Seismic Wave Picking with Transformer Networks

B.Sc. Thesis Proposal

Thomas Bornstein

February 27, 2020

Contents

1 Introduction **1**

2 Background & Related Work **3**

3 Adaption & Progression **4**

4 Data **6**

5 Evaluation **6**

6 Workflow **7**

1 Introduction

In order to optimally monitor seismogenic processes, seismologist would like to detect as many earthquakes as possible. However, the vast majority of earthquakes are very small and disappear in a wide range of noise. The precise measure of the arrival times of seismic waves at a certain station is an important key for a range of tasks, e.g. determination of earthquake locations, focal depth, or seismic tomography. An earthquake has a hypocenter in a certain depth from where it radiates two types of waves in all directions:

1. P-waves. Longitudinal, compression waves, hitting the earth's surface at first.

2. S-waves. Transversal, shear waves, slower.

Seismic waves interact with the material properties of the subsurface they travel through, e.g. they are refracted and reflected, such that there are many different kinds of P- and S-waves observable at a seismic station. Also, their speed depends on material properties, which altogether can lead to superposition of multiple wave phases. Still, P-waves are the first to arrive on a seismometer's record (seismogram). Historically, seismologists studied seismograms and estimated the time of P-waves arrivals by hand. This determination of the onset of a wave is referred to as "wave picking". Although the sudden change in amplitude that often characterizes a seismic signal is easy to see, it is much more difficult to pick a wave when its absolute amplitude is below that of the noise [3]. This applies even more to the initially mentioned majority of small events where less seismic energy is radiated leading to relatively smaller amplitudes on seismograms.

The difficulty of wave picking and the problem of processing large data volumes boosted the research on automatic picking algorithms. Particularly, seismic record data increased exponentially over the decades. While the first approaches on automatic picking delivered less accurate results than human analysts, especially in the recent years, machine learning algorithms provided fast and very precise solutions. Newer approaches applied deep learning neural networks like convolutional (CNN) [6] and recurrent networks (RNN) [11] in order to avoid explicit wavelet templates and to improve the sensitivity for small and weak events as well as for very large magnitude events.

In this thesis, we apply a machine learning technique, the Transformer, which relies on a sequence-to-sequence network using a concept called "attention" [8]. It was originally invented for natural language processing (NLP). We employ this model in our sense as follows.

A seismogram is a waveform signal that can be discretized into a time series of amplitudes of the signal. Given a piece of seismic event record we define windows of a certain size (e.g. 2 seconds) and feed them to a CNN. The output vectors of the CNNs are, to use the analogy of translation, interpreted as words forming a sentence altogether. This sentence will be the input sequence to the Transformer. The output will be the probability of a wave arrival over time. As we often have a rough estimate of the event

location, we can give this as an additional input to the network, probably given as approximate distance to the epicenter of the event. This hopefully helps to increase the accuracy of the pick.

The key concept of the original Transformer model is a technique called "self-attention", which helps to avoid the use of RNN (and even CNN according to the original setup) overall leading to much faster learning times [8].

The aim of this thesis is to evaluate the application of the Transformer approach on phase picking. Our first goal is to try P-wave picking. When successful, we will try out S-wave picking, as well. We hope to find a more accurate and general model than those relying on CNN and RNN.

2 Background & Related Work

Early machine learning methods to detect earthquake signals were based on fully connected neural networks (FCNN) [9, 10]. In general, neural networks (NN) take features as input to a non-linear mapping function with up to several million parameters (for a deep NN) mapping them to a continuous variable or a class prediction as output. Those parameters of the mapping function are organized in sequential layers of neurons which process the incoming data and pass the results on to the next layer. The parameters are empirically optimized with large amounts of training data.

The quality of automatic picking algorithms improved with the introduction of CNN, a variant of NNs where the data is fed to a set of locally connected convolution and pooling layers, each of which consists of learnable filters that are supposed to identify features within a data subset. ConvNetQuake [5] is a highly scalable method for earthquake detection and location based on a single waveform. It takes a window of three-channel waveform seismogram data as input and predicts this window's label either as seismic noise or as an event and outputs a probabilistic location of an earthquake's source. Other approaches in recent years had a focus on first-motion polarity and P-wave arrival picking [6] or generalizing seismic phase detections [7], employing similar models. A quite new model, CRED [4], combines CNN with RNN in a residual structure that learns time-frequency characteristics which are able to improve event detection by better characterizing seismic signals. This way, the model is more robust to noise. All discussed models use an FCNN in the last level.
Most relevant to this thesis is a paper that was published in 2017, "Attention

Is All You Need" [8]. The authors present a model called Transformer. It is a sequence transduction model, which unlike the dominant models, eschews the use of recurrence and convolutions. Typically, such models include an encoder and a decoder. The best performing ones also connect them through an attention mechanism. The Transformer network uses the encoder-decoder model together with a concept called self-attention, which allows for much shorter training times and better results in the field of language tasks.

3 Adaption & Progression

The aim of the thesis is to explore possible adaptions of the Transformer model for P-wave picking. A sketch of the planned model can be seen in Figure 1. The main challenges for the adaption are as follows.

The authors of the Transformer model train their model on language translation. The input is a sentence consisting of words. To represent words, they use word embeddings, in their case vectors with 512 dimensions (dimension d_{model}). In our scenario, the (3-channel) seismic data, each channel forming a sequence of amplitude values, will be cut into pieces by temporal windowing. The pieces pass through CNNs first. Each piece of fix length with 3 channels forms a 2D-matrix, which can be convoluted by a 1D kernel (moving only in one direction) as it is mostly used for time series. The CNNs' output X will be a sequence of values, one for each window, that serves as input to the Transformer unit. X can also be described as the internal representation of the seismic signal.
Since the order of its elements is important, we will apply positional encoding on each element. We will start with taking the sinusoidal functions and evaluate if they are suitable enough.
There will be no need for a decoder unit since the output length of our model will be proportional to the length of its input. To accomplish our final output sequence, we add a pointwise FCNN (and a Softmax Layer) calculating the desired probability values over time.
As soon as a properly tuned model following this architecture achieves satisfying performance, we will try to enhance our approach by including grographical information, in a next step. This will be done by interposing an enrichment unit, which processes the intermediate representation (X or Y)

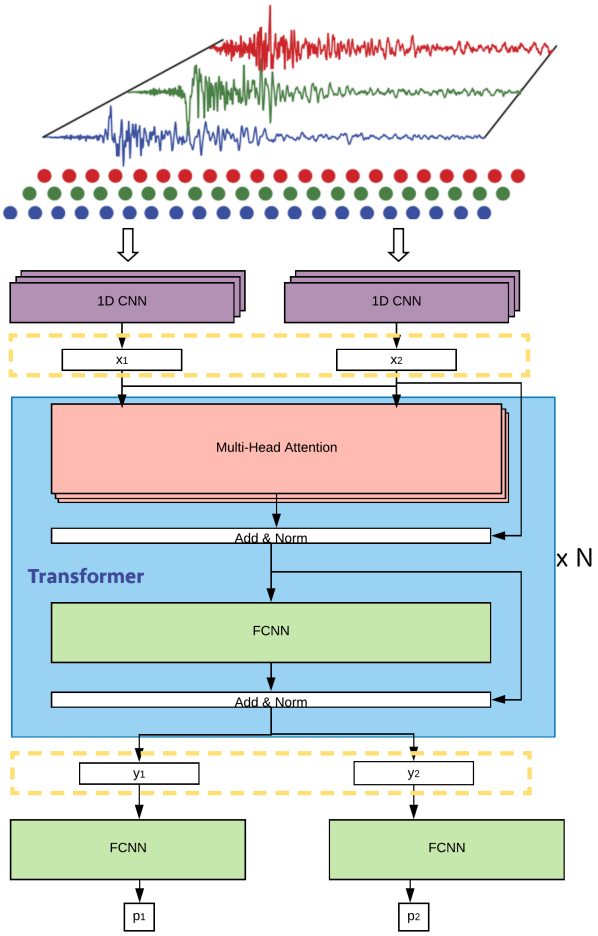

Figure 1: Preview of our model architecture. The seismic signal has 3 channels and gets sampled and windowed. In this example we only have two time windows for didactic purposes. Each of which passes several 1D-CNN layers, resulting in x_1,x_2 forming X, which is passed through the Transformer encoder unit. All elements of X are processed in parallel, multi-head attention can be parallelized, as well. The following FCNN is applied per each (attention-encoded) window. The Transformer unit as a whole repeats N times in sequence. In the end Y is passed to a pointwise FCNN and then to a softmax-layer (not shown) to calculate our final probability vector P. The yellow dashed boxes mark the positions where the enrichment unit may be attached. (Parts of the image are taken from [5])

once more in a separate FCNN. This FCNN takes the internal representation of the signal (resp. the Transformer output) and the provided geographical information of the station as input. The output is forwarded to the next unit in line.
As the most important work is done by the Transformer unit, the enrichment unit should be inserted right before or after the Transformer. We will try both ways.
Finally, a FCNN will be applied leading to an output for each element.

After having trained the model on picking P-waves with satisfying results, if time permits, we extend our model with a new target (S-wave) and enrich the training data with S-wave picks.

4 Data

The data will be provided by the GFZ Potsdam. It contains about 300,000 picks, distributed over approximately 3,000 events. The 3-component seismograms were recorded by different stations. We are going to detrend, bandpass filter, resample and normalize the data.
For ground truth, we will generate a label vector with the same length as the output vector. We will have to define a range around the true pick with a normal distribution between 0 and 1, setting all the rest of the vector to 0.
We will split the data into training, validation and test sets in a not yet specified ratio.

5 Evaluation

The training task is treated as a multi-label classification, hence we use binary cross-entropy as loss function. The tests are evaluated using the mean squared error based on the absolute error between predicted pick and true pick for each test case.
For being independent of the actual distribution of errors, we strive to catch outliers by considering the error quantiles (e.g. 5th, 10th, 90th, 95th percentile).

Our results will be compared with a deep learning and a classic approach.

1. CNN model
   As a deep learning approach, we feed [6] with the same data. It consists of CNNs which, on the one hand, act as regressors for determining the precise onset time for each P-arrival and, on the other hand, as classifier for first-motion polarities (which we will ignore).

2. STA/LTA
   It is based on short-term-average to long-term-average ratios (STA/LTA) calculated from an approximative squared envelope function of the seismogram [1, 2].

6 Workflow

We are going to build a modular software pipeline consisting of several modules, including one module to load all the data, one to build a model by a given configuration file, and one to train the model on the given data.
All model tuning steps and experiments will be documented and intermediate results will be saved.
The following phases should be completed:

1. *Data Pre-Processing.* Collect the data thoroughly, prepare it to be processed by the model, divide it into training and validation set and a (much smaller) test set.

2. *Build and Evaluate Baseline Models on Test Data.* Test the pipeline; test baseline models on test data.

3. *Basic Model Implementation.* Build our model for P-wave picking without location enrichment.

4. *Training and Evaluation.* Debug the model, tune hyperparameters; train the model using k-fold cross-validation; evaluate on test data set; compare with baseline tests.

5. *Design, Train and Evaluate Model with Location Enrichment Unit.* (If the previous tests were satisfying:) Build the location enrichment unit, attach it to the model; train it; evaluate it.

6. *Optional if success and time:*

    - *Add Support for S-Wave Picks to Model.* Extend the model to pick S-waves.
    - *Train and Evaluate Model on S-Wave Picks.* Train it; evaluate it on the test data.
    - *Compare with Baseline.* Prepare baseline models to pick S-waves (if possible) and compare the results with ours.

References

[1] Rex Allen. Automatic phase pickers: Their present use and future prospects. *Bulletin of the Seismological Society of America*, 72(6B):S225–S242, 1982.

[2] Rex V Allen. Automatic earthquake recognition and timing from single traces. *Bulletin of the Seismological Society of America*, 68(5):1521–1532, 1978.

[3] Tobias Diehl, Eduard Kissling, and Peter Bormann. Tutorial for consistent phase picking at local to regional distances. In *New Manual of Seismological Observatory Practice 2 (NMSOP-2)*, pages IS–11. GeoForschungsZentrum, 2012.

[4] S Mostafa Mousavi, Weiqiang Zhu, Yixiao Sheng, and Gregory C Beroza. Cred: A deep residual network of convolutional and recurrent units for earthquake signal detection. *Scientific reports*, 9(1):1–14, 2019.

[5] Thibaut Perol, Michaël Gharbi, and Marine A. Denolle. Convolutional neural network for earthquake detection and location. In *Science Advances*, 2018.

[6] Zachary E Ross, Men-Andrin Meier, and Egill Hauksson. P wave arrival picking and first-motion polarity determination with deep learning. *Journal of Geophysical Research: Solid Earth*, 123(6):5120–5129, 2018.

[7] Zachary E Ross, Men-Andrin Meier, Egill Hauksson, and Thomas H Heaton. Generalized seismic phase detection with deep learning. *Bulletin of the Seismological Society of America*, 108(5A):2894–2901, 2018.

[8] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NIPS*, 2017.

[9] Jin Wang and Ta-Liang Teng. Artificial neural network-based seismic detector. *Bulletin of the Seismological Society of America*, 85(1):308–319, 1995.

[10] Jin Wang and Ta-liang Teng. Identification and picking of s phase using an artificial neural network. *Bulletin of the Seismological Society of America*, 87(5):1140–1149, 1997.

[11] Jan Wiszniowski, Beata M Plesiewicz, and Jacek Trojanowski. Application of real time recurrent neural network for detection of small natural earthquakes in poland. *Acta Geophysica*, 62(3):469–485, 2014.