PickNet: Deep-Learning for Picking Seismic Arrival Times

# **PickNet**

This is the implementation of the PickNet framework for picking seismic arrivals with deep-learning. The framework and associated evaluations are well described in paper ‘Deep-Learning for Picking Seismic Arrival Times’ (June 2019, Journal of Geophysical Research: Solid Earth, DOI: 10.1029/2019JB017536)

We have also provided **trained model** in our paper and **example data** from IRIS to demonstrate how to use PickNet.

We thank NIED Hi-net for providing high quality waveform data in our research. However, we are sorry that we are not able to provide training and testing datasets used in our JGR study due to NIED Hi-net’s data policy, ***which requests all users to download the data from NIED Hi-net web page and does not offer a data hosting service for users.*** Therefore, we are not allowed to redistribute Hi-net waveform data.

An alternative solution is for researchers to download the waveform data and arrival information data directly from Hi-net web page (<https://hinetwww11.bosai.go.jp/auth/?LANG=en>).

If you use PickNet in your study, please cite our paper ‘Deep Learning for Picking Seismic Arrival Times’ (Wang et al., 2019, JGR, doi:10.1029/2019JB017536).

PickNet is licensed under the GNU General Public License v3.0.

# **Installing**

You will need a Nvidia GPU with driver installed (PickNet can run without GPU but it will be very slow). If you already have Python (version 3.6), Tensorflow (version around 1.8), CUDA, Pyyaml and Matplotlib installed. You just need to unzip the codes. Otherwise, we strongly recommend you install requisites via Anaconda following steps below:

1. Install Anaconda3 (https://www.anaconda.com/distribution/).
2. Open a terminal (Linux or macOS) or an Anaconda Prompt (Window) then type:

conda create -n picknet\_env python=3.6 tensorflow=1.8 cudatoolkit=X.X pyyaml matplotlib

(Your should choose cudatoolkit version based on your driver version. Please see https://docs.nvidia.com/deploy/cuda-compatibility/index.html)

1. Unzip the PickNet code.

# **Training**

1. Prepare training file. The training file is **a list** of **lists** of **tuples** saved using Numpy module (numpy.save()). The first element in each tuple is the wavefrom and the second element is the pickpoint. Tuples of different epicentral distances are divided into different list.
2. Write configuration file. You can find an example of train configuration file in ‘PickNet/fcn/configs/train\_config\_example.yaml’. You can set parameters such as learning rate, batch size, etc.
3. Type conda activate picknet\_env in terminal and get into folder ‘picknet’.
4. Type python seismic\_pick\_run.py --train --config-file FILE\_PATH --gpuid 0

(We prepared configuration examples to train PickNet for P and S picking under ‘Data/TrainSet’ folder. For example, you can run ‘python seismic\_pick\_run.py --train --config-file ../Data/TrainSet/P\_train.yaml --gpuid 0’ to train a first P-wave picking model. Be sure to change the configuration file path if you put it somewhere else)

# **Testing**

1. Prepare Testing file. The testing file is a k×m×n array saved using Numpy module (numpy.save()) where k is instance number, m is time point number, n is input channel number. You can generate your own testing file based on IRIS waveform picking example .
2. Write configuration file. You can find an example of test configuration file in ‘picknet/fcn/configs/test\_config\_example.yaml’
3. Type conda activate picknet\_env in terminal and get into folder ‘picknet’.
4. Type python seismic\_pick\_run.py --test --config-file FILE\_PATH --gpuid 0
5. We prepared an example of how to use PickNet to pick data from IRIS in folder ‘PickingDataFromIRISExample’. The example contains the process from downloading waveform data to generate txt pick file. There is a README file under that fold describe the process in detail. You can make some modifications to scripts in example folder to use PickNet pick your own data.

# **Acknowledgements**

# High-quality arrival-time datasets for this study were provided by the data centers of Tohoku University (http://www.aob.tohoku.ac.jp), the JMA Unified Earthquake Catalogue (https://hinetwww11.bosai.go.jp/auth/JMA/), the China Earthquake Administration (https://www.cea.gov.cn/), the International Seismological Centre (http://www.isc.ac.uk). Waveform data for this study were provided by the Hi-net (http://www.hinet.bosai.go.jp), IRIS (https://www.iris.edu), the International Federation of Digital Seismograph Networks (https://www.fdsn.org), the Southern California Earthquake Data Center (http://scedc.caltech.edu/), and Data Management Centre of the China National Seismic Network at Institute of Geophysics, China Earthquake Administration ([http://www.seisdmc.ac.cn](http://www.seisdmc.ac.cn/)). Great thanks to organizations above!

# **Feedbacks**

If you run into bugs or other problems while using PickNet, you can send an e-mail to Jian Wang (jianwang@mail.iggcas.ac.cn) or Zhuowei Xiao (xiaozhuowei@mail.iggcas.ac.cn). We would be happy to help you.

We will probably move our code to tensorflow 2.0 and host our code at github in the future for convenient version control and maintenance.