Practical deep learning inversion using neural architecture search and a flexible training dataset generator

Introduction

Seismic waveform inversion is a time-consuming process. Seismic simulations and interventions requiring expert knowledge are required many times during the procedure. Recently, researchers have started to implement deep learning techniques to estimate subsurface by the physical properties, such as velocities, directly from seismic record data. This approach would reduce the turn-around time of seismic inversion but has two major challenges: training data acquisition and neural network modelling.

Deep learning typically requires a large training dataset, including seismic data and the corresponding ground-truth data of the subsurface velocity. Kazei et al. (2020), Zhang and Lin (2020), and other researchers have presented various ways to generate velocity models for training purposes. However, a large-scale training dataset of realistic subsurface structures remains difficult to prepare without data leakage, that is, the contamination of the training data with illegitimate information (Kaufman et al. 2012). Creating a training dataset using a part of the test dataset is a clear example of data leakage; this leads to overfitting and results in a poor generalisation performance in practical situations.

Selecting the best-suited neural architecture for seismic inversion represents a further challenge. Since seismic inversion can be regarded as a variant of image-to-image translation problem, we can utilise well-established deep learning approaches to input seismic data and output velocity models immediately. The residual network (ResNet) (He et al. 2016) is a successful model in the field of computer vision and it has also been commonly used for seismic inversion (Liu et al. 2020, Wang et al. 2020). Meanwhile, Li et al. (2019) highlighted that seismic inversion has unique characteristics, for example, weak spatial correspondence between seismic data and velocity models. This observation is suggestive of significant improvements by designing neural architectures dedicated to seismic inversion. Despite its importance, the process of manually searching for a suitable neural architecture is time-consuming and cumbersome.

In this study, we address the two challenges: the practical acquisition of a large-scale training dataset and the search for a high-performance neural network model. First, we propose a flexible system which parametrically generates velocity models to create a large-scale, complex, and fully synthetic training dataset, based only on geological insights. Second, we employ a neural architecture search (NAS) using Optuna (Akiba et al. 2019) (https://optuna.org/), which is an automatic hyperparameter optimisation framework. We automate the design of a suitable ResNet-based neural network model in terms of hyperparameters, such as the number of layers and channels.

Parametric Velocity Model Generation System

We propose a parametric velocity model generation system to acquire a large-scale training dataset. This system is designed to prevent data leakage and enable the proper evaluation of the generalisation performance. First, the system generates synthetic subsurface structure models, that is, realistic and high-resolution velocity distributions. The velocity structure is generated through synthetic processes corresponding to geological events, including stratification, folding, faulting, intrusion, and erosion. Each process is modelled using geological parameters, such as layer thickness, wave velocity, and fault dip angle. Figure 1 shows examples of the velocity models. We simulated the seismic wave propagation on the generated subsurface model to generate the corresponding shot gathers.

Our system incorporates probabilistic distributions to draw many samples of geological parameters, thus creating a large-scale training dataset, including a wide variety of subsurface structures. In this study, we used uniform distributions with a given set of hyperparameters, that is, the upper and lower bounds of the geological parameter values. The hyperparameters were determined according to geological insights, for example, a rough estimation of the velocity structure. In particular cases, the insights of the target subsurface could be obtained using geological surveys such as conventional inversion methods and nearby well log data. A high-quality training dataset can be generated by placing reasonable assumptions on these geological parameters.
a few days. Finally, we sampled 10,000 of the 300,000 synthetic data, comprising velocity models and corresponding shot gathers. The generation parameters, which control the geological properties of the synthetic velocity models, were determined based on the geological insights of the Marmousi2 model (Martin et al., 2006). Abstract information about the geological properties was utilised, but the benchmark datasets themselves were never used; that is, the training data did not include any part of the benchmark data. The seismic wave propagation was simulated by MN-2, Preferred Networks’ private supercomputer. This process cost approximately 200,000 CPU core hours.

We applied nested dataset partitioning, which prevents data leakage during the NAS process. We split the 300,000 synthetic data into a large training dataset of 240,000 samples and a large validation dataset of 60,000 samples. These large datasets were used to train the optimal neural architecture. We further sampled 10,000 data from the large training dataset and split this subset into a small training dataset of 8,000 samples and a small validation dataset of 2,000 samples. These small datasets were used to search for the optimal neural architecture. We employed this partitioning to avoid illegitimate access to the large validation dataset during the NAS step.

We optimised our ResNet-based encoder–decoder models by tuning the hyperparameters, such as the number of channels and layers. This optimisation required approximately 1,000 GPU hours. However, the combination of the supercomputer MN-2 and parallelisation by Optuna reduced the running time to a few days. Finally, we obtained the optimal neural architecture, which had more than 100 hidden layers; this was much deeper than those determined by previous work.

**Experimental Results**

We applied the aforementioned methods for an inversion of two-dimensional (2D) acoustic velocity images from 2D shot gather images. Our experiment comprised the following four steps: synthetic dataset generation, dataset partition, NAS, and evaluation of the optimal neural architecture.

**NAS for Seismic Inversion**

NAS automatically designs a neural network suitable for a given task. Once the search space of the neural architecture has been defined, NAS tries to find an optimal architecture in the search space by successively sampling, training, and evaluating a candidate architecture. This architecture is determined by model hyperparameters, including the number of layers and channels. In this study, we employed Optuna, a hyperparameter optimisation framework. Optuna’s user-friendly interface allows easy and automatic hyperparameter optimisation through parallel processing, which reduces computational time.

In this study, we defined the search space of the neural architecture as ResNet-based encoder–decoder models. ResNet is a popular deep learning model that has been used for both seismic inversion and computer vision. Notably, we focused on encoder–decoder models, which associate inputs and outputs with a common latent feature space. While convolutional neural networks typically capture spatially local features, the encoder–decoder models tend to learn spatially global features. This characteristic should be necessary for seismic inversion.
Figure 2 Vp estimations of the Marmousi2 model: (a) ground truth, (b) estimation using our method, (c) estimation using the ResNet50-based encoder–decoder model, (d) one-dimensional (1D) profile at 2 km position corresponding to the red lines in (a)–(c), and (e) 1D profile at 11 km position corresponding to the blue lines.

Figure 3 Vp estimations of the 1994 Amoco statics test dataset: (a) ground truth and (b) estimation using our method.

We trained the optimal neural architecture using the large training dataset. We evaluated our method using two standard benchmark datasets: the Marmousi2 model and the 1994 Amoco statics test dataset. Figure 2 shows the inversion results of the Marmousi2 model using our optimal model, compared to the model based on the ordinary ResNet50 (He et al. 2016). Owing to the quality and quantity of our training dataset, the baseline ResNet50-based model roughly reproduced the velocity model. However, the optimal model showed a more comprehensible result. It predicted a more accurate velocity for the salt layer (4 km depth in Figure 2d) and even obtained a more detailed structure in the complex area around the faults (Figure 2e). Figure 3 shows the modelling results of the 1994 Amoco statics test dataset. Our method estimated a high-resolution and comprehensible output, despite the training dataset did not include any information on the 1994 Amoco statics test dataset.

Discussion and Conclusions

In this study, we addressed two challenges in practical deep-learning inversion: acquiring a large-scale training data and selecting the best architectures in a neural network. We proposed a flexible system which parametrically generates velocity models to create a large-scale, complex, and fully synthetic training dataset. We also proposed the use of NAS techniques to optimise a neural architecture. Our experimental results show that deep learning can provide detailed inversion directly from shot gathers and, therefore, could provide a shortcut to the inversion workflow.
We generated 300,000 velocity models using geological insights from the Marmousi2 model for our experiment; data leakage was carefully avoided. This elaborated training dataset allowed even the ordinary ResNet50-based model to estimate the Marmousi2 model to some extent. This result indicates that generating a large-quantity and high-quality training dataset is crucial to achieve high performance in practical situations. Meanwhile, it also indicates potential limitations. Training dataset based on the erroneous geological insights would cause poor generalisation performance. Without the geological insights, the training dataset needs to be large and diverse enough to cover any possible subsurface models. Conventional inversion methods remain most functional in taking practical hints in such cases.

This study also demonstrated that NAS, when combined with high-performance computing resources, is a powerful tool for realising precise deep-learning inversion. We employed Optuna to search for the optimal neural architecture and found quite a deep model with more than 100 hidden layers that showed successful and comprehensible results. The search space of NAS was limited to ResNet-based encoder–decoder models in this study, which may have overlooked the best in all the possible architectures. Therefore, future research should aim to find neural architectures that are more suitable for seismic inversion. The architecture can be trained either alone or in combination with a discriminator as a generative adversarial network (Zhang and Lin 2020); this also needs to be investigated in the future.

This study shows a comprehensible estimation of the Marmousi2 benchmark model. Moreover, despite no information of the 1994 Amoco statics test dataset was used to generate the training dataset, our model estimated it precisely. These results indicate that our framework can build neural-network-based inversions with high generalisation performances. Further work is underway to conduct inversion from real noisy seismic data and to predict other physical properties, such as S-wave velocity.

Acknowledgements

The authors would like to thank Tomoki Ito, Yuhei Kaneshita, and Kenichi Mori at MITSUI & CO., LTD. for valuable feedback. The authors are also grateful to Shoichiro Yamaguchi, Kentaro Minami, Daiki Higurashi, Akifumi Imanishi, Shotaro Sano, Takeru Ohta, and Yusuke Tsuzuku at Preferred Networks, Inc. for their technical advice and support. The 1994 Amoco statics test dataset was created by Mike O’Brien and Carl Regone and was provided courtesy of Amoco and BP.

References


Kaufman, S., Rosset, S., Perlch, C., and Stitelman, O. [2012] Leakage in data mining: Formulation, detection, and avoidance. ACM Transactions on Knowledge Discovery from Data, 6(4), article 15.


