learning method that can be used for the dimensionality reduction is self-organizing maps (Kohonen, 2013). Self-organizing maps have been applied to clustering of multiple event types (Esposito et al., 2006), clustering events based on waveform similarity (Esposito et al., 2008), and seismic waveform data for feature selection (Kohler et al., 2008). Masiello et al. (2006) compared several unsupervised learning methods, including principal component analysis (PCA) and self-organizing maps, for differentiating signals generated by the different sources, which included quarry blasts and underwater explosions. Hierarchical clustering is another commonly used algorithm in data exploration. It has been applied to identify families of correlated event waveforms in induced earthquake swarms (Bardainne et al., 2006) or in aftershock sequences (Slinkard et al., 2013).

Several studies attempted to use unsupervised learning techniques for earthquake detection. Kohler et al. (2010) detected rockfalls and volcanic tectonic signals in continuous waveform data using a self-organizing map for clustering and classifying each cluster based on a small number of known examples. Skoumal et al. (2016) used agglomerative clustering to detect signals from repeating earthquakes. Finally, Harris and Dodge (2011) used clustering to dynamically update subspace detectors in real time.

Frequent pattern mining, motif discovery, and anomaly detection are some of the data mining methods for the unsupervised pattern discovery. They do not rely on training data and are designed to identify unknown pattern in large data sets. Aguilar and Beroza (2014) used PageRank (link analysis algorithm) for identifying possible low frequency earthquakes. The FAST earthquake detector (Yoon et al., 2015) poses earthquake detection as a data-mining problem and leverages locality-sensitive hashing (Andoni and Indyk, 2006) for identifying all pairs of similar waveform segments in single-channel continuous ground motion data.

1.4 Thesis Outline

In this work, I address the task of detection of the hydraulic fracturing induced seismic events in the Kaybob-Duvernay production region of Alberta. Specifically, I apply matched-filtering analysis to a data set of continuous waveform recordings from the regional seismic monitoring network to establish a baseline detection performance. After establishing the baseline, a convolutional neural network is designed, optimized, trained, and deployed with the goal of outperforming the detection metrics achieved by the matched-filtering method. This work is organized as follows:

- Chapter 2 discusses the available dataset, and application and optimization of the matched-filtering method to the available data. MFA is widely used algorithm for the detection of earthquakes in the areas where some event catalogs already exist. I describe how MFA was optimized before being applied to Kaybob-Duvernay data.
- Chapter 3 discusses detection of seismic events using the convolutional neural network (CNN). I discuss how to prepare the data to be used for the detection with the CNN. Moreover, I discuss how CNN-based detector is designed, optimizes, trained, and deployed. Different architectures and loss functions are discussed and tested. A novel Focal Loss function is introduced and used to help target class imbalance problem. Finally, I identify the most effective CNN to be used in chapter 4.

- Chapter 4 further describes the difference in the performance of two detection methods. I perform testing, using both synthetic data and Kaybob-Duvernay real dataset to compare performance of MFA and CNN based methods.
- Finally, Chapter 5 summarizes the results and conclusions of my research and describes recommendations for future research.

Chapter 2

OPTIMIZATION OF THE MATCHED FILTERING ALGORITHM FOR THE DETECTION OF INDUCED SEISMICITY

This chapter illustrates the development and optimization of a Python-based workflow for detection of induced seismic events from 3-component continuous recordings using matched-filtering analysis. The workflow applies a matched-filtering algorithm to the continuous data from nearby stations, utilizing events from independent catalogs as parent events.

2.1 Introduction to the Problem

Induced seismicity in the Fox Creek/Crooked Lake region of central Alberta has become a more pressing issue within the last five years because of heightened interest in the region by operators. This increase of seismic activity in the Fox Creek/Crooked Lake area has prompted the installation of the Regional Alberta Seismic Observatory of Earthquake Studies Network (RAVEN) (Schultz and Stern, 2015), University of Calgary DSA array, and multiple other local arrays installed by industry operators in the area (figure 2.3).

The objective of this chapter is to present a robust Python-based workflow for applying a matched-filtering algorithm (MFA) to continuous data from local and regional seismic arrays (Caffagni et al., 2016). MFA helps to identify small-magnitude earthquakes from continuous recordings of local seismic arrays. The goal is to generate a more extensive catalog of the regional seismicity as a way to achieve more accurate and faster magnitude calculation and event location in the future. For the purpose of testing and optimizing of this workflow, I chose a period between January 5 and January 19, 2016, because it contains sequence of seismic events associated

with a M_W 4.1 event that occurred on January 12, 2016.

Once the MFA algorithm is optimized I use it to establish a baseline detection performance with the goal of surpassing it by convolutional neural network in Chapters 3 and 4.

2.1.1 Theory of Matched-filtering

In signal processing, a matched-filter is obtained by correlating an unknown signal, with a known signal, "parent" or template, to detect the presence of the template in the unknown signal (Caffagni et al., 2016). Events detected in the unknown signal are typically referred to as "child" events. Mathematically it is equivalent to convolving the unknown signal with a conjugate time reversed version of the template, which is called cross-correlation. The matched-filter is the optimal linear filter for maximizing the signal-to-noise ratio (SNR) in the presence of white (stochastic) noise (Turin, 1960).

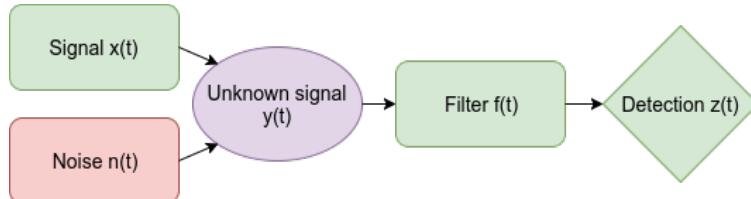


Figure 2.1: Basic 1D Matched Filter

To illustrate the 1D model for matched-filtering please consider the model illustrated in Figure 2.1 where the unknown input signal is $y(t)$, comprised of signal of interest, $x(t)$, and stochastic noise $n(t)$. The role of a matched-filter $f(t)$ is to maximize the SNR of the output signal, $z(t)$.

Matched-filters are commonly used in radar, sonar, digital communications, and they play a central role in gravitational-wave astronomy (Schultz, 1999). In geophysics, match-filtering is commonly applied for adaptive multiple subtraction (Fomel, 2008). For our purposes, matched-filtering requires a collection of well located templates "parent" events, selected on the basis of a high SNR and representative spatial distribution of the recorded seismicity. A visual representation of the matched-filtering application for detection of new events is shown in Figure 2.2. Detection

robustness is calculated as the ratio between the cross-correlation sum (*cccsu*m) and the detection threshold.

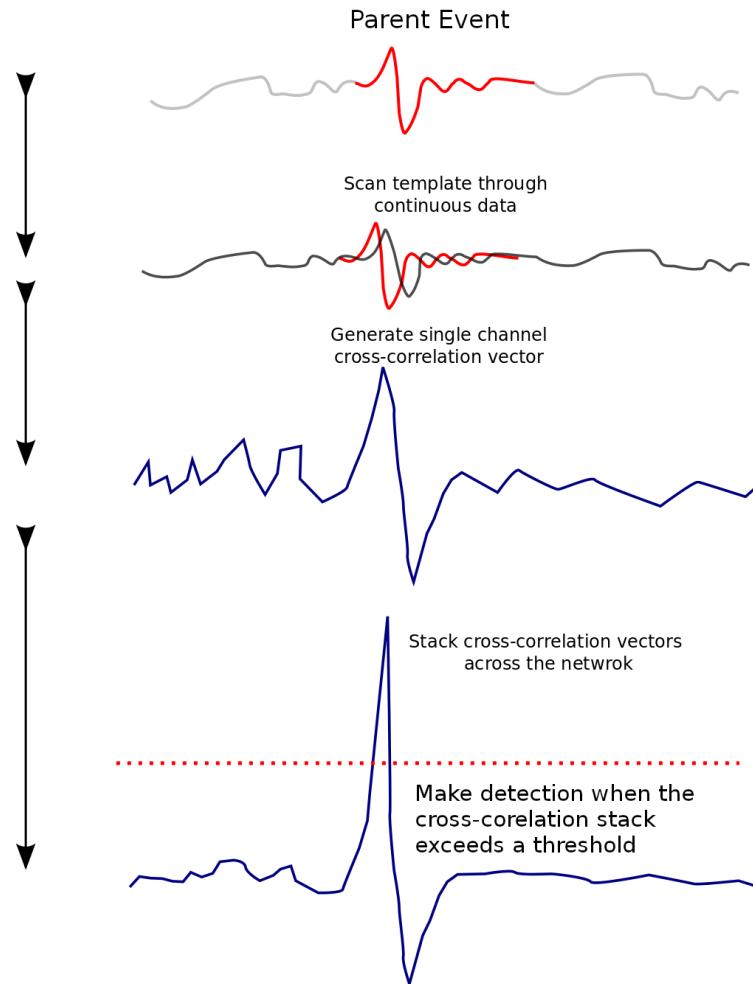


Figure 2.2: Visual representation of the cross-correlation process. Modified from <http://eqcorrscan.readthedocs.io/en/develop/tutorials/matched-filter.html>

2.2 Methodology, Data and Tools

In this section I list and discuss all of the available data and tools used for the workflow development and testing. In addition, I discuss the methodology of optimizing the parameters for the more efficient "child" events detection using MFA.

2.2.1 Data



Figure 2.3: The location of seismic stations and M4.1 January 12, 2016 event. Purple boxes containing the University of Calgary (DSA) stations. The orange boxes contain RAVEN stations, yellow boxes contain industry installed stations. The red dot shows the M_w 4.1 January 12, 2016 event.

The hypocentre of the major January 12, 2016 event is located approximately 30 km west of the Town of Fox Creek, Alberta. Our dataset contains recordings of three-component seismic data from regional and local networks. By regional arrays we refer to the networks that were installed for monitoring longer term larger scale seismic activity. By local arrays we refer to the networks that were installed for monitoring particular area during performance of the particular human activity. The full list of networks used in this study can be found below:

- **Regional Alberta Observatory for Earthquake Studies Network (RAVEN).** The deployment of RAVEN network began in support of the Earthquake Studies Project at the Alberta Geological Survey (AGS) with the initial station, LGPLA, deployed

in October of 2013 (AER, 2013). By 2018, the network had grown to 13 real-time, continuously telemetered, 3-component, broadband stations. RAVEN stations record ground motion at a sample rate of 100Hz. The ATSN network deployed by University of Calgary is incorporated into a RAVEN network.

- **University of Calgary UC/DSA array.** This array is composed of six broadband stations installed by Nanometrics, which record ground motion at a sample rate of 200Hz (figure 2.4). The network was installed in early January 2016 and operated continuously for 12 months. Daily continuous recordings are archived at the University of Calgary and will be released through IRIS in 2020. Detailed information about instrumentation can be found in Appendix A.
- **Local XTO and WSK arrays.** Local arrays installed by XTO Energy and Repsol. XTO array has 5 stations and WSK array has 4 stations, recording ground motion at a sample rate of 100Hz (figure 2.5). These datasets have been licensed to the University of Calgary for research.

All of the stations used in this study were continuously recording the data during the period surrounding the January 12, 2016 M_W 4.1 event (Figure 2.4). Selected stations are located between 1.1 and 167 km away from the January 12 event.

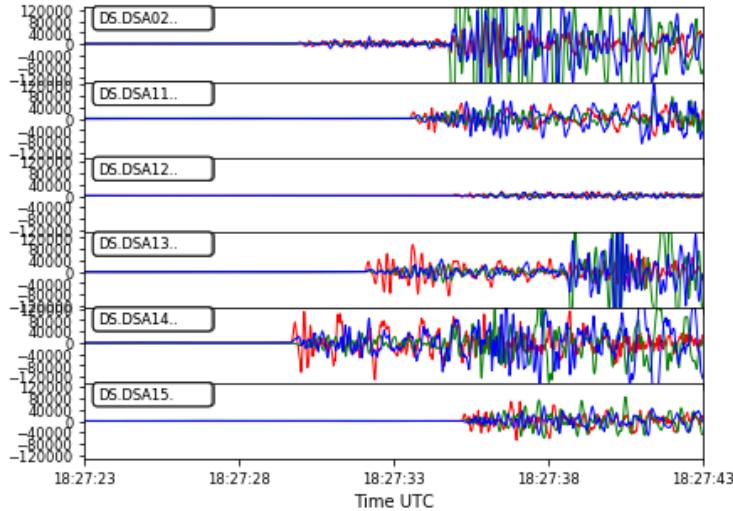


Figure 2.4: DSA waveforms from 6 stations for M_W 4.1 January 12, 2016 event. Z-orientation is red, N-orientation is green, E-orientation is blue.

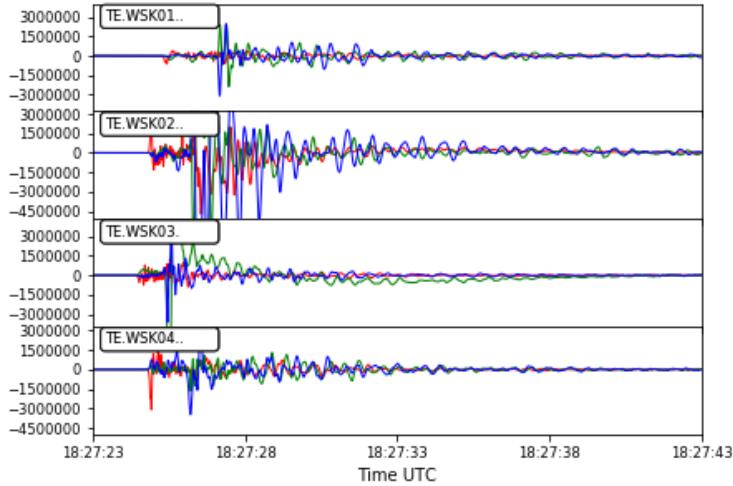


Figure 2.5: WSK waveforms from 4 stations for M_W 4.1 January 12, 2016 event. Z-orientation is red, N-orientation is green, E-orientation is blue. These stations are located significantly closer to the epicenter than DSA stations (Figure 2.4)

This distribution provides relatively good angular coverage, except for 85°towards the North-West of the event. Stations were chosen based on the SNR of the recorded waveforms for the 19 events used as "parents" in the MFA. In this study, the TransAlta catalog maintained by Nanometrics is used to generate parent events for the matched-filtering detection. For the workflow testing

I use continuous data from the nineteen stations listed in Table 2.1 below.

Table 2.1: Stations available for this study in the Crooked Lake area

Network	Station	Latitude	Longitude	Elevation (m)
RAVEN	SWHSA	54.8994	-116.7518	914
RAVEN	BDMTA	54.8129	-118.9149	935
RAVEN	STRPA	55.6606	-115.8323	761
RAVEN	BRLDA	54.0920	-117.4038	1224
UC/DSA	DSA02	54.4128	-116.7649	818
UC/DSA	DSA11	54.0401	-116.6920	1070
UC/DSA	DSA12	54.3683	-116.2924	950
UC/DSA	DSA13	54.2161	-116.6413	985
UC/DSA	DSA14	54.1945	-116.9317	936
UC/DSA	DSA15	54.0424	-116.4800	1061
XTO	XTO105	54.2249	-116.8332	1062
XTO	XTO106	54.2810	-117.0671	977
XTO	XTO107	54.4149	-117.1775	858
XTO	XTO108	54.0966	-116.6554	1023
XTO	XTO109	54.3320	-117.3216	1009
WSK	WSK01	54.4745	-117.2348	828
WSK	WSK02	54.4597	-117.3260	842
WSK	WSK03	54.4168	-117.2722	858
WSK	WSK04	54.3827	-117.3599	916

2.2.2 Tools

Currently Python is one of the most popular programming languages for seismological research. It has an extensive standard library, flexibility for interactive scripting and a large variety of freely available scientific modules that cover most needs in developing scientific processing workflows (Krischer et al, 2015).

Python open-source packages such as NumPy, SciPy, Matplotlib, IPython, Pandas, ObsPy, lxml, PyQt, ObsPy, and EQcorrscan enable the scientific community to construct complete work-

flows in Python. These vary from reading locally stored data, requesting data from one or more different data centers and carry through to signal analysis and data processing, as well as visualizations through different graphic user interfaces (GUI) and web applications, output of modified/derived data and the creation of publication-quality figures. ObsPy, Matplotlib, ObsPyck and EQcorrscan packages are used in the workflow developed in this thesis.

2.2.3 ObsPy and EQCorrscan

ObsPy is an open-source project dedicated to providing a Python framework for processing seismological data (Beyreuther et al., 2010). It allows users to parse through common file formats, use clients to access data centers and provides access seismological signal processing routines which allow the manipulation of seismological time series. ObsPy has earned a large world-wide rate of adoption in the scientific community. Applications successfully using it include time-dependent and rotational seismology, big data processing, event relocation, and synthetic studies about attenuation kernels and full-waveform inversions to name a few examples (Beyreuther et al., 2010). All functionality is well documented and the ObsPy gallery (<http://www.obspy.org>) gives a good impression of the wide range of possible use cases.

EQcorrscan is a Python-based package for the detection and analysis of repeating and near-repeating seismicity. EQcorrscan contains an efficient, multi-parallel, matched-filter detection routine, as well as routines to implement subspace detection, and detection based on brightness. EQcorrscan uses Obspy bindings when reading and writing seismic data, and for handling most of the event meta-data, which ensures that detections can be easily migrated between softwares (Frank, 2014). This package was written by Calum Chamberlain and Chet Hopp of Victoria University of Wellington, New Zealand (Chamberlain et al., 2017).

One of the major benefits of this package is that it allows parallel processing of the templates which significantly reduces the processing time for the matched-filter routine. Moreover this

package has a capability to include other algorithms including subspace detection and brightness source-scanning. Finally it contains *obspy.core.events* integration, which opens up lots of other functions from other libraries like Seishub and hypoDDpy.

2.2.4 Workflow

The matched-filter algorithm utilizes known events with high SNR ("parent" events) and searches for similar waveforms ("child" events) via cross correlation.

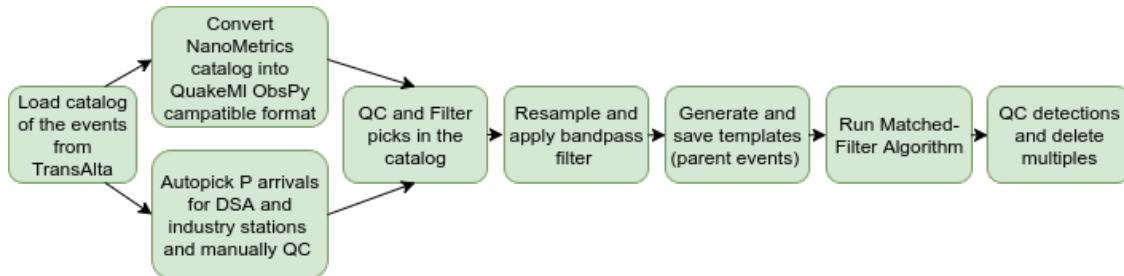


Figure 2.6: Workflow for the application of the MFA algorithm

To facilitate easier code development, simplify quality control (QC), and streamlining testing, the workflow was broken down into following blocks (figure 2.6):

1. Acquire events from a pre-existing source, such as a published earthquake catalog.
In this study, the Nanometrics TransAlta catalog was used. Between January 1, 2016 and March 31, 2016 a total of 36 events were detected. The 19 events with the magnitude above 2.0 are selected as "parent" events. The catalog is read in Nanometrics QuakeML format.
2. Convert Nanometrics/TransAlta catalog into IRIS QuakeML catalog that can be easily read into ObsPy. This catalog contains P and S arrivals for the RAVEN stations.
3. First arrivals for the parent events are autopicked and QCed for the UC/DSA, XTO and WSK industry arrays.

4. A pre-selected window of the template waveforms beginning 0.1 seconds before the S-wave arrival are picked on both the vertical and horizontal components includes the S-wave arrival and S-coda. Both templates and continuous data are band-pass filtered from 0.5 to 10 Hz before applying MFA. The reason for using S-wave and coda for MFA is due to higher SNR of the post S-wave arrival signal over post P-wave arrival in the "parent" events.
5. Using Obspy functionality access RAVEN data directly from IRIS (through FDSN server) and download it in predefined segments. That's why it requires minimum input from the researcher. Data for UC/DSA and industry data are available from external storage.
6. Evaluate "child" events and delete multiples.

2.2.5 Parameter Optimization

To maximize the number of detections that can be later located and used for the interpretation purposes, one should optimize input and detection parameters for the MFA. The number and quality of "child" events detected depends on how one chooses the following parameters: appropriate selection of "parent" events, length of the template, type of trigger, value for the trigger that will constitute a detection, and finally the distance of stations from the epicenter of detected events. In this study, I concentrate on evaluating the quality and quantity of the detections based on length of the "parent" events and trigger value, taking into account different networks available.

The distance from the hypocentre of the events that I attempt to detect to the stations is extremely important. In particular, the closer the station is to the hypocentre, the more likely it is to have a detectable signal in the waveform. Therefore, I organize four available network stations in the order of average distance from the January 12, 2016 event to the station in every network. The WSK array is the closest, followed by the XTO array, DSA array

and finally chosen station from RAVEN. I optimize MFA detection parameters for each network and come up with the set of parameters for the length of the template and detection threshold value.

Before optimizing the parameters, it is a good idea to find an optimal type of trigger that will be used for matched-filtering. Changing the type of trigger should affect the amount and quality of the detections. EQcorrscan detects both positively and negatively correlated waveforms. There are three thresholds that one can use:

1. **Median Absolute Deviation (MAD)** is the most commonly used detection statistic in matched-filtering. MAD threshold which is calculated as follows:

$$MAD = \text{threshold} * \text{median}(\text{abs}(cccsum)) \quad (2.1)$$

where $\text{abs}(cccsum)$ is a the absolute cross-channel correlation sum.

2. **Absolute (A)** threshold is a true absolute value based on the following value:

$$A = cccsum \quad (2.2)$$

3. **Average channel correlation (ACC)** threshold is based on the mean values of a single-channel cross-correlations assuming all data are present as required for the template, e.g.:

$$ACC = \text{threshold} * cccsum / \text{len}(template) \quad (2.3)$$

The absolute value of the correlation sum is used here as the detection statistic. To define a detection threshold the median absolute deviation (MAD) is used. MAD is an estimator of the variability in a distribution that is robust with respect to outliers, which in this case would correspond to positive detections.

First, I evaluate how the length of the parent events affects the number and quality of the detections. It is important to note that S-wave is the most important part of the signal as it has the higher

SNR compared to P-wave. In this study, the length of the template waveform includes 0.1s before S-wave arrival and S-coda. Average length of the S-coda on the available stations lasts between 8 and 12 seconds. To test the sensitivity of the detections to the length of the templates I use 8, 10, and 12 seconds "parent" event templates. Decreasing the length of the template will lead to the loss of some S-wave coda, but keeps the highest SNR aspects of it, which should lead to more detection.

Secondly, I test the threshold value to maximize the detection of child events, while minimizing the false detection rate. For a normally distributed random variable the standard deviation is $1.4826 * \text{MAD}$ (Ruppert, 2010). I test the threshold for a positive detection between 4 and 8 times MAD, which corresponds to 2.7 and 5.4σ above the mean for the normal case. These levels were chosen to see how well it can suppress spurious detections while retaining as many legitimate detections as possible. The probability of exceeding 2.7 and 5.4σ for a normally distributed random variable is approximately 1.0×10^{-3} and 3.3×10^{-8} respectfully. Decreasing threshold values will lead to the increase in the amount of detections, however the SNR of the detections will suffer as the result. This approach ensures that "child" events that are being generated are of good quality and can be further located.

Finally, I compare the results from running an MFA on regional networks like RAVEN and UC/DSA and local industry XTO and WSK networks.

2.3 Results

In this section, I provide an overview of the results of the MFA optimization for the available dataset. Specifically, I demonstrate the effect that the different "parent" events length and threshold value have on the SNR of the detections. This is followed by the results from the investigation of whether one should use different input parameters for MFA for different networks in the Kaybob Duvernay area.

2.3.1 Length of the Template

Before optimizing the threshold value, it is necessary to identify the optimal length of the template. I tested 5 sets of "parent" events with length varying from 8 to 12 seconds. All of the tested events start 0.1 second before the S-wave arrival to capture the high SNR part of the arrivals. I tested other values for the pre-event time window, but the differences in results were negligible.

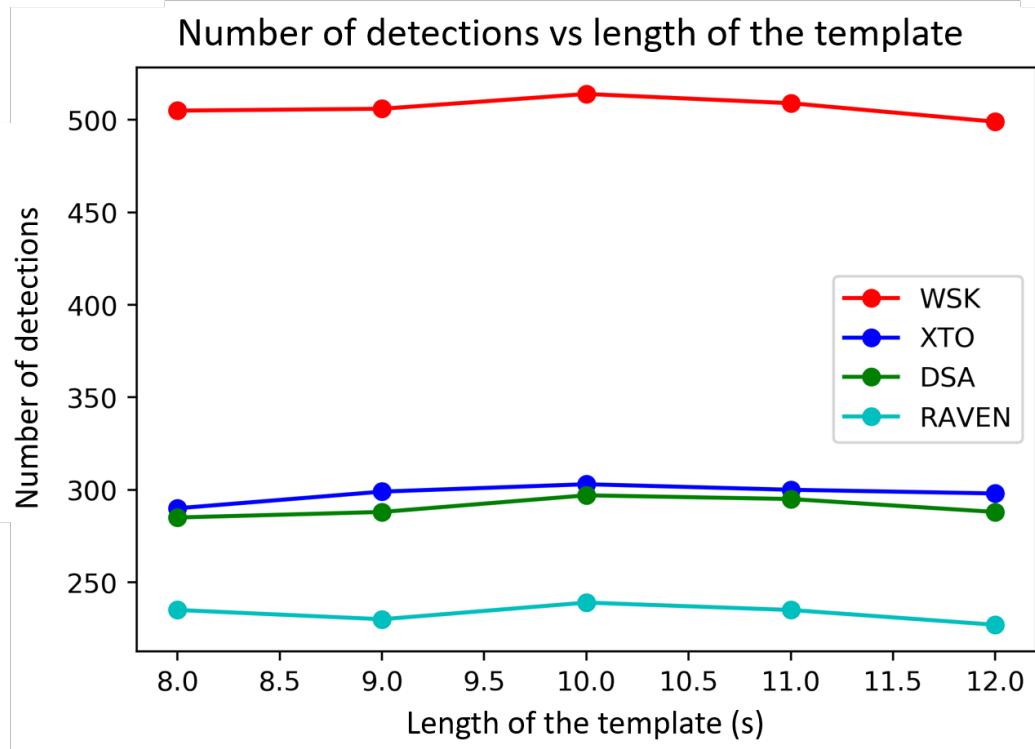


Figure 2.7: Number of detections (January 5-19, 2016) versus the length of the template used in the MFA applied to each seismic network independently.

For the testing purposes all of other parameters are kept the same including MAD threshold, which is set to 6 for the consistency of results. Results of this testing can be seen in the Figure 2.7 which shows relatively small variability (<4%) in the amount of registered detections. Important to notice here that for all seismic station networks the 10-second long template generated the greatest number of detections, which indicates that 10-second "parent" event templates that start 0.1 second before first S-wave arrival would give us the optimal solution.

2.3.2 Threshold Value

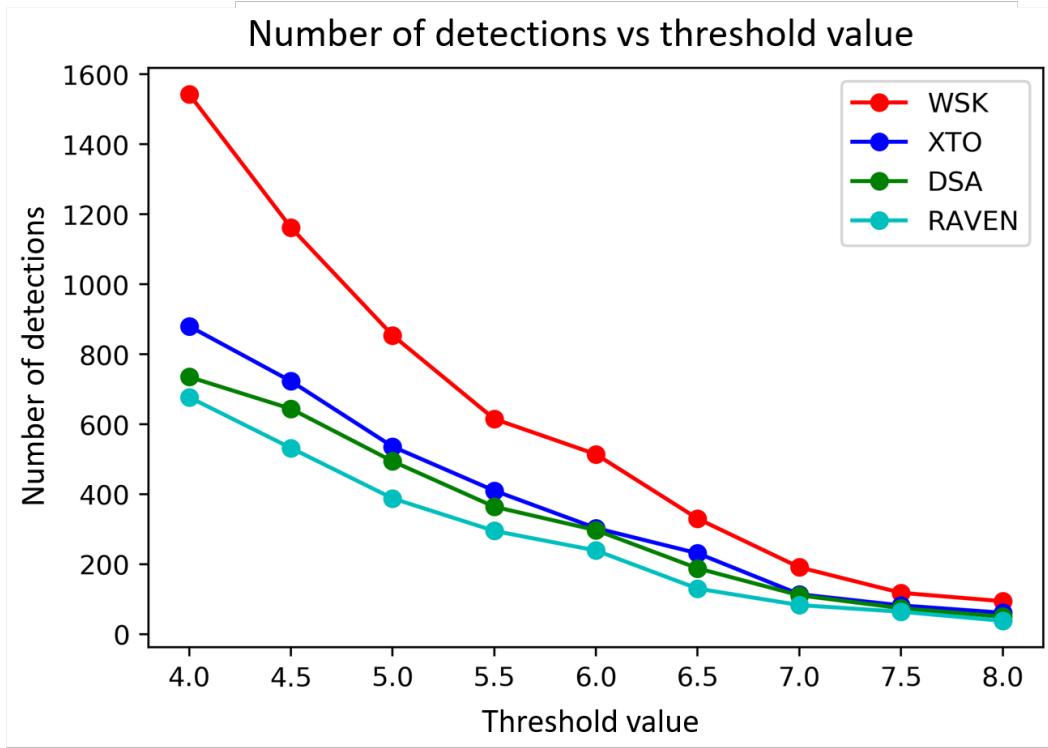


Figure 2.8: Number of detections (January 5-19, 2016) versus the threshold value used in the MFA applied to each seismic network independently.

By decreasing the threshold value of MAD, MFA should detect larger number of events. This might lead to having more "false" detections that are based on the background noise or extremely weak signals that can only be detected on a one or two stations, and would not be useful for further analysis, as it would not be possible to locate them. On the other hand, by increasing the threshold value MAD to high value, the quality of the detections would go up, which would lead to using less memory and less computational time when further steps of analysis will be taken. However, one might miss some events that otherwise would be detected, if less conservative MAD was used.

To find the optimal MAD parameter, I ran the MFA algorithm with 9 different MAD values ranging from 4 to 8, keeping everything else constant including the template length of 10 seconds,

which was previously identified as optimal template length to use. The cumulative number of detections between January 5-19, 2016 for all of the available seismic stations can be seen on Figure 2.8. (Individual distribution for each tested threshold can be seen in Appendix B). Having a threshold value of 4 allows one to detect over 1500 events on the local WSK industry array. However, a significant amount of the detections with triggers MAD <6.0 did not lead to meaningful events, because it is impossible to algorithmically identify the P or S-arrival times on the waveforms from those detections. By visually inspecting the waveforms of triggered events, as well as running STA/LTA, and AR-AIC + STA/LTA algorithms I found that the minimum acceptable MAD threshold that should be used for the MFA is 6. Graph showing the distribution of the detections between January 5 and 19, 2016 for each of the station networks can be seen in Figure 2.9.

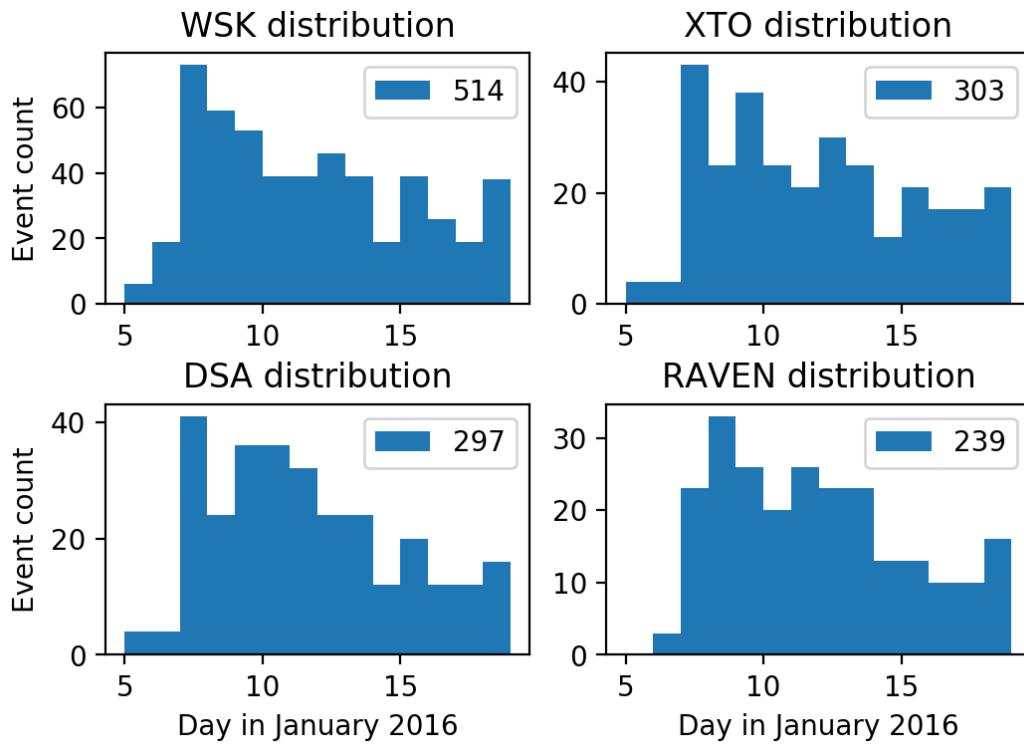


Figure 2.9: Distribution of the detections for each of the available seismic station networks with MAD threshold value of 6 and template length of 10s for the the period of January 5-19, 2016.

2.3.3 Local vs Regional Networks

In this section I look at how quantity and the quality of detections varies depending on whether detection was done on the local industry arrays or the regional array. Firstly, based on Figure 2.9 the WSK array detected the largest number of events. This is expected, because WSK stations are the closest to the area of induced seismic activity, as well as have the best angular distribution with the biggest gap being only 85% to the North-West. As a result, WSK stations detected over 99% of all of the events detected by XTO, RAVEN and UC/DSA station arrays. Secondly, XTO array detected 95% of the detections by UC/DSA array and 95% of the RAVEN stations. Combinations of WSK and XTO array stations detected all of the events that were detected by UC/DSA and RAVEN array stations. Finally, UC/DSA array detected 92% of the RAVEN array detections. An explanation for this is that UC/DSA array is lacking spatial distribution and therefore does not detect some of the events that RAVEN array does due to having stations North-West and North-East of the main activity area.

2.4 Discussion

In this section, I discuss the influence of the studied parameters on the quantity and quality of the detected by matched-filtering "child" events. I also provide recommendations regarding the application of the MFA to the Kaybob Duvernay, as a result of this study.

This workflow has been applied to a small dataset which encompasses January 5-19, 2016 and because it is based on Python/ObsPy it allows direct/immediate access to multiple expanding seismic networks through IRIS. Due to availability of multiple seismic arrays in the area, one can use this workflow to integrate even more data for the successful creation of the new catalogs.

2.4.1 The Role of Parent Events and MAD Threshold Value

The number of detections used in the location determination increases the level of confidence with which the location of the event is determined. Therefore the goal is to detect the signal from the same event at as many stations as possible. Moreover, greater the number of templates, the more accurate the interpretation of induced seismicity.

As one can observe from the results, the quantity of detected "child" events is extremely sensitive to the value of the threshold used. Figure 2.8 illustrates how decreasing the detection threshold can lead to more detections. Unfortunately the lower SNR that detections are triggered at with the lower MAD prevents the autopicker in detecting P- or S-wave arrivals and therefore makes those detections of little to no use for interpretation. Through manual and automatic picker testing, I established the optimal threshold level of 6 that will be applied later in this thesis.

Another important factor, was the length of the "parent" templates used. Within used testing ranges, the number of detections is not as sensitive to the length of the "parent" templates, however on all of the arrays I noticed that 10-second templates that start 0.1 second before first S-wave arrival worked best.

2.4.2 Implications on the Use of Regional Stations in the Presence of Local Arrays

Proximity to the study area as well as spatial resolution clearly have an impact on the number of detections by the seismic monitoring array. Regional networks play an important role as early detection tools, where data is available to researcher and public for quick access and processing. Locally installed arrays represent an opportunity to expand existing catalogs with the events that were not picked up by the regional stations, as well as to help with the more precise location of the previously detected events.

The RAVEN array works as a regional array, which is extremely easy and fast to access with seamless integration with Obspy. Both the WSK and the XTO were installed by industry and are located closer to the existing hydraulic fracturing operations. Local arrays allow one to monitor seismic activity with the stations which are located significantly closer. The UC/DSA array is an intermediate array between XTO and RAVEN. It detects many of the same events as XTO array, while having inferior spatial resolution than RAVEN or XTO. The WSK seismic array detected the significant majority (>98%) of the events detected by other arrays. Moreover, due to its proximity to the induced seismicity, WSK array managed to detect events that were not picked up by other arrays.

In light of this results, careful inspection of the distribution of stations with respect to the area of interest on the map is recommended, to better evaluate the importance of the stations/arrays. It is advisable to run simulations to evaluate detection capabilities of proposed stations depending on expected location of induced seismicity. In this case, WSK and XTO local industry arrays are of crucial importance as they provide the most complete catalog and the most accurate locations due to their proximity. Regional catalogs have an important role to play in preliminary analysis and they can complement the local arrays depending on the relative locations of each individual stations and seismic events that one is trying to detect.

2.5 Conclusions

Continuous data recorded using 19 stations in the RAVEN, UC/DSA, XTO and WSK broadband seismograph arrays for the time period of January 5-January 19, 2016 and TransAlta catalog were used to optimized matched-filtering analysis for further application to larger dataset and comparison to convolutional neural network detector. This exercise resulted in over 500 new unique "child" events (detections) to add to the previously sparse public catalog.

I optimized two crucial parameters in the matched-filtering algorithm to use for generating catalogs. Specifically, the catalog generation was optimized by selecting parameters, such as length of the "parent" events as well as threshold type and value. Recognizing higher SNR of the S-wave I identified a 10-second window containing mostly S-wave arrival and S-wave coda as being optimal "parent" event length. Selecting this window length resulted in detecting up to 4% more events than choosing longer window.

By analyzing different threshold types and triggers, I identified an optimal combination for our study. Median Absolute Deviation (MAD) threshold with the trigger of 6.0 was selected based on optimal amount and SNR of detections that it generates. It was selected as a universal detection criteria and will be used further in the Kaybob Duvernay area for application to the bigger dataset.

Chapter 3

TRAINING AND OPTIMIZATION OF CONVOLUTIONAL NEURAL NETWORK FOR DETECTION OF INDUCED SEISMICITY

This chapter illustrates development and optimization of a Python-based workflow for detection of the induced seismic events from 3-component continuous recordings using a convolutional neural network (CNN). I set up an earthquake detection as a supervised classification problem and train a convolutional neural network to detect earthquakes from the continuous waveforms. I then evaluate two different CNN architectures, and two different loss functions to find an optimal approach for the detection of induced events.

3.1 Introduction

3.1.1 Basics of Neural Networks

In 1989 Dr. Robert Hecht-Nielsen (inventor of the first neurocomputers) defined artificial neural network (ANN) as: "... computing systems made up of a number of simple, highly interconnected processing elements, which process information by their dynamic state response to external inputs" (Caudill, 1989). ANN are typically organized in layers. Each layer is made up of a number of interconnected nodes. Each node contains an activation function which helps to introduce non-linearity to the ANN. Data is inputed into the network via the input layer, which is connected to the hidden layer where the actual processing is done via a system of weights and biases. The last hidden layer then is connected to the output layer where the task of classification or regression is performed (Goodfellow et al., 2016).

3.1.2 Why Use Neural Networks?

In recent decades the volume of seismic data has increased significantly. Automatic and more efficient detection methods are needed. The ability of multilayer neural networks to learn high-dimensional, non-linear, complex mappings from large collections of examples makes them obvious candidates for time-series pattern recognitions (LeCun et al., 1998). In the traditional model of pattern recognition, a customized feature extractor gathers relevant information from the input and eliminates irrelevant variabilities. A trainable classifier then categorizes the resulting feature vectors (matrices) into different classes. In this scheme, fully-connected multilayer networks can be used as classifiers. A more interesting approach is to eliminate the feature extractor, feeding the network with **raw** inputs (3 component continuous seismic recordings), and to rely on back-propagation to turn a couple first layers into an effective feature extractor. While this can be done with an ordinary fully connected feed-forward network (ANN) successfully, there are some fundamental problems that need to be addressed:

- Typical input 10s long 3 component (100Hz) seismic recording are large and contain 3000 variables (i.e. data samples). A fully-connected first layer with, for example 100 hidden units, would contain 300,000 weights. Overfitting problems are likely to occur if training data is scarce. Moreover, the memory requirement for that many weights may rule out certain hardware implementations.
- Another deficiency of unstructured neural nets is that they have no built-in invariance with respect to translations, or local distortions of the inputs (LeCun et al., 1998). Before being sent to the fixed-size input layer of a neural net 2D or 1D signals must be appropriately size-normalized and centered in the input field. Unfortunately, no such preprocessing can be perfect. In principle, a fully-connected network of sufficient size could produce outputs that are invariant with respect to such variability in the inputs. Unfortunately, learning such a task is likely to result in multiple units with identical weight patterns positioned at various locations in the

input (LeCun et al., 2015). Learning the weights for such configurations requires a very large and complete training dataset to cover the space of possible variations. Conveniently, convolutional neural networks shift invariance is automatically obtained by forcing the replication of weights across space.

- A further deficiency of fully-connected architectures is that the topology of the input vector is ignored. The input variables can be presented in any fixed order without affecting the outcome of the training. On the contrary, time-series data have a strong 1D structure: variables that are temporally nearby are highly correlated. Local correlation is the reason for the advantage of extracting and combining localized features before running a detector. Convolutional networks force the extraction of local features by restricting the receptive field of hidden units to be local.

3.1.3 Convolutional Neural Networks

Convolutional neural networks (CNN) have proved to be successful in computer vision, time series analysis, and many other domains due to the significant improved computational capabilities (Krizhevsky et al, 2012, He et al., 2016, Huang et al., 2017). LeNet is known as the first kind of CNNs (LeCun et al., 1995). CNNs received a widespread recognition in 2012, when AlexNet won the ImageNet competition (Krizhevsky et al., 2012). Authors of AlexNet introduced fully connected and max-pooling layers which became a foundation of most CNN architectures developed in the last 7 years. After original AlexNet, a variety of different structures were proposed and tested, such as VGGNet (Simonyan and Zisserman, 2014), ResNet (He et al., 2016), GoogleNet (Szegedy et al., 2017), DenseNet (Huang et al., 2017).

The first published earthquake detection method using a CNN architecture is called "ConvNetQuake" (Perol et al., 2018). It is a variant of VGG network with 10 layers: 8 convolutional, 1 flattening layer, 1 fully connected layer. Perol et al. (2018) successfully applied ConvNetQuake

to detect induced seismicity events in Oklahoma. "DeepDetect" is another example, where a cascaded region-based CNN was developed to capture earthquake events in different sizes while incorporating contextual information to improve features for each proposal (Wu et al., 2018). The backbone of "DeepDetect" is a DenseNet structure, which has 64 layers and generates positive results to laboratory earthquakes (Wu et al., 2018).

In the following sections, I describe some fundamentals of neural networks, then develop and discuss a few possible architectures and loss functions. I then finally perform a hyperparameter tuning using the architecture and loss function combination that had the best performance.

3.2 Data, Methodology and Tools

Regardless of the specific CNN architectures, all of them consist of several common components including convolution layers, activation layers, batch normalization, and a loss function. Below, I provide brief descriptions of these components, as well as discuss data used for training and testing of the CNNs and tools used to develop these systems.

3.2.1 Data

The dataset used for the optimization of the convolutional neural network model contains recordings from three-component seismic data using the following regional and local networks (figure 3.1): Alberta Geological Survey installed Regional Alberta Observatory for Earthquake Studies Network (RAVEN) and University of Calgary UC/DSA array. The reason why I use only these two arrays (and not WSK or XTO) is because only RAVEN and UC/DSA array were continuously recording and we have access to the data during the period January 1 - June 30, 2016. Selected stations provide limited azimuthal coverage relative to the events used for training. The signals from all of the stations are converted to 100Hz on three channels: Z oriented vertically, N oriented North-South, and E oriented West-East. An example of a recorded event from one of the local

networks installed in the area is shown in figure 2.4. A catalog containing 450 events, which was provided by one of the operators in the area, was used to optimize and train the neural network. I divided the continuous waveform data into monthly streams. I normalized each stream individually by subtracting the mean over the month and dividing by the absolute peak amplitude. I extract two types of 10-second long windows from these streams: windows containing events and windows free of events. After data augmentation (further discussed in 3.4.1) by adding extra positive (event) examples I end up with $N = 1,543,511$ windows available for training. $N_+ = 2594$ are positive examples and $N_- = 1,540,917$ negative examples.

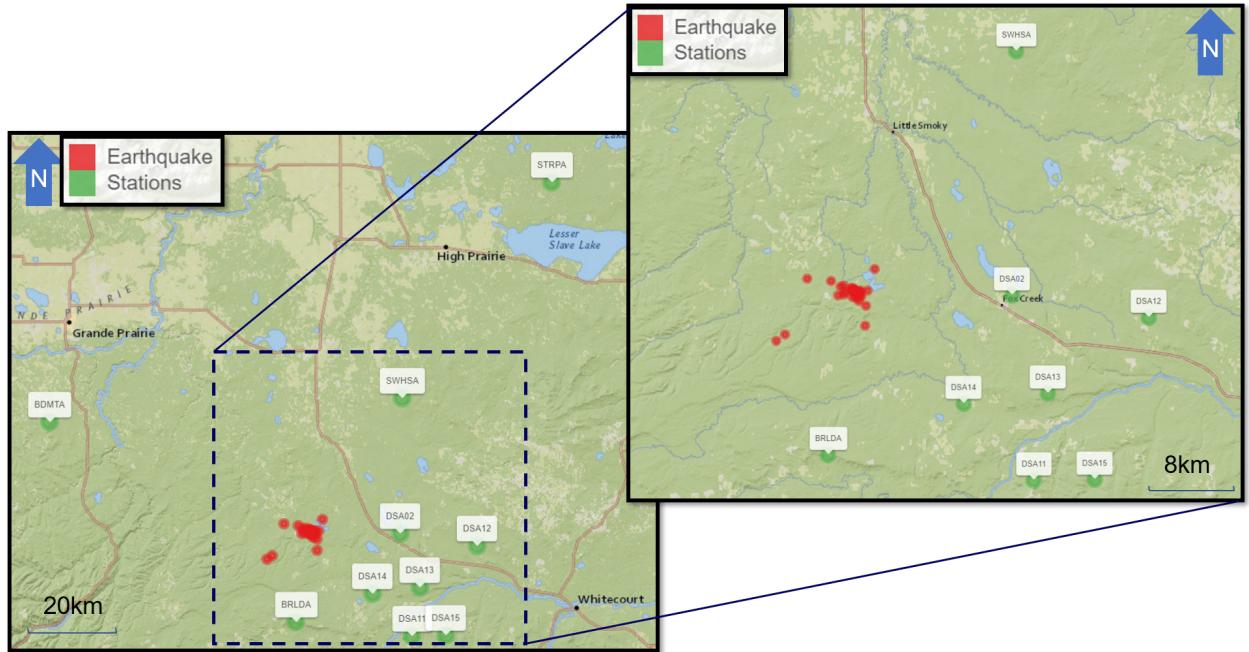


Figure 3.1: Location of seismic stations and events used for training of the convolutional neural network. UC/DSA stations (start with DS) are the University of Calgary DS network stations. Other stations were installed by the AER (RAVEN network).

I set up an earthquake detection as a supervised classification problem and train a convolutional neural network to detect earthquakes from the continuous waveforms. As input, this method takes a 3-channel 10 second window of waveform data and predicts whether it contains a signal in it. For every event in the continuous recordings, it contains $10^3 - 10^5$ examples of noise. This leads to the

class imbalance problem, that I discuss in the next subsection 3.2.2.

3.2.2 Class Imbalance

Most machine learning algorithms work best when the number of instances of each class are roughly equal. Class imbalance references to the situation where the number of a class of data is far less than the total number of another class of data (Provost, 2000). This applies to our dataset as for every event in the continuous recordings, our dataset contains 10^3 - 10^5 examples of noise. This imbalance causes two problems: (1) training is inefficient as most of the signal samples are easy negatives that contribute no useful learning signal; (2) en masse, the easy negatives can overwhelm training and lead to a degenerate model. A common solution is to perform some form of dataset augmentation to create more positive events (Cui et al., 2016, Kvamme et al., 2018). On top of using data augmentation I also test the use of focal loss (Lin et al., 2018), which is designed to address class imbalance by down-weighting inliers (easy noise examples) such that their contributions to the total loss is small even if their number is large.

3.2.3 Convolution Layer

A convolution layer consists of filters that are used to extract the feature map from signals. The discrete cross-correlation operation for time-series data is defined as:

$$X'_i = \sum_n K_n * X_{(s-1)*i+n}, \quad (3.1)$$

where X is the input signal, X'_i is the value of the i^{th} location of the output signals, K is the trainable kernel, and s is the stride between each sliding location of the kernel.

3.2.4 Max-pooling Layer

A pooling layer is a new layer added typically after the convolutional layer, which operates upon each feature map separately to create a new set of the same number of pooled feature maps. Pooling layer reduces the size of each feature map by a factor of 2 (assuming a stride of 2), therefore each

dimension is halved as a result (Goodfellow et al., 2016). Max-pooling calculates the maximum value for each patch (typically 2-by-2) of the feature map. Resulting feature map is down-sampled and summarized version of the previous feature map.

3.2.5 Activation Layer

The activation layer is applied to introduce non-linearity. It allows neural networks to increase the level of complexity in learning mapping relationships (Liu & Wang, 2008). Rectified linear unit (ReLU) and sigmoid functions (figure 3.2) are the most commonly used functions as activation layers.

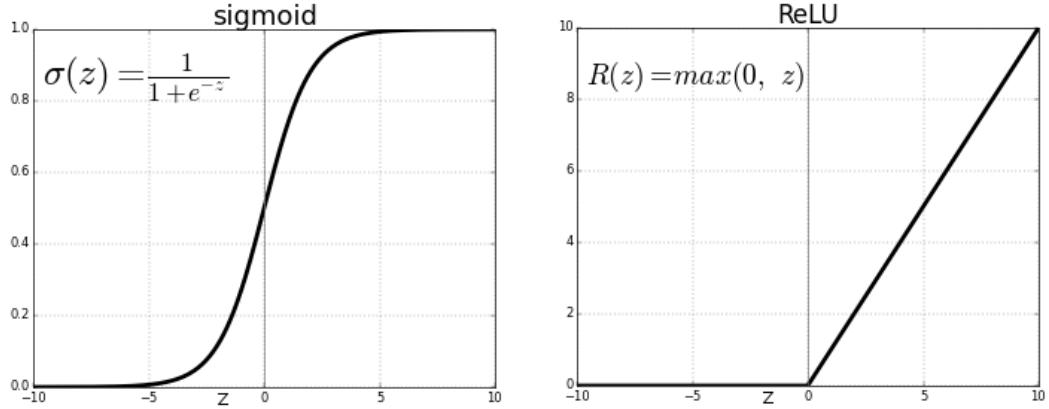


Figure 3.2: Visualization of sigmoid and ReLU activation functions.

- ReLU function was designed to alleviate the problem of gradient vanishing (Nair & Hinton, 2010), is given as:

$$ReLU(x) = \max(0, x), \quad (3.2)$$

where x is the input.

- Sigmoid function is a function that refers to the special case of the logistic function.

It is defined as:

$$Sigmoid(x) = \frac{e^x}{e^x + 1} = \frac{1}{1 + e^{-x}}, \quad (3.3)$$

where x is the input. The sigmoid function is used to restrict the value range of the high-level features before they are fitted into a regressor.

3.2.6 Batch Normalization

A difference in the distributions and statistics of training and testing data can significantly impact the detection results of CNNs. To help resolve this issue, batch normalization can be applied (Ioffe and Szegedy, 2015). In addition, batch normalization can also be helpful in accelerating the training speed by normalizing the inputs for each layer. For each activation x_i :

$$B_{\gamma,\beta}(x_i) = \gamma * \frac{x_i - \mu}{\sqrt{\sigma^2 + \eta}} + \beta, \quad (3.4)$$

where γ and β are parameters to optimize; i refers to i^{th} location of the feature map after a convolution layer; μ is calculated by averaging all values on the same feature map of x_i in the mini-batch. Batch normalization allows each layer of a network to learn by itself a bit more independently of other layers (Ioffe and Szegedy, 2015).

3.2.7 Convolution Block

The convolutional block is a combination of several operations mentioned above including convolutional layers (Conv), a batch normalization (BN), and an activation layer (ReLU)

$$x^{j+1} = \text{ReLU}(BN(\text{Conv}(x^j))), \quad (3.5)$$

where x^{j+1} and x^j are the output and input of the j^{th} convolutional block.

3.2.8 Loss Function

Loss functions for binary classification measure how well the prediction results match training labels (Rosasco et al., 2004). It is important to chose a loss function that will work for our training data. In this chapter I evaluate two different loss functions:

- Cross-entropy (CE) loss for binary classification, which is a continuous function that can be calculated in the following way:

$$CE(p, y) = \begin{cases} -\log(p) & \text{if } y = 1 \\ -\log(1 - p) & \text{otherwise.} \end{cases} \quad (3.6)$$

where $y \in \{\pm 1\}$ specifies the ground-truth class and $p \in [0, 1]$ is the model's estimated probability for the class with label $y = 0$. For notation convenience, let's define p_t :

$$p_t = \begin{cases} p & \text{if } y = 1 \\ (1 - p) & \text{otherwise.} \end{cases} \quad (3.7)$$

The CE loss is shown by the blue curve in Figure 3.3. A notable property of this loss is that even examples that are easily classified ($p_t \gg 0.5$) incur a loss with non-trivial magnitude. When summed over a large number of easy examples, these small loss values can overwhelm the rare class.

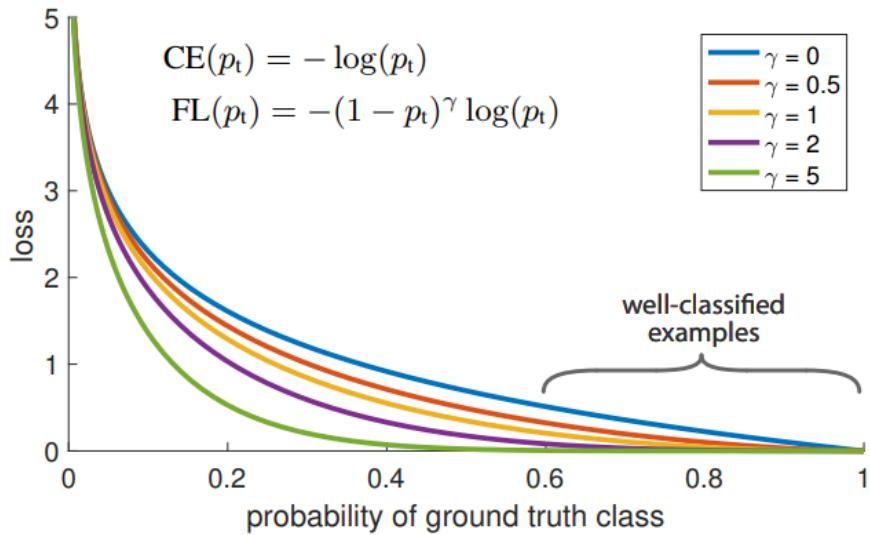


Figure 3.3: Comparison between standard cross-entropy loss (blue line) and focal loss as a function of probability of ground truth class. Modified from Lin et al., 2018.

- Focal loss (FL) for binary classification, adds a modulating factor $(1 - p_t)^\gamma$ to the cross-entropy loss, with tunable focusing parameter $\gamma \geq 0$. Focal loss is defined as:

$$FL(p_t) = -(1 - p_t)^\gamma \log(p_t). \quad (3.8)$$

Visualizations for several values of $\gamma \in [0, 5]$ can be seen in the Figure 3.3. Note two properties of the focal loss. (1) When an example is misclassified and p_t is small, the modulating factor is near 1 and the loss is unaffected. As p_t is approaching 1, the factor goes to 0 and the loss for well-classified examples is down-weighted. (2) The focusing parameter γ smoothly adjusts the rate at which easy examples are down-weighted. When $\gamma = 0$, FL is equivalent to CE, and as γ is increased the effect of the modulating factor reduces the loss contribution from easy examples and extends the range in which the example receives lower loss. Finally, in practice I use an α -balanced variant of the focal loss to allow for even more flexibility:

$$FL(p_t) = -\alpha(1 - p_t)^\gamma \log(p_t). \quad (3.9)$$

3.2.9 Tools

Development and optimization of the workflow is carried out in Python using Obspy (Beyreuther et al., 2010), TensorFlow (Abadi et al., 2015), and EQcorrscan (Chamberlain et al., 2017). TensorFlow is an end-to-end open source platform for machine learning. It has a comprehensive, flexible ecosystem of tools and libraries for us by researchers and developers. Obspy was used to access data centers and process seismological signal. TensorFlow was used for development, training and testing of the CNNs.

3.3 Investigating Different CNN Architectures For Event Detection

In this section I discuss two CNN architectures that were tested and optimized for the detection of induced seismicity. Firstly, I evaluate an architecture proposed by Perol et al. (2018) called ConvNetQuake. Secondly, I propose and evaluate a less deep architecture I call VGG-Junior to evaluate the detection performance of a shallower CNN.

3.3.1 ConvNetQuake

The input to ConvNetQuake (figure 3.4) is a 2-D tensor $Z_{c,t}^0$ that consists of input waveform data from a fixed 10s 3-component window. The rows of Z^0 for $c \in \{1, 2, 3\}$ corresponds to the channels of the waveform. The time index, $t \in \{1, 2, \dots, 999, 1000\}$ contains a 1000 values since I am using 10 second-long waveforms recorded at or downsampled to 100Hz frequency. The model is a feed-forward stack of eight convolutional layers (Z^1 to Z^8) followed by 1 fully connected (FC) layer z that outputs binary classification prediction scores. Each layer contains multiple channels and are represented by 2-D tensors.

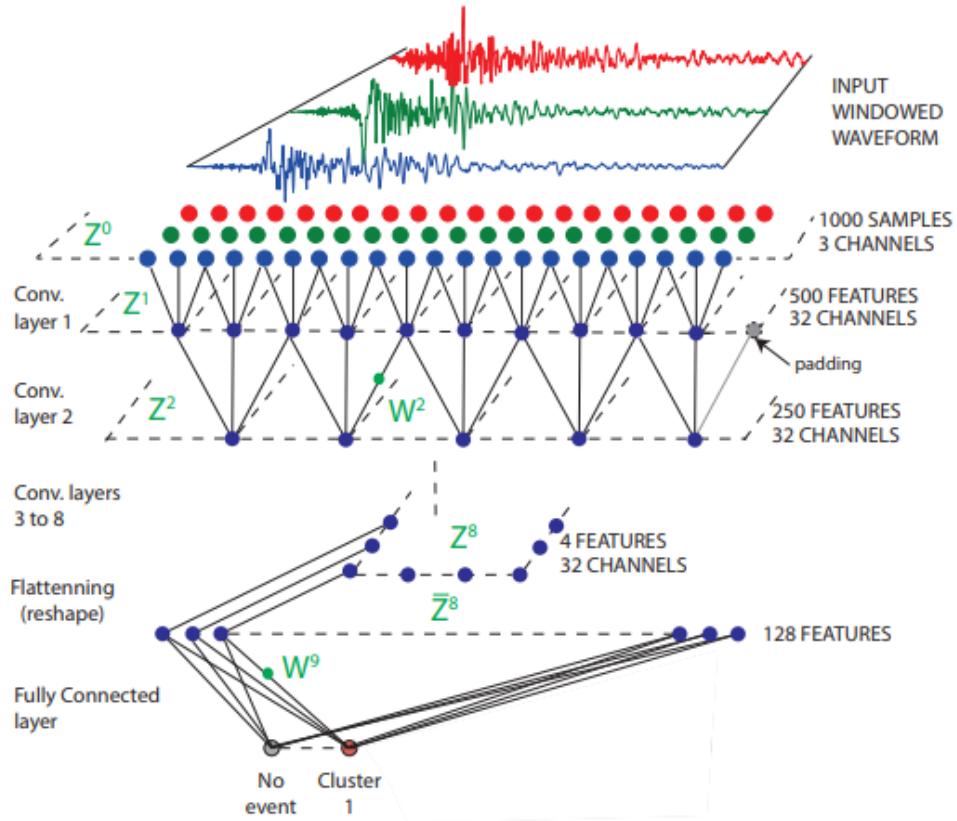


Figure 3.4: ConvNetQuake architecture. The input is a waveform of 1000 points (100Hz, 10s) on 3 channels. Each layer consists of 32 filters that downsample the data by a factor of 2. After the eighth convolutional layer, the features are flattened into a 1D vector of 128 features. Modified from Perol et al. (2018).

Each channel of the convolutional layers is calculated by convolving the channels of the previous layer with a variety of linear 1-D filters, adding a bias term, and applying a point-wise non-linearity:

$$Z_{c,t}^i = \sigma(b_c^i + \sum_{c'=1}^{C_i} \sum_{t'=1}^3 Z_{c,s+t'-1}^{i-1} \cdot W_{cc't'}^i) \quad (3.10)$$

where $i \in \{1, 2, \dots, 8\}$, and $\sigma(x) = \max(0, x)$ is the ReLU activation function. The output and input channels are indexed with c' and c respectively, and the time dimensions t' and t . C_i is the number of channels for layer i . ConvNetQuake uses 32 channels in each of the convolutional layers, while the input waveform (layer Z_0) has 3 channels. It stores the filter weights for each of the layers in a 3-D tensor W^i with dimensions $C_{i-1} \times C_i \times 3$. The biases are stored in a 1-D tensor b^i . All convolutions use zero-padding as the boundary condition. Full architecture with dimensions for each layer can be seen in Table 3.1.

Stage	Layers	Dimensions
Input	-	1000 x 3
Conv1	Conv + BN + ReLU	500 x 32
Conv2	Conv + BN + ReLU	250 x 32
Conv3	Conv + BN + ReLU	125 x 32
Conv4	Conv + BN + ReLU	64 x 32
Conv5	Conv + BN + ReLU	32 x 32
Conv6	Conv + BN + ReLU	16 x 32
Conv7	Conv + BN + ReLU	8 x 32
Conv8	Conv + BN + ReLU	4 x 32
Flattening	-	1 x 128
Fully Connected layer		

Table 3.1: ConvNetQuake architecture. The input is a waveform of 1000 points (100Hz, 10s) on 3 channels. Each layer consists of 32 filters that downsample the data by a factor of 2. After the eighth convolutional layer, the features are flattened into a 1D vector of 128 features.

ConvNetQuake accomplishes down-sampling of data by the factor of two along the time axis in each convolutional layer using strided convolutions with stride $s=2$, which means the kernel slides horizontally in increments of 2 samples. This is equivalent to performing a regular

cross-correlation followed by subsampling with a factor of 2.

Because ConvNetQuake’s kernel size is 3 (small filters), the first few convolutional layers have a local view of the input signal and can only extract higher frequency features. Through progressive down-sampling, the deeper layers have an exponentially increasing receptive field over the input signal. This allows deep convolutional layers to extract low-frequency features.

After the final convolutional layer, network flattens the Z^8 tensor with 4×32 shape into a 1-D tensor with 128 features Z^9 . This feature vector is further processed by a Fully Connected (FC) layer to compute binary class scores z_c with $c \in \{0, 1\}$ calculated in the following way:

$$z_c = \sum_{c'=1}^{128} Z_{c'}^9 \cdot W_{cc'}^9 + b_c^9 \quad (3.11)$$

FC layer allows the network to combine multiple parts of the signal (P, S, coda), and from multiple orientations (Z, N, E) to generate a class score and can detect events that exist anywhere within the input 10s window.

The last step is to apply the function to the class scores to obtain a properly normalized probability, which can be interpreted as a posterior distribution over the classes conditioned on the input Z^0 and the network parameters W and b :

$$p_c = P(class = c | Z^0, W, b) = \frac{e^{z_c}}{\sum_{k=0}^1 e^{z_k}}, \quad (3.12)$$

where $c \in \{0, 1\}$, $W = \{W^1, \dots, W^9\}$ is the set of all weights, and $b = \{b^1, \dots, b^9\}$ is the set of all biases.

ConvNetQuake optimizes the network weights by minimizing a L_2 -regularized cross-entropy function on a dataset. Regularization adds an L_2 penalty on the weights W , and reduces the potential of overfitting (Ng, 2004).

To train a CNN one should use an optimization algorithm that would search for parameters that would minimize the loss function and therefore improve performance of the network. Gradient descent is one of the most important and widely used variations of the optimization algorithm. Gradient descent is an iterative algorithm for finding the minimum of the given function (Goodfellow et al., 2016). The CNNs are trained by optimizing the parameters of the network to obtain the minimum of the loss function. The idea is to pass the training dataset through the hidden layers (filters) of the CNN and then update the weights and biases of each node by computing the gradients using the training examples from the dataset. There are a few variations of the gradient descent algorithms widely used in deep learning (Goodfellow et al., 2016):

- **Stochastic Gradient Descent:** in this variation one training example is passed through the neural network at a time and the weights and biases of each node are updated based on the performance of one training example.
- **Batch Gradient Descent:** in this variation the whole training dataset is passed through the neural network and the weights and biases of each node are updated based on the performance of the model on the whole dataset.
- **Mini-batch Gradient Descent:** is the variation of Stochastic and Batch Gradient Descents. The training set is subdivided into multiple mini-batches. Each mini-batch has a predetermined number of training samples in it. The weights and biases are updated based on the performance of the model on the mini-batch.

Since both the weights and the training data set are too large to fit in memory, ConvNetQuake uses a mini-batched stochastic gradient descent algorithm to minimize the cost function. The training data is divided into multiple groups called batches. The parameters of the CNN are updated every time the single batch of training data is passed through the network. A sequence of batches containing 128 windows each is formed from randomly shuffled large dataset of windows. Since the seismic recordings dataset is typically unbalanced, each batch is composed of 64 windows

of noise and 64 event windows. By using mini-batched stochastic gradient descent we avoid a potential problem of overloading memory, while efficiently training the CNN.

3.3.2 Variation of Shallow VGG Network

The second type of the CNN that I test is a modified version of VGG net (Simonyan and Zisserman, 2014). In particular, I developed the detection method based on shallow VGG Net. I call it "VGG - Junior". It has 4 convolutional layers (fig 3.5) and is somewhat simplified version of ConvNetQuake architecture.

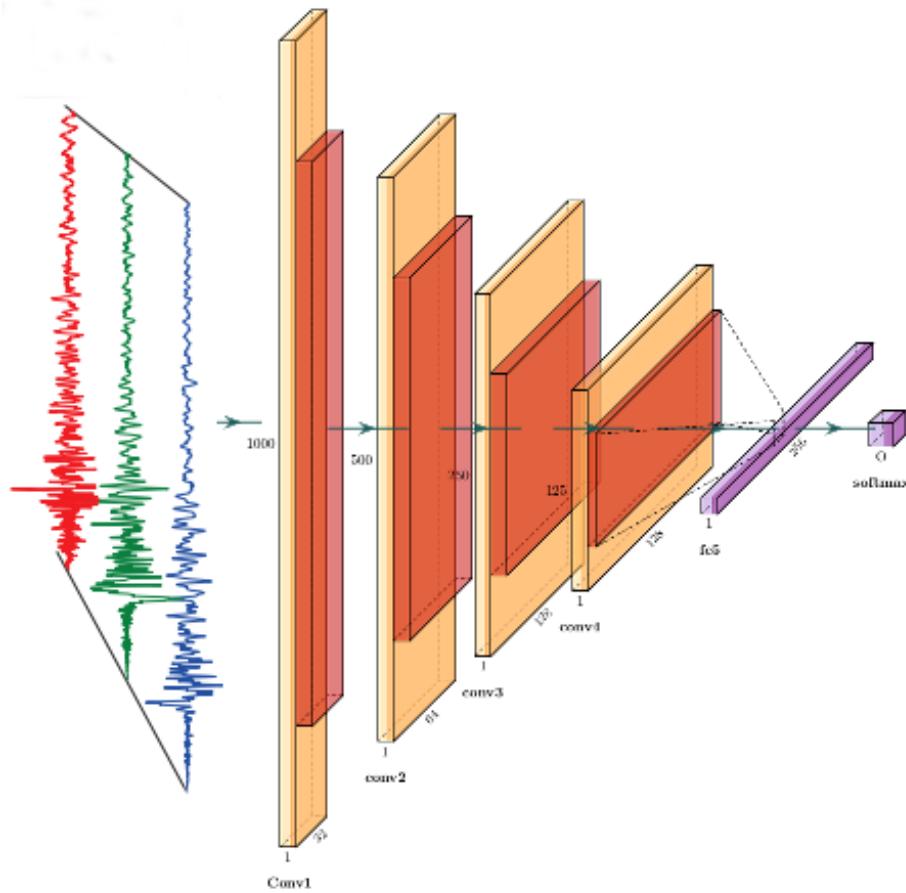


Figure 3.5: VGG - junior architecture. The input is a waveform of 1000 points (100Hz, 10s) on 3 channels. After the 4 convolutional layers, the features are flattened into a 1D vector of 256 features

Table 3.2 illustrates the overall architecture of "VGG-junior". BN stands for Batch Normaliza-

tion, ReLU stands for the activation layer applied, 2 (or 64) refers to the stride of the convolutional kernel or pooling kernel. The last layer is a fully connected layer containing 256 features which is used to determine our predictions. Initially I calculate the loss as logistic loss, however later the effect of focal loss on detection accuracy and training time are investigated.

Stage	Layers	Dimensions
Input	-	1000 x 3
Conv1	Conv + BN + ReLU	1000 x 32
Pool1	Avg pool, 2	500 x 32
Conv2	Conv + BN + ReLU	500 x 64
Pool2	Avg pool, 2	250 x 64
Conv3	Conv + BN + ReLU	250 x 128
Pool3	Avg pool, 2	125 x 128
Conv4	Conv + BN + ReLU	125 x 256
Pool4	Avg pool, 2	64 x 256
Global Pool	Avg global pool, 64	1 x 256
Fully Connected layer		

Table 3.2: VGG - junior architecture. The input is a waveform of 1000 points (100Hz, 10s) on 3 channels. After the 4 convolutional layers, the features are flattened into a 1D vector of 256 features

3.4 Methodology and Workflow

In this section I discuss the dataset augmentation for addressing the class imbalance problem, general workflow used for training the CNNs, and methodology used for systematic evaluation of the performance of each version of the CNN.

3.4.1 Dataset Augmentation

Neural networks like the ones tested here have many trainable parameters. They require a large number of examples for each class to generalize correctly to unseen examples and to avoid overfitting. Therefore instead of solving imbalance problem by limiting amount of negative examples I propose to perform data augmentation to increase number of positive examples for the CNN to learn from. The input into the tested CNN is a single waveform recorded at any of the available

10 stations. I generate additional event windows by perturbing existing ones by adding zero-mean Gaussian noise. This increases the number of positive examples in the dataset from 450 to 2594. For higher SNR events I generate more derivative events that go into the training dataset.

3.4.2 Workflow

The workflow (figure 3.6) consists of preparing the data, training the neural network, and testing the proposed model.

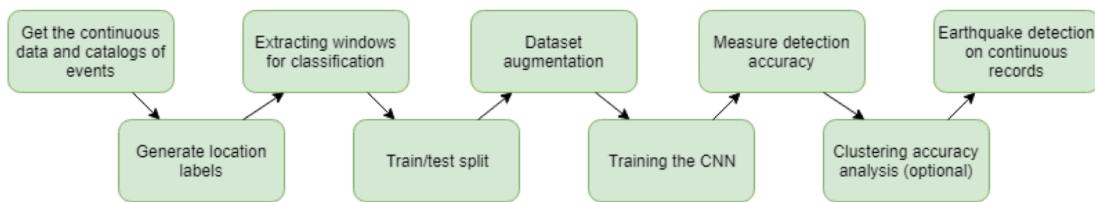


Figure 3.6: Workflow used in this study for the convolutional neural network method.

Regardless of the CNN architecture or loss function used, the important steps in this process are:

- Download the continuous data and catalogs of events.
- Generate two output classes catalogs: events or noise.
- Prepare the data by extracting windows containing noise and signal in the continuous data.
- Randomly split the data into training, validation, and testing sets, where 75% is training, 15% is testing, and 10% is validation set. Training data is used to fit the model. Testing data is used to provide unbiased evaluation of a model performance while tuning model hyperparameters. Finally, validation data is used to provide an unbiased evaluation of a final model performance.

- Perform dataset augmentation by generating more events by adding Gaussian noise to the already existing events. This is done to balance the amount of event and noise windows for the more accurate training of the neural network.
- Train the Neural Network, which is performed once and can be run overnight on laptop.
- Evaluation of the neural network on a validation set. After training, I test the accuracy of our network on validation set of windows not used in the training.
- Originally I set up a problem as binary classification (noise, event). Perol et al. (2018) proposed to cluster seismic events by surface coordinates and instead of just predicting presence of event in the signal, authors proposed to predict the approximate location from where the signal was coming from. Therefore, if more than two clusters are used, one can also evaluate the accuracy of event classes by location.
- Use the model to identify more events in the continuous data.

3.4.3 Methodology

To validate the performance of the proposed CNN architectures and loss functions, I use precision, recall, *F*-score and accuracy as the evaluation metrics, which are defined as:

$$Precision = \frac{TP}{TP + FP}, \quad (3.13)$$

$$Recall = \frac{TP}{TP + FN}, \quad (3.14)$$

$$F-score = 2 \times \frac{Precision \times Recall}{Precision + Recall}, \quad (3.15)$$

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}, \quad (3.16)$$

where TP is "true positive", TN is "true negative", FN is "false negative", and FP is "false positive". F-score is defined as the weighted harmonic mean of the model's precision and recall. It provides a more realistic measure of the model's performance, especially in the case of the highly imbalanced datasets (Goodfellow et al., 2016). I compare the performance of four detection methods: ConvNetQuake with CE loss, ConvNetQuake with FL loss, VGG-Junior with CE loss, and VGG-Junior with FL loss. In particular I first compare the detection accuracy of all the methods and then I compare the overfitting issue of each detection approach.

3.5 Results

3.5.1 Detection Accuracy

Using evaluation metrics defined in equations 3.13, 3.14, 3.15, and 3.16 I evaluate each of the 4 proposed CNNs and loss function combinations and provide in Table 3.3 results of its performance. Importance of using F-score over accuracy as performance metric can be illustrated by considering a case of a model that predicts 'noise' (or negative example) for every input waveform. In this case the accuracy will be 99.99% (due to class imbalance), however precision, recall and F-score would be equal to 0. Therefore, if model only predicts "noise" its F-score is going to be zero and that model would have the worst possible performance based on F-score metric.

Name	Precision	Recall	F-Score	Accuracy	Test time [ms]
ConvNetQuake, CE loss	94.94	96.54	95.73331523	96.7	5.01
ConvNetQuake, FL loss	98.35	98.51	98.42993498	98.1	5.04
VGG-Junior, CE loss	91.11	92.13	91.6171611	91.6	3.87
VGG-Junior, FL loss	95.01	96.78	95.88683247	95.7	3.88

Table 3.3: The classification results and implementation parameters given by proposed CNN methods.

Overfitting is an important phenomenon to look for when training neural networks, due to the large number of network parameters. Overfitting leads to detection models performing well on the training data, while not generalizing well to new data. To ensure, that none of the proposed models

overfit the data, it is critical to monitor performance of the CNNs on both testing and validation sets through accuracy and loss. Table 3.4 shows the number of parameters for both architectures. Use of different loss function does not affect the amount of the parameters.

Final important step is to chose the correct optimization function that would allow for the fastest training of the CNNs. Adam optimizer is used for loss function minimization due to its superior performance when compared to stochastic gradient descent (Kingma et al., 2014). Stochastic gradient descent maintains a single learning rate for all weight and bias updates and the learning rate does not change during training. Adam modifies a learning rate (size of step in updating the weights of the neural network) based on calculations of an exponential moving average of the gradient and the squared gradient (Kingma et al., 2014). Because of the flexibility with regard to the learning rate Adam is currently recommended as the default algorithm to use.

	ConvNetQuake	VGG-Junior
#Parameters	590944	167777

Table 3.4: The number of network parameters of two CNN architectures based on ConvNetQuake (Perol et al., 2018) and VGG-Junior.

ConvNetQuake has 3.5 times more parameters than VGG-Junior, which means that ConvNetQuake might be more prone to overfitting. This difference in the amount of parameters can be explained by shallower architecture of the VGG-Junior, as well as presence of the max-pooling layers. To monitor this I provide the plots of training and validating accuracy and loss in Appendix C. I set a batch size to 128.

3.6 Discussion

3.6.1 Detectability of Different Models

Size of the CNN and the choice of loss function can impact the detectability of the earthquakes. Table 3.3 shows that the deeper ConvNetQuake has higher classification accuracy using both CE and FL loss functions. On the other hand, the detection architecture with fewer layers costs less computationally and also yield the smallest number of network parameters to store among all other CNN-based detection methods. Both CNNs combined with FL as a loss function perform better than CE loss function versions of the same CNNs.

Therefore, I conclude that with small number of training sets, shallower deep networks (like VGG-Junior) is more suitable for use where computational efficiency and minimum use of memory is essential. Deep networks (like ConvNetQuake) are more suitable for use where amount of training data is greater, there are less computational limitations and where classification accuracy (detection accuracy) is more important. Convolutional neural network pruning can be used to decrease the size of the network and enable more efficient training and deployment (Molchanov et al., 2016). Pruning will delete the least active nodes (or filters in case of CNNs) without unduly penalizing performance of the model. Pruning usually results in significantly lighter architecture with less weights and biases to optimize. After pruning, CNN generally maintains good generalizations (weights) in the network and drops the inefficient ones, which leads to faster runtime and less used memory.

3.6.2 Avoiding Overfitting

Increasing the depth (size, complexity) of a particular network increases the accuracy of the detection model. However, by increasing the depth of the CNN one risks introducing overfitting issues. To become more representative and accurate deeper model requires bigger (ideally more class balanced) datasets. Without enough training data, the CNN will tend to over-fit the data

or memorize easier features in the signal instead of learning the intrinsic pattern in the data. To address class imbalance I suggest a combination of data augmentation (by adding different levels of Gaussian noise to existing events) and adopting Focal Loss instead of widely used Cross-Entropy Loss, that is not as effective with imbalanced datasets.

One of the best ways to check whether your model is overfitting the data is to plot training and validation set accuracies and losses. Appendix C shows the training versus validation accuracy and loss plots. One can see that training accuracy is comparable to the validation accuracy for all 4 models tested, which is a good indicator that there are no overfitting issues present.

3.7 Conclusions

Convolutional neural networks are a good example of an idea inspired by biology that resulted in seismological algorithms that compare favorably with other methods. Continuous data recorded using 10 stations in the RAVEN, UC/DSA broadband seismograph arrays for the time period of January 1-June 30, 2016 and local array catalog were used to test two different CNN architectures and two different loss functions to identify the most accurate model to deploy on our data from Kaybob Duvernay region. I name the highest performing model DuverNet and it is based on the ConvNetQuake architecture combined with using Focal Loss function, and Adam optimizer. DuverNet takes as an input a 10s 3 component recording from a single station and outputs a prediction on whether this recording contains an event in it. After 30,000 learning iterations, DuverNet achieved an accuracy of 98.09%, and once deployed takes 5.04ms to analyze one 10s waveform. It achieved precision of 98.35, recall of 98.51 and F-Score of 98.43.

Chapter 4

COMPARISON BETWEEN MATCHED FILTERING AND CONVOLUTIONAL NEURAL NETWORK BASED DETECTION METHODS

This chapter discusses the differences in performance between two Python-based workflows for the detection of induced seismic events from 3-component continuous recordings. First method (chapter 2) is based on application of matched-filtering analysis, while second method (chapter 3) is based on convolutional neural network (CNN) method.

4.1 Introduction

Detecting events present in seismic data is vital for gaining an understanding of physical processes occurring in the subsurface. However, the low magnitude of induced seismic events and the presence of background noise degrades the accuracy of event detection methods. Several algorithms are examined in order to evaluate their performance on synthetic and Kaybob-Duvernay datasets. The matched filtering algorithm (MFA) (discussed and optimized in chapter 2) has low false detection rate and easily detects similar events, while suffering from a number of missed events due to poor capture of waveform variation and long execution time. The highest performing CNN called DuverNet (discussed and optimized in chapter 3) is a method that better generalizes to detecting previously unseen events, and uses less computational time, however it suffers from higher false positives rate and requires extra time and a large dataset for training.

4.2 Data and Methodology

4.2.1 Synthetic data

To assess the detection accuracy of the MFA and DuverNet algorithms and test how well these approaches generalize to the waveforms that are "similar" and "dissimilar" from those in the training set, I perform synthetic data testing. Two templates are referred to as "similar" if they have the same polarity and pulse directions. Two templates are referred to as "dissimilar" if they do not have the same polarity and pulse directions. I quantify this using synthetic seismograms and compare the neural network method to template matching. I generate month-long synthetic waveforms by inserting multiple copies of a waveform template (from real data) over a Gaussian noise floor to form a day-long seismic records. SNR varies from -1 dB to 5 dB in the similar waveforms case and from -1 dB to 8 dB in the dissimilar waveforms case. I generate synthetic seismic records with SNR ranging from -1 dB to 8 dB. The decibel (dB) is a unit of measurement used to express the ratio of one value of a power to another on a logarithmic scale, the logarithmic quantity being called the power level. Two signals whose levels differ by one decibel have a power ratio of $10^{0.1}$, which is approximately 1.259, and an amplitude ratio of $10^{0.05}$, which is approximately 1.122 (Yost, 1985).

The SNR of the signal is defined as follows:

$$SNR = 10 \log_{10} \left[\left(\frac{A_s}{A_n} \right)^2 \right], \quad (4.1)$$

where A_s is the signal amplitude and A_n is the noise amplitude. An example of a day-long synthetic record is shown in the Figure 4.1. For a 10-second long template of 3-channels waveform data \mathbf{m} sampled at 100 Hz, I define the amplitude of a signal as the L_2 norm of the waveform:

$$A_s = \left(\sum_{c=1}^3 \sum_{t=1}^{100 \times 10} m_{c,t}^2 \right)^{1/2}, \quad (4.2)$$

where t is the time index and c is the channel index. Similarly A_n for a 10-second long template of

3-channel waveform data \mathbf{n} sampled at 100 Hz, is defined as:

$$A_n = \left(\sum_{c=1}^3 \sum_{t=1}^{100 \times 10} n_{c,t}^2 \right)^{1/2} \quad (4.3)$$

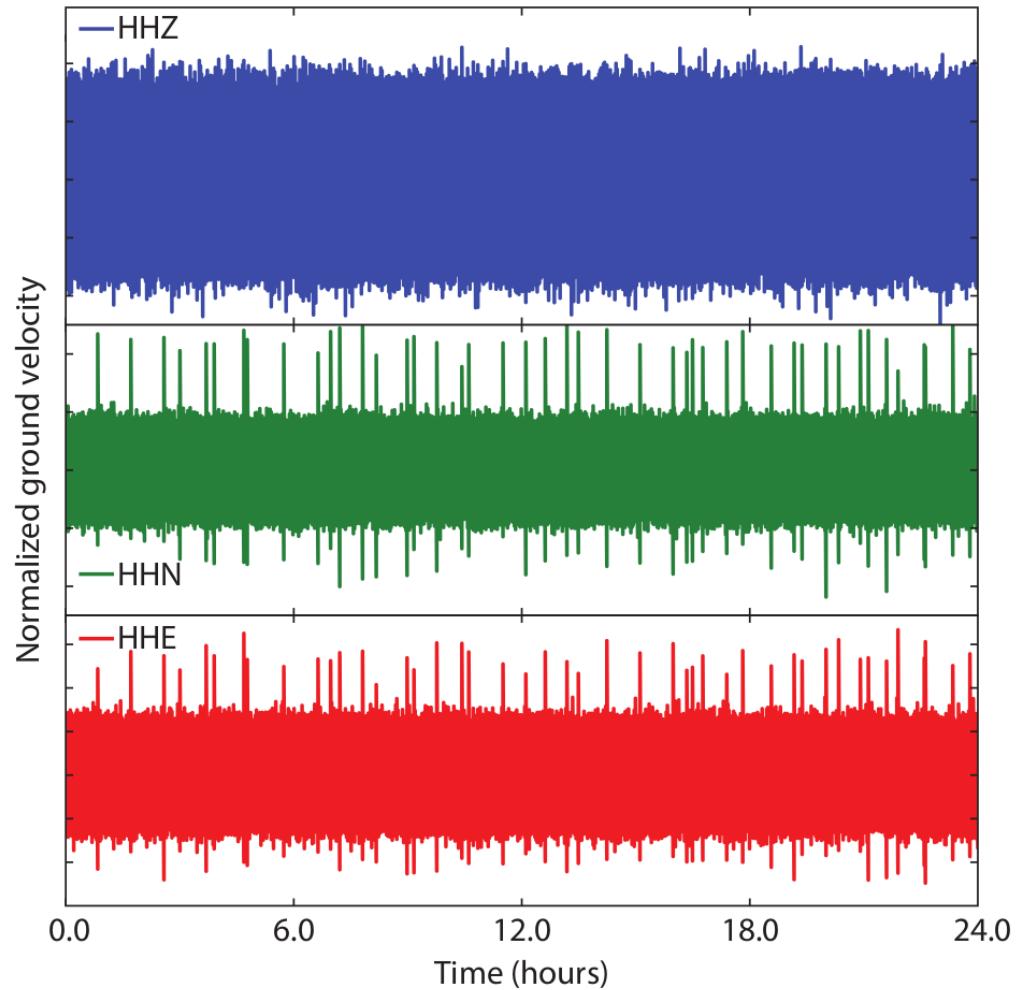


Figure 4.1: Day-long synthetic seismic record. The average SNR of this synthetic seismic record is 3 dB.

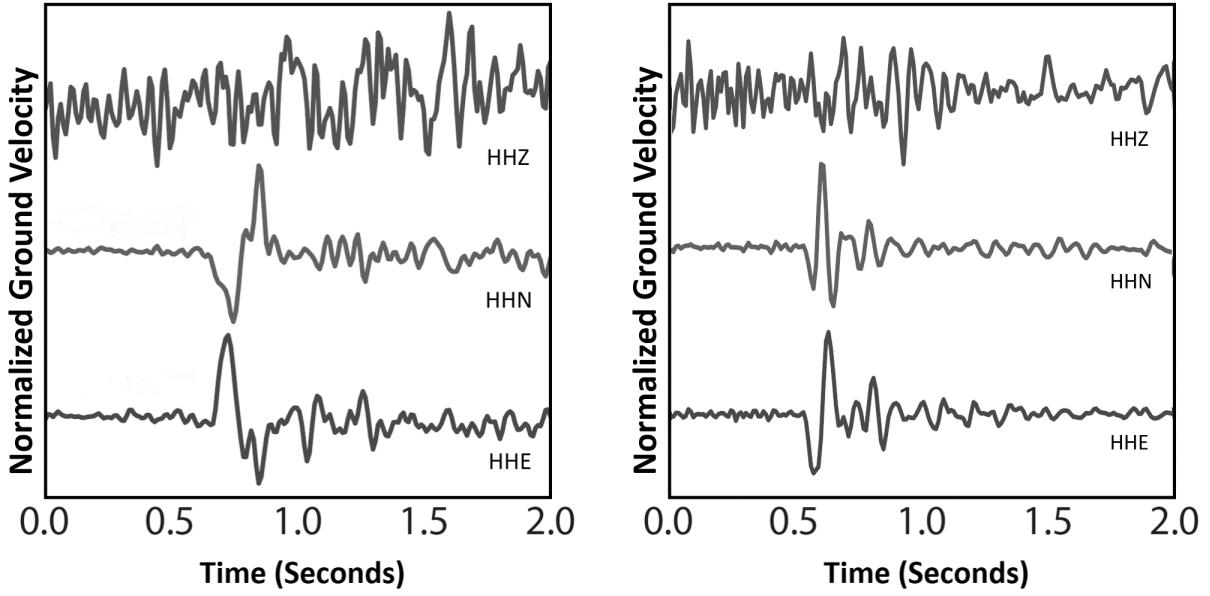


Figure 4.2: Templates used to generate synthetic data. Scaled copies of these two templates were inserted at random times over a Gaussian noise background. Figure on the left: template W_1 used to build both the synthetic training set and the first synthetic test set. Figure on the right: template W_2 used to create the second synthetic test set.

I chose two template waveforms W_1 and W_2 (figure 4.2), which have opposite polarities and different lengths to use as examples for generating similar and dissimilar recordings. Using the procedure described above, I generate a set of day-long records to evaluate capability of both methods to generalize and detect events not present in the training dataset.

For training the DuverNet we partition the continuous synthetic waveform data used for training into windows labeled as either seismic noise or earthquake. Training of the DuverNet on these two categories is done using the procedure previously described in the chapter three of this thesis. This allows us to test the detectability of the CNN method.

Synthetic data is generated to establish **ground truth** performance and test how both methods generalize to "similar" and "dissimilar" waveforms. There are two performance metrics that I

evaluate using synthetic data:

- **Detection accuracy for similar events.** Matched filtering performs well when the goal is to detect events with "similar" polarity and pulse direction as "parent" events. When one is looking for a better method to detect events it is important to ensure similar or better performance on the "similar" events.
- **Detection accuracy for the dissimilar events.** Matched filtering performs poorly when the goal is to detect events with "dissimilar" polarity and pulse direction as "parent" events. When one is looking for a better method to detect events it is important to ensure better performance on the "dissimilar" events.

4.2.2 Kaybob-Duvernay data

To compare two methods, I use the dataset previously used for the testing of the convolutional neural network architectures (see 3.2.1). It contains recordings from three-component seismic data using the following regional and local networks (figure 3.1): Alberta Geological Survey installed Regional Alberta Observatory for Earthquake Studies Network (RAVEN) and University of Calgary UC/DSA array. All of the stations were continuously recording the data during the period of the January 1 - June 30, 2016. Selected stations provide limited azimuthal coverage relative to the events used for training. The signals from all of the stations are converted to 100Hz on three channels: Z oriented vertically, N oriented North-South, and E oriented West-East. A catalog containing 450 events, which was provided by one of the operators in the area, was used to train the DuverNet. After data augmentation for adding extra positive examples (events) I end up with $N = 1,543,511$ windows available for training. $N_+ = 2594$ are positive examples and $N_- = 1,540,917$ negative examples. The reason why I keep high amount of negative examples for the training of the DuverNet is to ensure that the model does not overfit the data, which can happen if the size of the training dataset decrease. I used highest 139 SNR events with SNR above 1.6 as "parent" events for the MFA analysis.

Kaybob-Duvernay data is used to compare the performance on a real-life data and to populate sparse local catalog. Three performance metrics used for evaluating the methods using the Kaybob-Duvernay data are:

- **Runtime** is a crucial measurement. Lower runtime would allow for faster and more efficient processing of the datasets. Moreover, it would allow to process data with less expensive hardware and thus allow for easier real time deployment.
- **Memory used**, similarly to the **runtime**, is important from a hardware optimization standpoint. Using less memory would allow for processing more data before running into lack of memory issues.
- **Amount of the new events detected** in the available dataset. The ultimate goal of using a new method is to detect previously undetectable events and generate more complete catalogs.

4.3 Results and Discussion

4.3.1 Synthetic data

In a synthetic dataset experiment I show how both methods generalize to similar and dissimilar signals in the dataset. First, both DuverNet and MFA methods detect most of the events inserted using template W_1 ("similar") seen during training (see fig 4.3). The number of missed detections is low for all the records with SNR between -1 dB and 5 dB. Figure illustrates that DuverNet manages to detect more events at the same SNR relative to the MFA method.

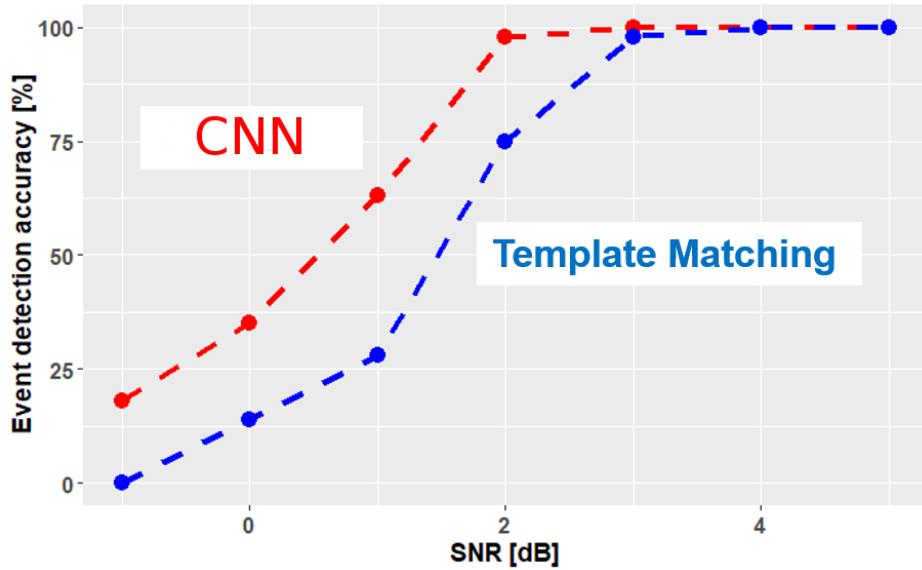


Figure 4.3: Detection accuracy between DuverNet (CNN based) and MFA for the similar events.

Second, DuverNet performs significantly better and detects more events in the case of "dissimilar" event detection (template W_2). Figure 4.4 illustrates that MFA method fails to detect the inserted events (which is expected). However, the DuverNet method recognizes the new (previously unknown) events and the accuracy of the CNN method tend to increase with SNR of the signal. For SNRs higher than 8dB, DuverNet detects all of the inserted seismic events.

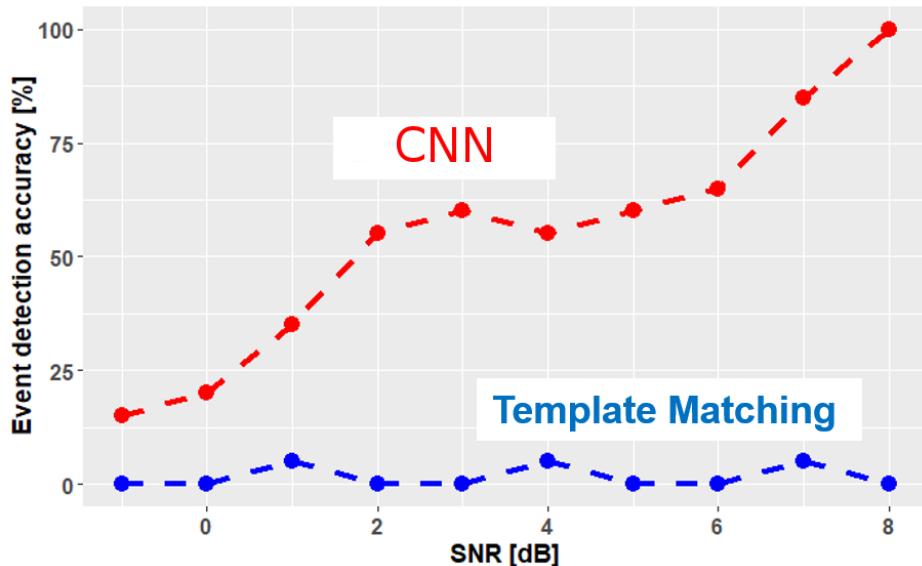


Figure 4.4: Detection accuracy between DuverNet (CNN based) and MFA for the dissimilar events.

Many events in our dataset from Kaybob-Duvernay are non-repeating events. This experiment on synthetic data suggests that methods relying on matched-filtering cannot detect them while DuverNet can.

4.3.2 Kaybob-Duvernay data

DuverNet is highly scalable and can easily handle large datasets.

	Match Filtering	Convolutional Neural Net
Noise detection accuracy	100.00%	99.80%
Event detection accuracy	98.00%	99.00%
Processing time	~ 4hr 30min	1min 34 sec

Table 4.1: Performance of the Convolutional Neural Network vs matched filtering method. Processing time here is referred to processing one month of continuous waveform data.

Table 4.1 illustrates the runtimes for MFA and DuverNet methods necessary to analyze one month of continuous waveform data from one station. Reported runtime excludes the training time for the neural network, which took 4.5 hours on 4-core Intel I7 processor laptop. Training need to be performed only once and the CNN does not require often retraining. Excluding the training time DuverNet is approximately 180 times faster than MFA method. The runtimes for long time series also indicate that DuverNet present an almost linear scaling between runtime and duration of time series to analyze. For the 1-month-long continuous time series, DuverNet runtime is 1 minute and 34 seconds, while that of matched-filtering is 4 hours and 30 minutes (see fig 4.5).

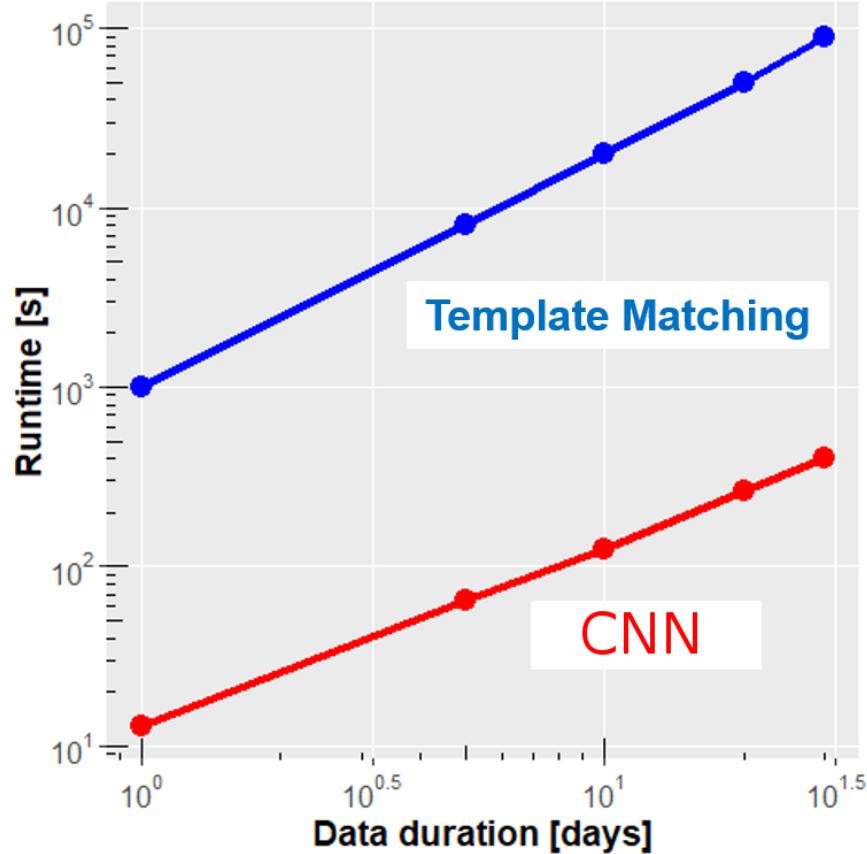


Figure 4.5: Runtime of the two methods: DuverNet (CNN based) and MFA.

Like other template matching techniques MFA database grows as it detects and stores more events during detection ("child" events). If we want to later use "child" detection as "parent" events later it would require more memory per station. For 5 days of the continuous recording, template matching requires approximately 400 MB of memory per station (see fig 4.6). Methodologies that require growing databases of templates eventually see their performance decreasing with size. The proposed DuverNet approach only needs to store a compact set of neural network parameters, which implies a use of relatively constant, small amount of memory (2MB).

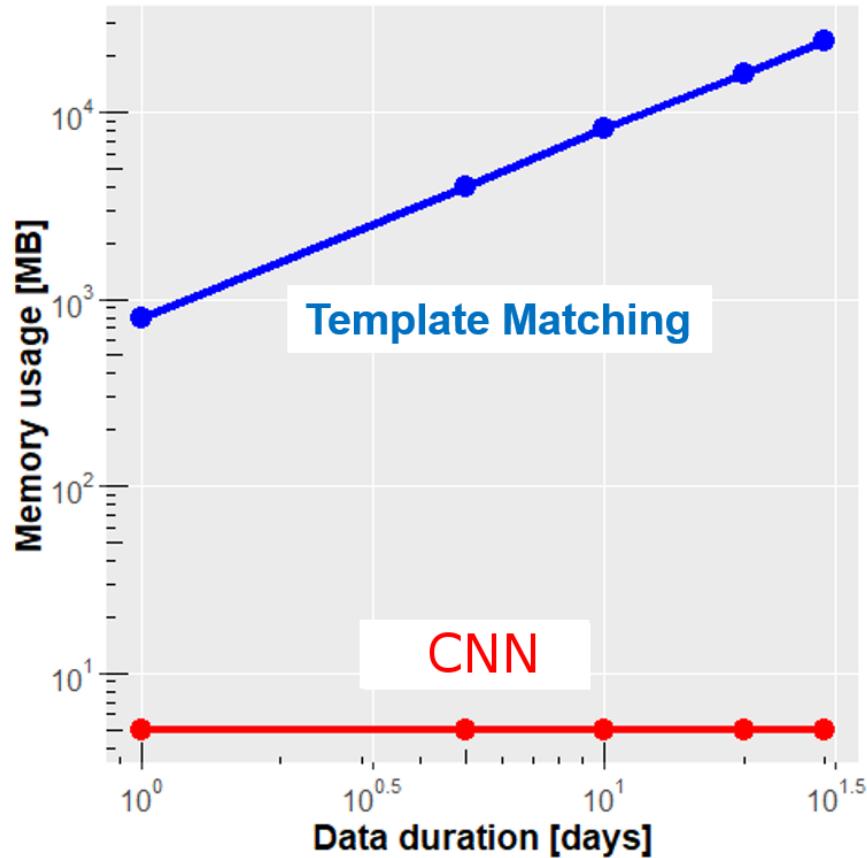


Figure 4.6: Memory use of the two methods: DuverNet (CNN based) and MFA.

Finally, I evaluate the detection power of the DuverNet method when compared to MFA method on Kaybob-Duvernay dataset. Between the months of January and June 2016, the DuverNet based method detected 605 new events, while matched-filtering detected 353 previously unseen events. In addition, DuverNet detected 252 additional events that were not previously detected using matched filtering or other methods. Example of the event that was detected by both DuverNet and MFA can be seen on figure 4.7.

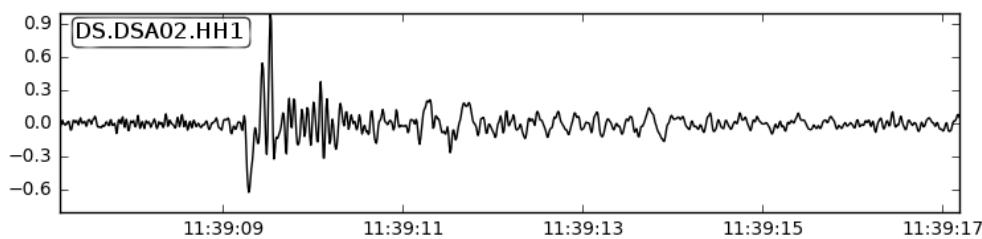


Figure 4.7: Seismogram of an event that was detected by DuverNet and MFA detector.

This event has high SNR and high cross-correlation coefficient with one of the MFA template "parent" events which led to detection. On the other hand, event shown on figure 4.8 was detected by DuverNet and missed by MFA.

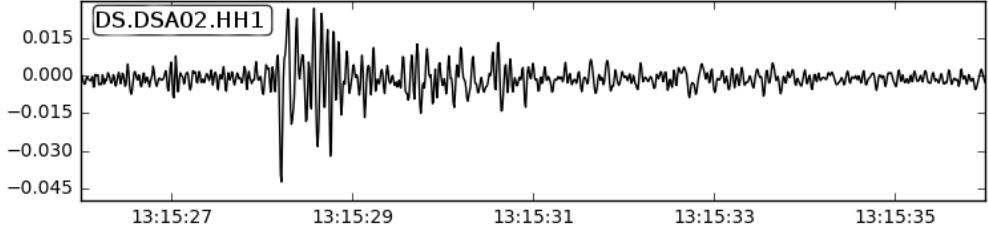


Figure 4.8: Seismogram of an event that was detected by DuverNet, but was missed by MFA detector.

Potential explanation can be a low SNR of the event as well as relatively unique source mechanism, which was not represented in the MFA "parent" events database.

4.4 Conclusions

In this chapter I compared the performance of optimized MFA and CNN based (DuverNet) methods for the detection of induced seismicity applied to synthetic and Kaybob-Duvenray datasets. After allowing time for appropriate training, CNN performs best in computational runtime, memory use, and superior detection performance compared to the MFA. Synthetic data test illustrated that DuverNet method better generalizes to previously unseen events and is better at detecting more events at lower SNR compared to the MFA method. This helps with mitigation of the representation power problem, which would allow for easy scaling of this approach to more broadband networks as more data becomes available in the future. Between months of January and June 2016, DuverNet based method detected 252 additional events that were not found using matched-filtering or other methods before.

Chapter 5

Conclusions and Future Work

5.1 Conclusions

In this thesis, two methods for the detection of the hydraulic fracturing induced seismic events are discussed: matched filtering method and convolutional neural network based method. Chapter one discusses the importance of the Duvernay Formation, gives a brief overview of the history of seismic event detection, lists some of the crucial challenges associated with automatic earthquake detection, and describes latest development in the area of application of machine learning in seismology.

Chapter two presents an optimization of matched filtering based detection method, later to be used for establishing baseline performance for the detection of induced seismicity in the Kaybob Duvernay production area of Alberta. Specifically, the algorithm was tailored to the available data by optimizing the length of the "parent" events, detection threshold type and value. Higher signal-to-noise of the S-waves compared to P-waves led to identify a 10-second window containing mostly S-wave arrival and S-coda as being optimal "parent" event length. An optimal detection threshold type and value was identified to be Median Absolute Deviation with the trigger of 6.0. Using the following parameters resulted in over 500 new unique "child" detections to add to the catalog for the time period of January 5-19, 2016. Furthermore, this chapter introduces a novel dataset collected by the 6 UC/DSA array stations installed by Nanometrics for the University of Calgary.

Chapter three is dedicated to the optimization of earthquake detection using convolutional neural networks. Two different architectures were proposed and tested: ConvNetQuake and

VGG-Junior. ConvNetQuake detects more events, but requires bigger memory, longer training time and computational time to run. VGG-Junior is faster to train and deploy at the cost of relatively lower detection performance. I also tested two different loss functions: cross-entropy loss and focal loss. My thesis is the first time focal loss is used for tackling class imbalance problem in earthquake detection. The highest performing model trained is based on ConvNetQuake architecture combined with Focal Loss and Adam optimizer. After 30,000 learning iterations on the data recorded between January 1 and June 30, 2016, chosen CNN architecture achieved accuracy of 98.09%, and once deployed takes 5.04ms to analyze one 10s waveform.

Chapter four presents a comparison between matched filtering (optimized in chapter two) and convolutional neural network (optimized in chapter three) based methods for the detection on induced seismicity. After allowing time for appropriate training, CNN performs best in computational runtime and memory use. Moreover it delivers superior detection performance compared to the MFA detector. A synthetic data test illustrate that CNN method better generalizes to previously unseen events and is better at detecting more events at lower SNR compared to the MFA method. This helps with mitigation of the representation power problem, which would allow for easy scaling of this approach to more broadband networks as more data becomes available in the future.

The importance of this thesis is that it introduces a novel way for the detection of induced seismicity in the Kaybob Duvernay production region of Alberta. This method can also be deployed in other area where human induced seismicity is present. Moreover I demonstrate the power of Focal Loss when dealing with class imbalance issue in earthquake detection problem. Finally, this thesis is the first time UC/DSA array stations data have been used for earthquake detection. This contribution is a step towards developing an automated earthquake detection system that requires minimal human input and learns with time.

5.2 Future work

There are several possible directions for the future research that are important for better understanding of the earthquake detection and subsequently achieving a higher quality catalogues with lower SNR for more precise fault interpretation accuracy.

In this thesis, I used raw seismograms from individual stations to feed into convolutional neural network. This method analyses full information from the data, but also risks to introduce unnecessary noise to the data, that can be avoided with some simple filtering. Alternatively, one can perform a spectral decomposition of the signal using Fast Fourier Transform (FFT) and then use 2D convolutional neural network to classify into earthquakes and noise. Moreover, band-pass filter can be applied to avoid frequencies which do not contribute to the earthquake detection accuracy.

Instead of inputting data from one station at a time, it might be interesting to investigate whether feeding in data from all of the stations in the array into the CNN would lead to more efficient detection. Instead of the input being 3 channels by 100Hz times 10s (3 by 1000) for example, we can use all six UC/DSA stations which would lead to the input of size 6 stations times 3 channels per station by 100Hz times 10s (18 by 1000).

Another potential area of improvement could be integration of matched filtering and CNN methods into one workflow that can be applied in the areas where historic catalogues are sparse and CNN based method would not have sufficient amount of training examples to train. First step would be to identify a couple high SNR events. Second step would be to run MFA using high SNR events as "parents". Finally, we would use newly detected events as positive examples used to train CNN classifier.

Transfer learning is a method where a model (neural network) developed for a task is reused as the starting point of training for a model on another task. It is an optimization technique that allows rapid progress or improved performance when modeling the second task (Goodfellow et al., 2016). In our case one might try to take trained DuvenNet weights and biases and use them as starting point (initialization) for training a new CNN for detection of earthquakes in other areas of induced seismicity in North American or around the world.

To conclude, convolutional neural networks have tendency to increase in complexity as they get bigger and get more layers. Convolutional neural network pruning can be used to decrease the size of the network and enable efficient inference (Molchanov et al., 2016). After pruning, CNN maintains good generalizations (weights) in the network and drops the inefficient ones, which leads to faster runtime and less used memory. One of the approaches I would recommend to start with would be a "try-and-learn" algorithm to train pruning agents that remove unnecessary CNN filters in a data-driven way (Huang et al., 2018).

Bibliography

- Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G.S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., Levenberg, J., Mane, D., Monga, R., Moore, S., Murray, D., Olah, C., Schuster, M., Shlens, J., Steiner, B., Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Vasudevan, V., Viegas, F., Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y., Zheng, X., 2015, "TensorFlow: Large-scale machine learning on heterogeneous systems": <https://arxiv.org/abs/1603.04467>.
- Aguiar, A. C., Chao, K., and Beroza, G. C. (2017). Tectonic tremor and LFEs on a reverse fault in Taiwan. *Geophysical Research Letters*, 44(13):6683-6691.
- Akram, J., Eaton, D. (2016). A review and appraisal of arrival-time picking methods for downhole microseismic data. *GEOPHYSICS*. 81. KS67-KS87. 10.1190/geo2014-0500.1.
- Alberta Energy Regulator (2016): Duvernay Reserves Report: A comprehensive analysis of Alberta's foremost liquids-rich shale resource; Alberta Energy Regulator, Calgary, Alberta, 74 p.
- Alberta Geological Survey / Alberta Energy Regulator (2013): Regional Alberta Observatory for Earthquake Studies Network. International Federation of Digital Seismograph Networks. Dataset/Seismic Network. 10.7914/SN/RV
- Allen, R. 1978, Automatic earthquake recognition and timing from single traces: *Bulletin of the Seismological Society of America*, 68,1521-1532.
- Allen, R. (1982). Automatic phase pickers: their present use and future prospects. *Bulletin of the Seismological Society of America*, 72(6B):S225-S242.
- Anant, K. S. and Dowla, F. U. (1997). Wavelet transform methods for phase identification in three-component seismograms. *Bulletin of the Seismological Society of America*, 87(6):1598-1612.
- Andoni, A. and Indyk, P. (2006). Near-optimal hashing algorithms for approximate nearest neighbor in high dimensions. In *IEEE Symposium on Foundations of Computer Science (FOCS06)*, pages 459-468. IEEE.
- Andrichuk, M. J., (1961). Stratigraphic Evidence for Tectonic and Current Control of Upper Devonian Reef Sedimentation, Duhamel Area, Alberta, Canada. *AAPG Bulletin*. 45. 10.1306/BC743681-16BE-11D7-8645000102C1865D.
- Atkinson, G.M., Eaton, D.W., Ghofrani, H., Walker, D., Cheadle, B., Schultz, R., Shcherbakov, R., Tiampo, K., Gu, J., Harrington, R.M., Liu, Y., Van Der Baan, M., Kao, H., 2016, Hydraulic fracturing drives induced seismicity in the Western Canada Sedimentary Basin, *Seismological Research Letters*, 87(3), 631-647.
- Baillard, C., Crawford, W.C., Ballu, V., Hibert, C., Mangeney, A., 2014. "An automatic Kurtosis based P- and S-phase picker designed for local seismic networks", *Bull. seism. Soc. Am.*, 104(1), 394-409.

- Baker, T., Granat, R., Clayton, R. (2005). Real-time Earthquake Location Using Kirchhoff Reconstruction. *Bulletin of the Seismological Society of America*. 95. 699-707. 10.1785/0120040123.
- Bao, X. and Eaton, D.W. (2016). Fault activation by hydraulic fracturing in western Canada. *Science*, 354: 1406-1409.
- Bardainne, T., Gaillet, P., Dubos-Salle, N., Blanco, J., and Snchal, G. (2006). Characterization of seismic waveforms and classification of seismic events using chirplet atomic decomposition. Example from the Lacq gas field (Western Pyrenees, France). *Geophysical Journal International*, 166(2):699-718.
- Beyreuther, M., Carniel, R., and Wassermann, J. (2008). Continuous hidden Markov models: application to automatic earthquake detection and classification at Las Canadas caldera, Tenerife. *Journal of Volcanology and Geothermal Research*, 176(4):513-518.
- Beyreuther, M., Barsch, R., Krischer, L., Megies, T., Behr Y. and Wassermann J. (2010), ObsPy: A Python Toolbox for Seismology, *SRL*, 81(3), 530-533.
- Beyreuther, M., Hammer, C., Wassermann, J., Ohrnberger, M., Megies, T. (2012). Constructing a Hidden Markov Model based earthquake detector: Application to induced seismicity. *Geophysical Journal International*. 189. 602-610. 10.1111/j.1365-246X.2012.05361.x.
- Beyreuther, M. and Wassermann, J. (2008). Continuous earthquake detection and classification using discrete hidden Markov models. *Geophysical Journal International*, 175(3):1055-1066.
- Bostock, G. M., Trehearne, A. (2012). Wave-Field Decomposition of Ocean Bottom Seismograms. *The Bulletin of the Seismological Society of America*. 102. 1681-1692. 10.1785/0120110162.
- British Columbia Oil and Gas Commission, 2012, Investigation of observed seismicity in the Horn River basin: Technical Report.
- Bromirski, P. D. (2001). Vibrations from the Perfect Storm. *Geochemistry, Geophysics, Geosystems*, 2(7).
- Brown, R. J., Beroza, C. G., Shelly, R. D. (2008). An autocorrelation method to detect low frequency earthquakes within tremor. *Geophysical Research Letters - GEOPHYS RES LETT*. 35. 10.1029/2008GL034560.
- Caffagni, E., Eaton, D., Jones, J., Van der Baan, M. (2016). Detection and analysis of microseismic events using a Matched Filtering Algorithm (MFA). *Geophysical Journal International*. 206. 10.1093/gji/ggw168.
- Maureen Caudill, "Neural Network Primer: Part I", *AI Expert*, 1989.
- Chamberlain, C. J., Hopp, C. J., Boese, C. M., Warren-Smith, E., Chambers, D., Chu, S. X., Michailos, K., Townend, J., 2017, "EQcorrscan: Repeating and near-repeating earthquake detection and analysis in Python": *Seismological Research Letters*.

Chen, X., Nakata, N., 2017. "Preface to the Focus Section on the 3 September 2016 Pawnee, Oklahoma, Earthquake". *Seismological Research Letters*. 88 (4): 953-955. doi:10.1785/0220170078. ISSN 0895-0695.

Cui, Z., Chen, W., Chen, Y.: "Multi-Scale Convolutional Neural Networks for Time Series Classification". ArXiv e-prints (2016)

Del Pezzo, E., Esposito, A., Giudicepietro, F., Marinaro, M., Martini, M., and Scarpetta, S. (2003). Discrimination of earthquakes and underwater explosions using neural networks. *Bulletin of the Seismological Society of America*, 93(1):215-223.

Diaz, J., Ruiz, M., Sanchez-Pastor, P. S., and Romero, P. (2017). Urban seismology: On the origin of earth vibrations within a city. *Scientific Reports*, 7(1):15296.

Dix, G.R. 1990. Stages of platform development in the Upper Devonian (Frasnian) Leduc Formation, Peace River Arch, Alberta. In: *Geology of the Peace River Arch*. S.C. O'Connell and J.S. Bell (eds.). *Bulletin of Canadian Petroleum Geology*, v. 38A, p. 66-92.

Dowla, F. U., Taylor, S. R., and Anderson, R. W. (1990). Seismic discrimination with artificial neural networks: preliminary results with regional spectral data. *Bulletin of the Seismological Society of America*, 80(5):1346-1373.

Dunn, L., Humenjuk, J. Taylor, N. and McHarg, D. (2014): Fifty shades freed: from pore to seismic and integrated workflow identifying sweet spots in the liquids rich Duvernay Formation Kaybob, Alberta, Canada; CSPG Technical Luncheon, Calgary, Alberta, 23 September, p. 1-46.

Draelos, T. J., Peterson, M. G., Knox, H. A., Lawry, B. J., Phillips-Alonge, K. E., Ziegler, A. E., Chael, E. P., Young, C. J., and Faust, A. (2018). Dynamic tuning of seismic signal detector trigger levels for local networks. *Bulletin of the Seismological Society of America*.

Dysart, P. S. and Pulli, J. J. (1990). Regional seismic event classification at the NORESS array: Seismological measurements and the use of trained neural networks. *Bulletin of the Seismological Society of America*, 80(6B):1910-1933.

Eaton, D.W. (2018). *Passive Seismic Monitoring of Induced Seismicity: Fundamental Principles and Application to Energy Technologies*. Cambridge University Press.

Eaton, D.W., Milkereit, B., Ross, G. M., Kanasewich, E. R., Geis, W., Edwards, D. J., Kelsch, L. and Varsek, J., 1995. Lithoprobe basin-scale seismic profiling in central Alberta: Influence of basement on the sedimentary cover. *Bulletin of Canadian Petroleum Geology*, 43(1), pp.65-77.

Eaton, D.W., Ross, G.M., and Clowes, R.M., 1999, "Seismic-reflection and potential-field studies of the Vulcan structure, western Canada: A paleoproterozoic Pyrenees?": *Journal of Geophysical Research*, v. 104, p.23,255-23,269.

Eaton, D. and Mahani, A. (2015). Focal Mechanisms of Some Inferred Induced Earthquakes in Alberta, Canada. *Seismol. Res. Lett.* 86: 1078-1085.

Ellsworth, W. L. (2013). Injection-induced earthquakes. *Science*, 341(6142).

- Esposto, A., Giudicepietro, F., DAuria, L., Scarpetta, S., Martini, M., Coltelli, M., and Marinaro, M. (2008). Unsupervised neural analysis of very-long-period events at Stromboli volcano using the self-organizing maps. *Bulletin of the Seismological Society of America*, 98(5):2449-2459.
- Esposto, A., Giudicepietro, F., Scarpetta, S., DAuria, L., Marinaro, M., and Martini, M. (2006). Automatic discrimination among landslide, explosion-quake, and microtremor seismic signals at Stromboli volcano using neural networks. *Bulletin of the Seismological Society of America*, 96(4A):1230-1240.
- Fomel, S. (2009): Adaptive multiple subtraction using regularized nonstationary regression; *Geophysics*, Vol. 74(1), V25-V33
- Frank, W. B., Shapiro, N. M. (2014). Automatic detection of low-frequency earthquakes (LFEs) based on a beamformed network response, *Geophysical Journal International*, 197(2), 1215-1223.
- Galiana-Merino, J. J., Rosa-Herranz, J., Parolai, S. (2008). Seismic *P* Phase Picking Using a Kurtosis-Based Criterion in the Stationary Wavelet Domain. *Geoscience and Remote Sensing, IEEE Transactions on*. 46. 3815-3826. 10.1109/TGRS.2008.2002647.
- Gentili, S., Michelini, A. (2006). Automatic picking of P and S phases using a neural tree. *Journal of Seismology*. 10. 39-63. 10.1007/s10950-006-2296-6.
- Gibbons, S., Ringdal, F. (2006). The detection of low magnitude seismic events using array-based waveform correlation. *Geophysical Journal International*. 165. 149-166. 10.1111/j.1365-246X.2006.02865.x.
- Grigoli, F., Cesca, S., Rinaldi, A. P., Manconi, A., Lopez-Comino, J. A., Clinton, J. F., Westaway, R., Cauzzi, C., Dahm, T., 2018, "The November 2017 Mw 5.5 Pohang earthquake: A possible case of induced seismicity in South Korea". *Science*. 360 (6392): 1003-1006. Bibcode:2018Sci...360.1003G. doi:10.1126/science.aat2010. ISSN 0036-8075. PMID 29700226.
- Goodfellow, I., Bengio, Y., Courville, A., "Deep Learning", MIT Press, 2016.
- Haq, B.U. and Shutter, S. R. (2008) A Chronology of Paleozoic Sea-Level Changes. *Science*, 322, 64-68. <http://dx.doi.org/10.1126/science.1161648>
- Harris, D. B. (2006). Subspace detectors: Theory. Technical Report UCRL-TR-222758.
- Harris, D. B. and Dodge, D. A. (2011). An autonomous system for grouping events in a developing aftershock sequence. *Bulletin of the Seismological Society of America*, 101(2):763.
- He, K., Zhang, X., Ren, S., Sun, J. (2016). In: Proceedings of the IEEE conference on computer vision and pattern recognition pp. 770-778.
- Healy, J., Rubey, W., Griggs, D., Raleigh, C. (1968), The Denver Earthquakes, *Science*, 161(3848), 1301-1310.

- Huang, Y., Beroza, C. G. (2015). Temporal Variation in the Magnitude-Frequency Distribution during the Guy-Greenbrier Earthquake Sequence. *Geophysical Research Letters*. 42. 10.1002/2015GL065170.
- Huang, G., Liu, Z., Van Der Maaten, L., Weinberger, K. Q. (2017). In: CVPR volume 1 p. 3.
- Qiangui H., Zhou S. K., You S., Neumann U., 2018. Learning to Prune Filters in Convolutional Neural Networks, CoRR, abs/1801.07365. <http://arxiv.org/abs/1801.07365>
- Hunter, I.G. and Wilcox, G. (2012): Duvernay Shale - an unconventional reservoir; CSPG/CSEG/CWLS Geoconvention 2012, May 14-18 Calgary, Alberta.
- Howarth, R. W., Ingraffea, A., Engelder, T., 2011, "Should fracking stop?", *Nature*, volume: 477, pages: 271-275.
- IEA (2018), World Energy Outlook 2018, IEA, Paris, <https://doi.org/10.1787/weo-2018-en>.
- Ioffe, S., Szegedy, C. (2015). arXiv preprint arXiv:1502.03167.
- Joswig, M. (1990). Pattern recognition for earthquake detection. *Bulletin of the Seismological Society of America*, 80(1):170-186.
- Jurkevics, A. (1988). Polarization analysis of three-component array data. *Bulletin of the Seismological Society of America*, 78(5):1725-1743.
- Kao, H., Eaton, D., Atkinson, G., Maxwell, S., Mahani, A. (2016). Technical Meeting on the Traffic Light Protocols (TLP) for Induced Seismicity: Summary and Recommendations. 10.4095/299002.
- King, G., 2012, "Hydraulic fracturing 101: what every representative, environmentalist, regulator, reporter, investor, university researcher, neighbor and engineer should know about estimating frac risk and Improving frac performance in unconventional gas and oil", Proceedings of the SPE Hydraulic Fracturing Technology Conference.
- Kingma, D., Ba, J., (2014). Adam: A Method for Stochastic Optimization. International Conference on Learning Representations.
- Kohler, A., Ohrnberger, M., Riggelsen, C., and Scherbaum, F. (2008). Unsupervised feature selection for pattern search in seismic time series. In *Journal of Machine Learning Research, Workshop and Conference Proceedings: New challenges for feature selection in data mining and knowledge discovery*, volume 4, pages 106-121.
- Kohonen, T. (2013). Essentials of the self-organizing map. *Neural networks*, 37:52-65.
- Krawczyk, B. (2016). Learning from imbalanced data: open challenges and future directions. *Progress in Artificial Intelligence*, 5(4):221-232.
- Krischer, L., Megies, T., Barsch, R., Beyreuther, M., Lecocq, T., Caudron, C., Wassermann, J. (2015), "ObsPy: a bridge for seismology into the scientific Python ecosystem", *Computational Science and Discovery*, 8(1), 014003.

Krizhevsky, A., Sutskever, I., Hinton, G. E. (2012). In: Advances in neural information processing systems pp. 1097-1105.

Kuperkoch, L., Meier, T., Lee, J., and Friederich, W. (2010). Automated determination of P-phase arrival times at regional and local distances using higher order statistics. Geophysical Journal International, 181(2):1159-1170.

Kvamme, H., Sellereite, N., Aas, K., Sjursen, S.: "Predicting mortgage default using convolutional neural networks". Expert Systems with Applications 102, 207-217 (2018).

LeCun, Y., Bengio, Y., and Hinton, G. (2015). Deep learning. Nature, 521(7553):436.

LeCun, Y., Jackel, L., Bottou, L., Cortes, C., Denker, J. S., Drucker, H., Guyon, I., Muller, U. A., Sackinger, E., Simard, P., et al. (1995). "Neural networks: the statistical mechanics perspective", 261, 276.

LeCun, Y., Bottou, L., Bengio, Y., Haffner, Patrick. (1998). Gradient-Based Learning Applied to Document Recognition. Proceedings of the IEEE. 86. 2278-2324. 10.1109/5.726791.

Li, Z., Peng, Z., Hollis, D., Zhu, L., and McClellan, J. (2018). High-resolution seismic event detection using local similarity for Large-N arrays. Scientific reports, 8(1):1646.

Lin, t., Goyal, P., Girshick, R., He, K. and Dollar, P. "Focal Loss for Dense Object Detection," 2017 IEEE International Conference on Computer Vision (ICCV), Venice, 2017, pp. 2999-3007.

Liu, Q., Wang, J. (2008). IEEE transactions on neural networks, 19 (4), 558-570.

Lomax, A., Satriano, C., Vassallo, M. (2012), Automatic picker developments and optimization: FilterPicker - a robust, broadband picker for real-time seismic monitoring and earthquake early-warning, Seism. Res. Lett. , 83, 531-540, doi: 10.1785/gssrl.83.3.531.

Madureira, G., Ruano, A., (2009). A neural network seismic detector. Acta Tech. Jaurinesis. 2. 159-170.

Masiello, S., Esposito, A. M., Scarpetta, S., Giudicepietro, F., Esposito, A., Marinaro, M., et al. (2006). Application of self organized maps and curvilinear component analysis to the discrimination of the Vesuvius seismic signals. In Workshop On Self-Organizing Maps (WSOM).

Montgomery, C. T., Smith, M. B., 2010, "Hydraulic fracturing - history of an enduring technology", Journal of Petroleum Technology, p: 26-41.

Molchanov, P., Tyree, S., Karras, T., Aila, T., Kautz, J., 2016, "Pruning Convolutional Neural Networks for Resource Efficient Transfer Learning", CoRR, volume:abs/1611.06440.

Mousavi, S. M., Horton, S. P., Langston, C., Samei, B. (2016). Seismic features and automatic discrimination of deep and shallow induced-microearthquakes using neural network and logistic regression. Geophysical Journal International. 207. ggw258. 10.1093/gji/ggw258.

Nair, V., Hinton, G. E. (2010). In: Proceedings of the 27th international conference on machine learning (ICML-10) pp. 807-814.,

National Research Council of the National Academics, 2012, Induced seismicity Potential in Energy Technologies: The National Academic Press, Washington, D.C.

Ng, A., "Feature selection, L 1 vs. L 2 regularization, and rotational invariance", in: Proceedings of the twenty-first international conference on Machine learning, 2004, ACM, p. 78.

Oprsal I., Eisner L. (2014), Cross-correlational objective tool to indicate induced seismicity, Geophysical Journal International, January 2014, 196(3), pp: 1536-1543.

Pacala, S. and Socolow, R., 2004, "Stabilization wedges: solving the climate problem for the next 50 years with current technologies": Science, volume 305, 968-972.

Perol, T., Gharbi, M., Denolle, M., 2018, "Convolutional neural network for earthquake detection and location": Science Advances, Vol. 4, no. 2, e1700578.

Poiaia, N., Satriano, C., Vilotte, J., Bernard, P., Obara, K., "Multiband array detection and location of seismic sources recorded by dense seismic networks", Geophysical Journal International, Volume 205, Issue 3, June, 2016, Pages 1548-1573.

Poupinet, G., Ellsworth, W., and Frechet, J. (1984). Monitoring velocity variations in the crust using earthquake doublets: An application to the Calaveras Fault, California. Journal of Geophysical Research: Solid Earth, 89(B7):5719-5731.

Provost, F. (2000). "Machine learning from imbalanced data sets' 101". In Proceedings of the AAAI 2000 workshop on imbalanced data.

Rabiner, L. R. (1989). A tutorial on hidden markov models and selected applications in speech recognition. Proceedings of the IEEE, 77(2):257-286.

Reynen, A. and Audet, P. (2017). Supervised machine learning on a network scale: application to seismic event classification and detection. Geophysical Journal International, 210(3):1394-1409.

Riggelsen, C., Ohrnberger, M. (2012). A Machine Learning Approach for Improving the Detection Capabilities at 3C Seismic Stations. Pure and Applied Geophysics. 171. 10.1007/s00024-012-0592-3.

Riggelsen, C., Ohrnberger, M., and Scherbaum, F. (2007). Dynamic Bayesian networks for real-time classification of seismic signals. Knowledge Discovery in Databases: PKDD 2007, pages 565-572.

Rodriguez-Pradilla, G., Eaton, D., 2018. "The Application of the Signal Duration Method for Magnitude Estimation of Microseismic Events", CSPG/CSEG/CWLS Geoconvention 2018, May 7-11 Calgary, Alberta.

Rokosh, C. D., Lyster, S., Anderson, S. D. A., Beaton, A. P., Berhane, H., Brazzoni, T., Chen, D., Cheng, Y., Mack, T., Pana, C., Pawlowicz, J. G., 2012, "Summary of Alberta's shale- and siltstone-hosted hydrocarbon resource potential", Energy Resources Conservation Board, ER-CB/AGS Open File Report, volume: 2012-06, p:327.

- Rosasco, L., De Vito, E. D., Caponnetto, A., Piana, M., Verri, A. (2004), "Are Loss Functions All the Same?", *Neural Computation*. 16 (5): 1063-1076.
- Ross, Z. E., Meier, M. A., and Hauksson, E. (2018). P-wave arrival picking and first-motion polarity determination with deep learning. arXiv preprint arXiv:1804.08804.
- Ruppert, D. (2010). *Statistics and Data Analysis for Financial Engineering*. Springer. p. 118. ISBN 9781441977878.
- Saboy, L., Mountjoy, E., (1995). "Cratonic-margin and Antler-age foreland basin strata (Middle Devonian to Lower Carboniferous) of the southern Canadian Rocky Mountains and adjacent plains". 10.2110/pec.95.52.0213.
- Saragiotis, C., Hadjileontiadis, L., Panas, S. (2002). PAI-S/K: A robust automatic seismic P phase arrival identification scheme. *Geoscience and Remote Sensing, IEEE Transactions on*. 40. 1395-1404. 10.1109/TGRS.2002.800438.
- Scarpetta, S., Giudicepietro, F., Ezin, E., Petrosino, S., Del Pezzo, E., Martini, M., and Marinaro, M. (2005). Automatic classification of seismic signals at Mt. Vesuvius volcano, Italy, using neural networks. *Bulletin of the Seismological Society of America*, 95(1):185-196.
- Schaff, D., Waldhauser, F. (2010). One Magnitude Unit Reduction in Detection Threshold by Cross Correlation Applied to Parkfield (California) and China Seismicity. *Bulletin of The Seismological Society of America - BULL SEISMOL SOC AMER*. 100. 10.1785/0120100042.
- Schorlemmer, D. and Woessner, J. (2008). Probability of detecting an earthquake. *Bulletin of the Seismological Society of America*, 98(5):2103-2117.
- Schultz, B. F. (1999). "Gravitational wave astronomy". *Classical and Quantum Gravity*. 16 (12A): A131-A156.
- Schultz, R., Stern, V., Gu, Y.J., 2014, An investigation of seismicity clustered near the Cordel Field, west central Alberta, and its relation to a nearby disposal well: *Journal of Geophysical Research: Solid Earth* 119, 3410-3423.
- Schultz, R., Stern, V., Novakovic, M., Atkinson, G., and Gu, Y., (2015a), Hydraulic fracturing and the Crooked Lake sequence: Insights gleaned from regional seismic networks, *Geophys. Res. Lett.*, 42, 2750-2758.
- Schultz, R., Stern, V., 2015, "The Regional Alberta Observatory for Earthquake Studies Network (RAVEN)", *CSEG Recorder*, Vol: 40, No:08.
- Schultz, R., Mei, S., Pana, D., Stern, V., Gu, Yu., Kim, A., and Eaton, D. (2015b), The Cardston earthquake swarms and hydraulic fracturing of the Exshaw Formation (Alberta Bakken play), *Bull. Seismol. Soc. Am.*, 105(6).
- Schweitzer, J. *Pure appl. geophys.* (2001) 158:277.
- Sharma, M, Tyagi, A. (2010). Cyclic behavior of seismogenic sources in India and use of ANN for its prediction. *Natural Hazards*. 55. 389-404. 10.1007/s11069-010-9536-3.

- Shelly, D. R., Beroza, G. C., Ide S. (2007). Non-volcanic tremor and low-frequency earthquake swarms, *Nature* 446, no. 7133, 305.
- Simonyan, K., Zisserman, A. (2014). arXiv preprint arXiv:1409.1556.
- Skoumal, R. J., Brudzinski, M. R., and Currie, B. S. (2016). An efficient repeating signal detector to investigate earthquake swarms. *Journal of Geophysical Research: Solid Earth*, 121(8):5880-5897.
- Skoumal, R. J., Brudzinski, M. R., Currie, B. S., Levy, J. (2014). Optimizing multi-station earthquake template matching through re-examination of the Youngstown, Ohio, sequence. *Earth and Planetary Science Letters*, 405:274-280.
- Slinkard, M. E., Carr, D. B., and Young, C. J. (2013). Applying waveform correlation to three aftershock sequences. *Bulletin of the Seismological Society of America*, 103(2A):675.
- Smith, M. and Montgomery, C., 2015, "Hydraulic Fracturing". 1st edition, CRC Press.
- Switzer, S.B., Holland, W.G., Graf, G.C., Hedinger, A.S., McAuley, R.J., Wierzbicki, R.A. and Packard, J.J. (1994): Devonian Woodbend-Winterburn strata of the Western Canada Sedimentary Basin; in Geological atlas of the Western Canada Sedimentary Basin, Canadian Society of Petroleum Geologists and Alberta Research Council, p. 165-202.
- Szegedy, C., Ioffe, S., Vanhoucke, V., Alemi, A. A. (2017). In: AAAI volume 4 p. 12.
- Turin, G. L. (1960). "An introduction to matched filters". *IRE Transactions on Information Theory*. 6 (3): 311-329.
- Valentine, A. P. and Trampert, J. (2012). Data space reduction, quality assessment and searching of seismograms: autoencoder networks for waveform data. *Geophysical Journal International*, 189(2):1183-1202.
- Vallejos, J., McKinnon, S. D. (2013). Logistic regression and neural network classification of seismic records. *International Journal of Rock Mechanics and Mining Sciences*. 62. 86-95. 10.1016/j.ijrmms.2013.04.005.
- Vidale, J. E. (2011). Seattle "12th Man earthquake" goes viral. *Seismological Research Letters*, 82(3):449.
- Visser, R., Smith, B., Kao, H., Babaie Mahani, A., Hutchinson, J., McKay, J.E., 2017. A comprehensive earthquake catalogue for northeastern British Columbia and western Alberta, 2014-2016; Geological Survey of Canada, Open File 8335, 1 .zip file. <https://doi.org/10.4095/306292>
- Wang, R., Gu, Y. J., Schultz, R., Kim, A., Atkinson, G., 2016, "Source analysis of a potential hydraulic-fracturing-induced earthquake near Fox Creek, Alberta", *Geophysical Research Letters*, volume:83, p:564-573.
- Wang, R., Jeffrey, G., Schultz, R., Zhang, M., Kim, A., 2017, Source characteristics and geological implications of the January 2016 induced earthquake swarm near Crooked Lake, Alberta, *Geophysical Journal International*, Volume 210, Issue 2, 1 August 2017, Pages 979-988.

- Weichert, D., Horner, R. B., and Evans, S. G. (1994). Seismic signatures of landslides: The 1990 Brenda Mine collapse and the 1965 Hope rockslides. *Bulletin of the Seismological Society of America*, 84(5):1523.
- Wiemer, S. and Baer, M. (2000). Mapping and removing quarry blast events from seismicity catalogs. *Bulletin of the Seismological Society of America*, 90(2):525.
- Wiszniewski, J., Plesiewicz, B. M., and Trojanowski, J. (2014). Application of real time recurrent neural network for detection of small natural earthquakes in Poland. *Acta Geophysica*, 62(3):469-485.
- Wu, Y., Lin, Y., Zhou, Z., Bolton, D. C., Liu, J., and Johnson, P. (2017). Cascaded contextual region-based convolutional neural network for event detection from time series signals: A seismic application. arXiv preprint arXiv: 1709.07943.
- Wu, Y. Lin, Z. Zhou, D. C. Bolton, J. Liu and P. Johnson, "DeepDetect: A Cascaded Region-Based Densely Connected Network for Seismic Event Detection" in IEEE Transactions on Geoscience and Remote Sensing, vol. 57, no. 1, pp. 62-75, Jan. 2019.
- Yoon, C. E., O'Reilly, O., Bergen, K. J., Beroza, G. C., 2015. "Earthquake detection through computationally efficient similarity search", *Science advances* 1, e1501057.
- Yost, W. (1985). Fundamentals of Hearing: An Introduction (Second ed.). Holt, Rinehart and Winston. p. 206. ISBN 978-0-12-772690-8.
- Zhao, Y. and Takano, K. (1999). An artificial neural network approach for broadband seismic phase picking. *Bulletin of the Seismological Society of America*, 89(3):670-680.
- Zhang, M. and Wen, L. (2015). An effective method for small event detection: Match and Locate. *Geophysical Journal International*, 200(3):1523-1537.
- Zhu, W. and Beroza, G. C. (2018). PhaseNet: A deep-neural-network-based seismic arrival time picking method. arXiv preprint arXiv:1803.03211.
- Ziegler, M. A. (1967). A study of the Lower Cretaceous facies developments in the Helvetic Border Chain, north of the lake of Thun. *Edeg. geol. Hele*, 60 (2), 509-527.

Appendix A

UC/DSA array instrumentation

UC/DSA array stations are based on Trillium broadband seismometer manufactured and installed by Nanometrics Inc. Trillium sensors require ultra-low power, which allows for smaller power systems and higher station reliability.

Technology

Technology: Symmetric triaxial

Feedback: Force balance with capacitive transducer

Mass Centering: Not required

Seismometer Module Performance

Self-noise: See self-noise graph of figure A.1

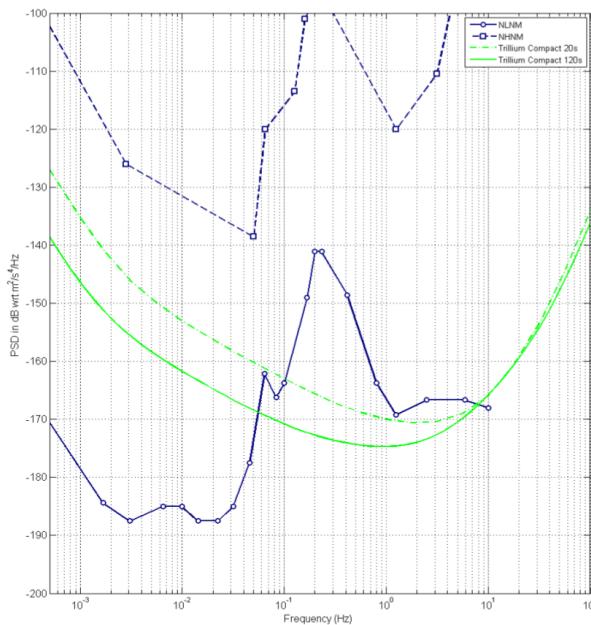


Figure A.1: Self-noise graph for Trillium broadband seismometer. Modified from https://www.nanometrics.ca/sites/default/files/2019-03/trillium_compact_datasheet.pdf.

Nominal Sensitivity: 750 V-s/m

Precision: 0.5% relative to User Guide specification

Bandwidth/120s: -3 dB points at 120s and 108 Hz

Bandwidth/20s: -3 dB points at 20s and 108 Hz

Off-axis Sensitivity: 0.5%

Clip level: 26 mm/s up to 10 Hz and 0.17g above 10 Hz

Operational Tilt Range/120s: 2.5°

Operational Tilt Range/20s: 10°

Parasitic Resonance: None below 200 Hz

Dynamic Range: >152 dB @ 1 Hz

Leveling and Alignment

Leveling: Adjust locking feet

Physical Bubble level: Included

Digital Bubble level: Graphical bullseye level is available via Centaur digital recorder GUI

Alignment: Vertical scribe marks for (N and S); precision guide in cover for straight-edge, line , or laser level

Interface

Connector: 14-pin, shell size 12, MIL-C-26482 Series I, top mounted

Velocity Output: 40V peak-to-peak differential

- Selectable XYZ to UVW mode

Mass Position Output: Single 4V output representing maximum mass position

- 3-channel mass positions available through serial port

Calibration Input: Single voltage input and one active high control signal to enable all 3 channels

- Remote calibration in XYZ or UVW mode

- Independent channel selection by serial port

Control Input: Cal. Enable or Long/Short Period mode, XYZ/UVW mode

Serial Port: RS-232 compatible serial IP (SLIP)

- Onboard web server standard HTTP
- For enhanced instrument control and status: UVW/XYZ mode, short/long period mode, firmware updates, temperature, mass position, case tilt, digital bubble level, serial number and factory info

Power

Supply Voltage: 9 to 36 VDC isolated input

Power Consumption:

- 180mW typical (model TC120-SV1)
- 195mW typical (model TC20-SV1)

Protection: Reverse-voltage and over-voltage protected

- Self-resetting over-current protection

Physical

Diameter: 90mm

Height:

- body and connector: 113mm
- with leveling feet fully retracted: 128mm
- with leveling feet fully extended: 135mm

Weight: 1.2kg

Housing: Resistant to corrosion scratches & chips

Environment

Operating temperature: -20°C to 60°C

Storage temperature: -40°C to 70°C

Shock:

- 100g half sine, 5ms without damage, 6 axes
- No mass lock required for transport

Magnetic: Insensitive to natural variations of the Earth's magnetic field

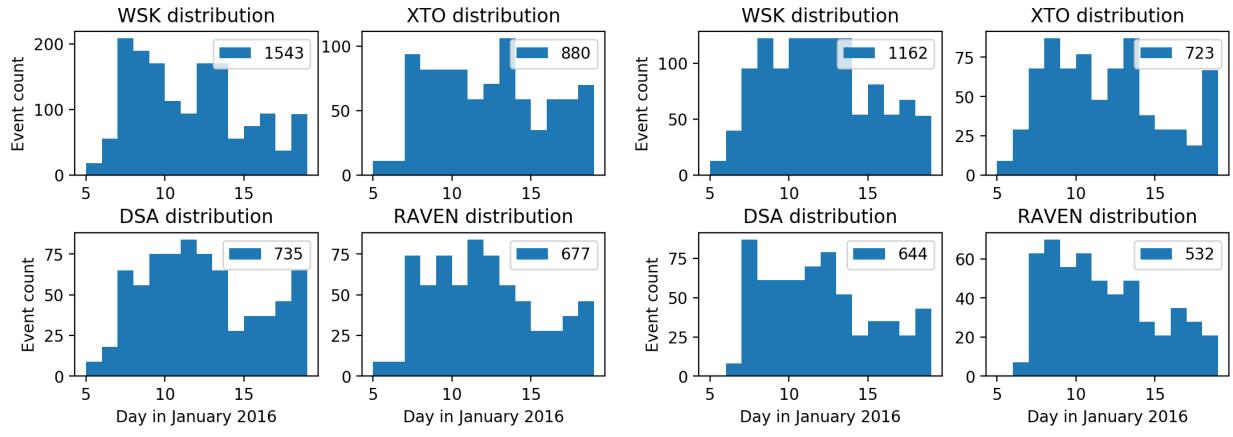
Weather Resistance: Rated for IP67

To learn more about the Trillium sensors please visit:
<https://www.nanometrics.ca/products/seismometers/trillium-compact>.

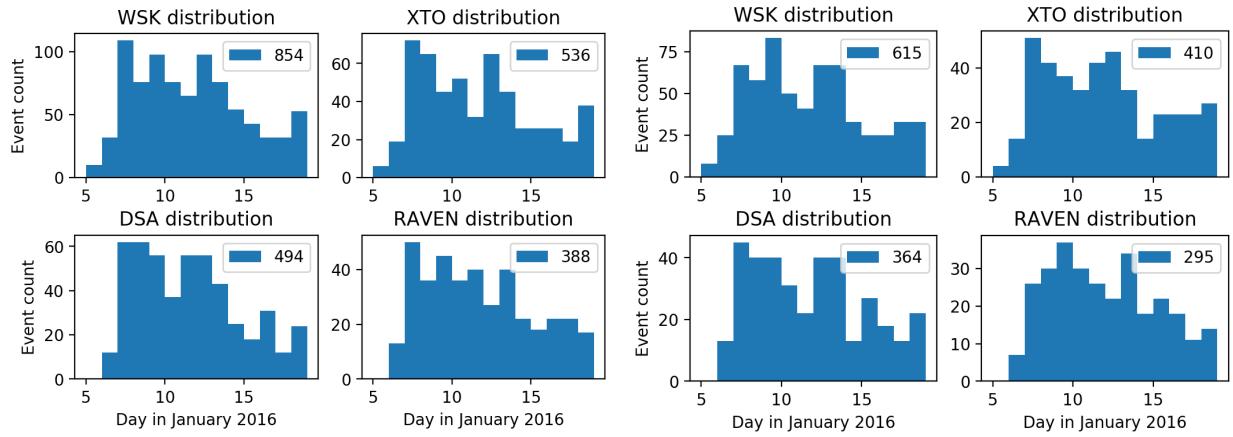
Appendix B

Distribution of the detection for each of the available seismic station networks with varying threshold values for the period of January 5-19, 2016.

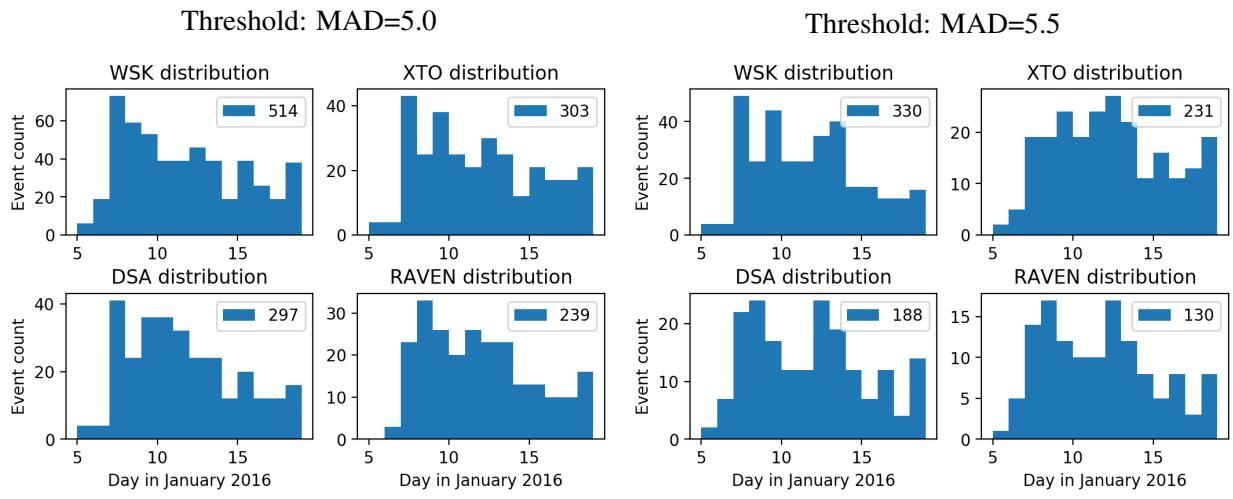
The following graphs show the daily distribution of the detections based on the MAD threshold values for different local arrays.



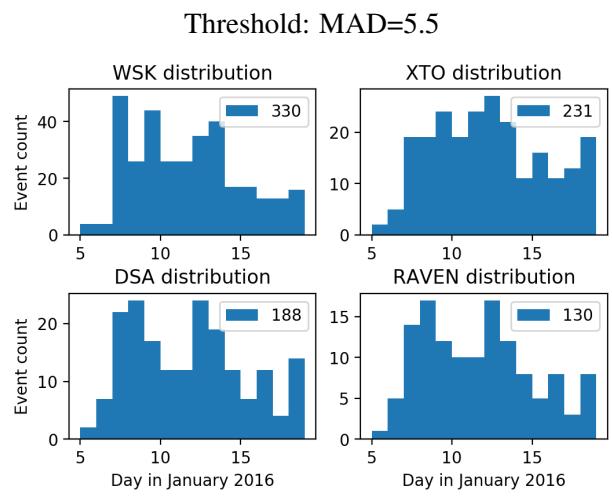
Threshold: MAD=4.0



Threshold: MAD=4.5



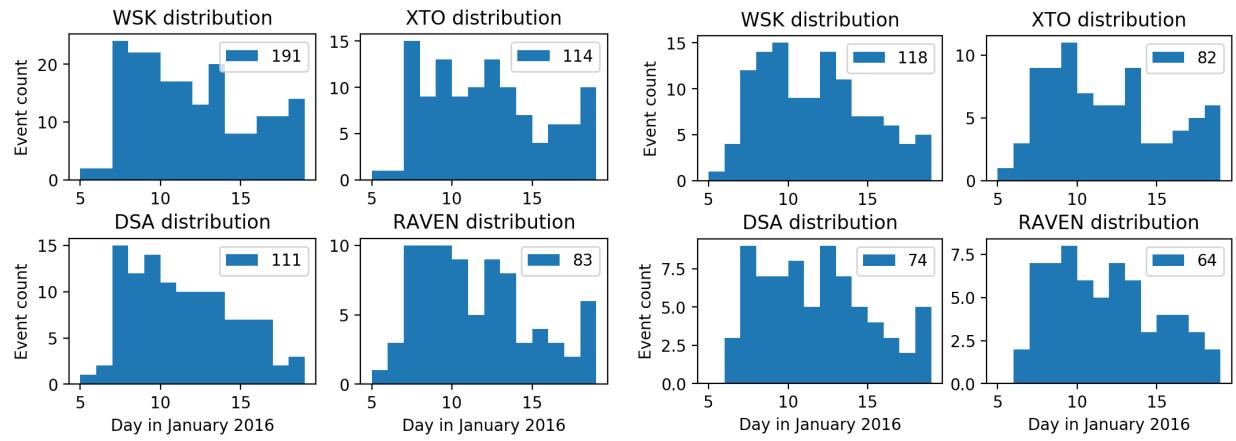
Threshold: MAD=5.0



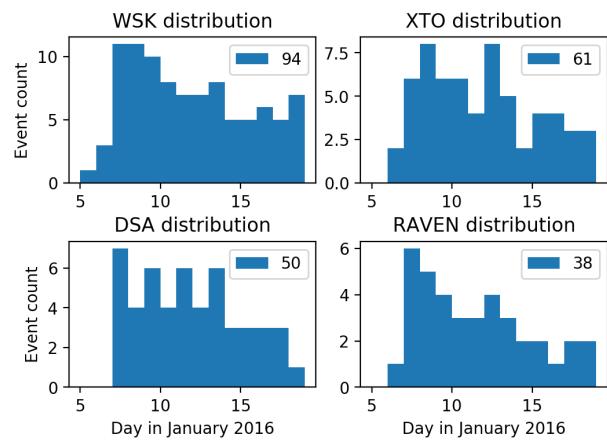
Threshold: MAD=5.5

Threshold: MAD=6.0

Threshold: MAD=6.5



Threshold: MAD=7.0



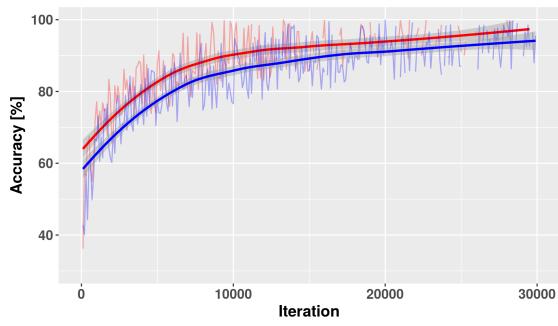
Threshold: MAD=7.5

Threshold: MAD=8.0

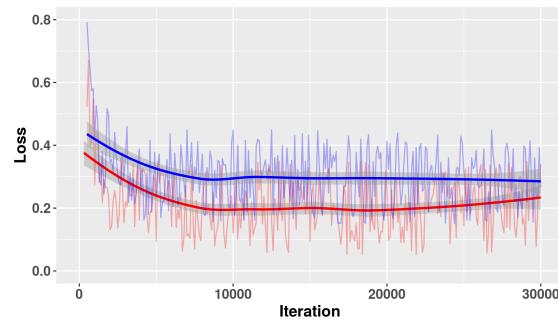
Appendix C

Accuracy and Loss Plots for the ConvNetQuake and VGG-Junior Architectures with Cross-Entropy and Focal losses

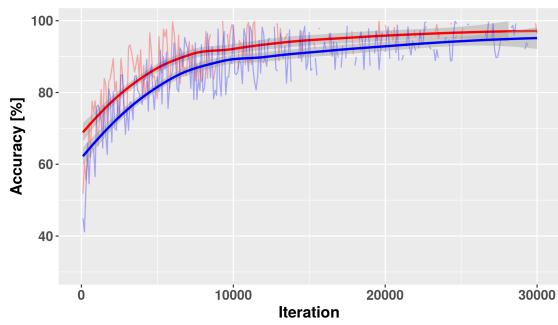
The following graphs show the Accuracy and Loss for training and validation sets for the ConvNetQuake and VGG-Junior architectures with Cross-Entropy loss and Focal loss to control for overfitting.



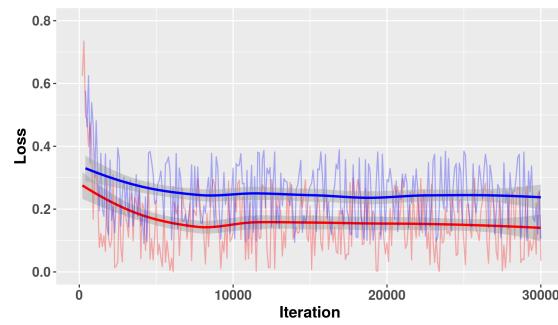
(a) Training (red) and Validation (blue) accuracy vs iteration for ConvNetQuake CE model.



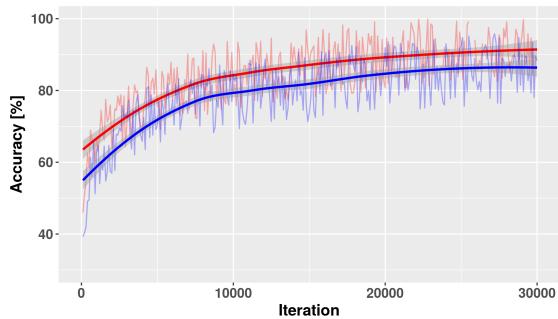
(b) Training (red) and Validation (blue) loss vs iteration for ConvNetQuake CE model.



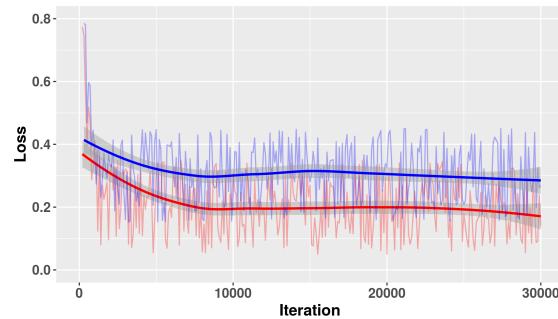
(a) Training (red) and Validation (blue) accuracy vs iteration for ConvNetQuake FL model.



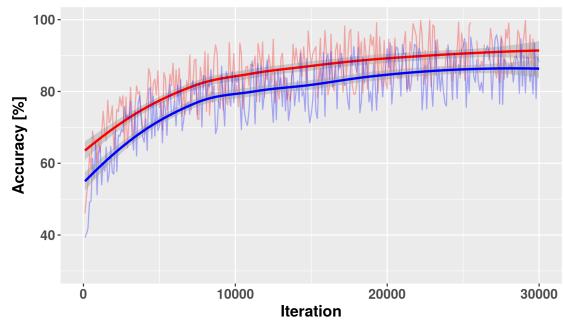
(b) Training (red) and Validation (blue) loss vs iteration for ConvNetQuake FL model.



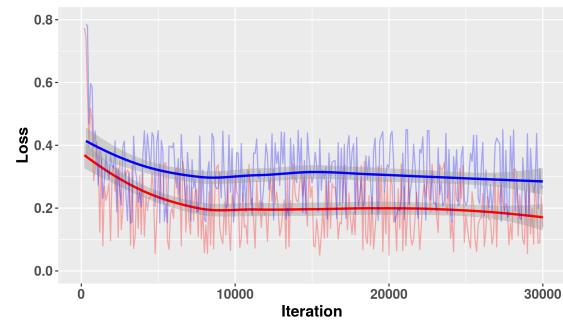
(a) Training (red) and Validation (blue) accuracy vs iteration for VGG-Junior CE model.



(b) Training (red) and Validation (blue) loss vs iteration for VGG-Junior CE model.



(a) Training (red) and Validation (blue) accuracy vs iteration for VGG-Junior FL model.



(b) Training (red) and Validation (blue) loss vs iteration for VGG-Junior FL model.