A deep learning perspective of the forward and inverse problems in exploration geophysics

Jian Sun[1,2], Zhan Niu[1], Kristopher Innanen[1], Junxiao Li,[1,3], Daniel Trad[1]

[1] CREWES Project, Department of Geoscience, University of Calgary [2] College of Earth and Mineral Sciences, Pennsylvania State University [3] PETRONAS Global

Abstract

Deep learning technique has drawn numerous attentions and becomes to be extremely powerful tool in many fields of industry, where the recurrent neural network (RNN) shows significant advantages of exhibiting temporal dynamic behavior of time-dependency tasks by build a directed graph of a sequence. To explore the potential benefits of deep learning technique in exploration geophysics, in this paper we illustrate the forward modeling problem in a perspective of deep learning, by recasting the wave propagation in a RNN framework. Furthermore, the inverse problem, becoming to be a `training' process of RNN, is analyzed by deriving the gradient in sense of neural network. The result shows that, under some assumption, the inverse problem using RNN is approximately equivalent to the full waveform inversion (FWI). With the proposed specific RNN framework, we theoretically and numerically analyze the best learning rate (i.e., usually being called step-size in geophysics) ranges for most popular gradient-based optimization algorithms used in deep learning tasks. The efficiency of the gradient-based algorithms is investigated by comparing with non-linear optimization methods, such as non-linear conjugate gradient (CG) and L-BFGS, on a 2D Marmousi model.

The forward problem in deep learning framework

The forward problem, namely, seismic wavefield modeling, usually is being denoted as the partial differential wave equation (Carcione et al., 2002). Assume an acoustic media in 2D with a constant density, the wave equation in time domain is written as

$$\nabla^2 \boldsymbol{u}(\mathrm{r},\mathrm{t}) = \frac{1}{\mathrm{v}^2(\mathrm{r})}\frac{\partial^2 u(r,t)}{\partial t^2} + s(r,t)\delta(r - r_s) \qquad (1)$$

where ∇^2 denotes the (spatial) Laplacian operator, and the spatial coordinates is described by r. u usually represents the pressure or displacement for acoustic medium wave propagation, with the time coordinate t. The source term is denoted by s.

The forward modeling of seismic wave propagation is conventionally performed by discretizing the wave equation (Carcione et al., 2002). To simplify the problem and make it intelligible for readers, the second-order finite difference method (Virieux, 1986), both in time and spatial coordinates, is applied to solve the wave propagation. The mathematical formulation is written as

$$\boldsymbol{u}(\boldsymbol{r}, t + \Delta t) = v^2(\boldsymbol{r})\Delta t^2[\nabla^2 \boldsymbol{u}(\boldsymbol{r}, t) - \boldsymbol{s}(\boldsymbol{r}, t)\delta(\boldsymbol{r} - \boldsymbol{r_s})] + 2\boldsymbol{u}(\boldsymbol{r}, t) - \boldsymbol{u}(\boldsymbol{r}, t - \Delta t) \qquad (2)$$

Equation (2) shows that the forward modeling of wave propagation can be considered as an iterative process which takes the source term $\mathrm{s}\,(\mathrm{r_s}, \mathrm{t})$ and the two previous time steps wavefields as inputs. It inspires us that it is possible to solve the forward modeling problem with a recursive neural network where each time layer (i.e., cell) denotes the wavefield modeling at one single time step. A much robust version can be found in paper by Richardson (2018).

To theory-guiding RNN architecture, a single cell of designed forward modeling RNN represents the finite-difference operator, which takes the sequence at one single time step as the input, outputs the modeled shot record at current time step, and save the memory and the modeled wavefield of this block for the next time step modeling. The cell architecture of forward modeling RNN is illustrated in Figure 1.

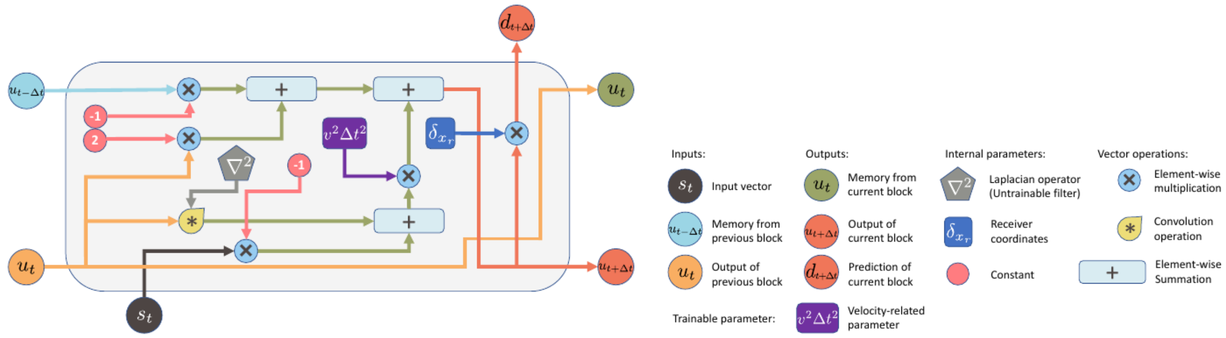

Figure 1. The cell architecture of designed RNN framework.

The inverse problem in deep learning framework

The inverse problem in geophysical exploration is to obtain the subsurface model parameters based on the approximation of medium type, which is usually being treated as a least-squares local optimization problem by minimizing the square of the misfit between the recorded seismic records and the modeled seismic data, i.e., $\delta \boldsymbol{d} = \boldsymbol{d_{obs}} - \boldsymbol{d_{pred}}$ (Tarantola,1983 Virieux and Operto, 2009). The formulation of least-square norm of misfit function is written as

$$\mathrm{J}(\boldsymbol{v}) = \frac{1}{2n_s} \sum_{r_s} \sum_{r_g} \sum_t (\boldsymbol{d}_t - \tilde{\boldsymbol{d}}_t)^2 \tag{3}$$

In perspective of RNN's framework, the gradient calculation of the objective function with respect to trainable parameters is achieved by the chain rule performing in sense of reversal time. The gradient formulation is written as

$$\mathbf{g} = \frac{\partial J}{\partial \boldsymbol{v}} = \sum_{t=0}^{T} \left[\frac{\partial J}{\partial \tilde{\boldsymbol{u}}_t}\right] \frac{\partial \tilde{\boldsymbol{u}}_t}{\partial \boldsymbol{v}} \tag{4}$$

In equation 4, the partial derivative $[\,\partial J / \partial \tilde{\boldsymbol{u}}_t]$ is calculated using chain rule with time-dependency terms determined by the order of finite-difference of forward modeling. Considering the second-order finite-difference modeling, its formulation is coherent to two dependency terms at next two time steps, which can be expressed as

$$\left[\frac{\partial J}{\partial \tilde{\boldsymbol{u}}_t}\right] = \left[\frac{\partial J}{\partial \tilde{\boldsymbol{u}}_{t+2}}\right] \frac{\partial \tilde{\boldsymbol{u}}_{t+2}}{\partial \tilde{\boldsymbol{u}}_t} + \left[\frac{\partial J}{\partial \tilde{\boldsymbol{u}}_{t+1}}\right] \frac{\partial \tilde{\boldsymbol{u}}_{t+1}}{\partial \tilde{\boldsymbol{u}}_t} + \frac{\partial J}{\partial \tilde{\boldsymbol{u}}_t} \tag{5}$$

where the initial conditions of RNN backpropagation are assumed to be zeros, i.e., $[\,\partial J / \partial \tilde{\boldsymbol{u}}_t]_{t=T+1,T+2} = \mathbf{0}$. After do the math, the partial derivative can be reformulated as

$$\left[\frac{\partial J}{\partial \tilde{\boldsymbol{u}}_t}\right] = \boldsymbol{v}^2 \Delta t^2 \left(\nabla^2 \left[\frac{\partial J}{\partial \tilde{\boldsymbol{u}}_{t+1}}\right] - \frac{1}{n_s \boldsymbol{v}^2 \Delta t^2} \sum_{r_s} \sum_{r_g} \delta d_t\right) + 2\left[\frac{\partial J}{\partial \tilde{\boldsymbol{u}}_{t+1}}\right] - \left[\frac{\partial J}{\partial \tilde{\boldsymbol{u}}_{t+2}}\right] \tag{6}$$

Equation 6 shows that the partial derivative of the objective function over predicted wavefield $[\,\partial J / \partial \tilde{\boldsymbol{u}}_t]$ is performed by propagating the scaled data residual in reversal time. Therefore, the gradient for model perturbation is rewritten as

$$\begin{aligned} \mathbf{g} &= \sum_{t=0}^{T} BP\left(-\frac{1}{n_s \boldsymbol{v}^2 \Delta t^2} \sum_{r_s} \sum_{r_g} \delta \boldsymbol{d}_t\right) \frac{\partial \tilde{\boldsymbol{u}}_t}{\partial \boldsymbol{v}} \\ &\approx \sum_{t=0}^{T} BP\left(-\frac{1}{n_s} \sum_{r_s} \sum_{r_g} \delta \boldsymbol{d}_t\right) \frac{2}{\boldsymbol{v}^3} \frac{\partial^2 \tilde{\boldsymbol{u}}_t}{\partial t^2} \end{aligned} \tag{7}$$

where, BP(s) indicates the back-propagation of source s in reversal time (T → 0). $\delta \boldsymbol{d}_t$ represents the residuals between observed and predicted data. Further, equation 7 indicates that the perturbation model obtained by the gradient is performed by the crosscorrelation between the second-order partial derivative of the forward wavefield over time and the back-propagation wavefield using the residual as the source, which is equivalent to the gradient of time-domain full-waveform inversion shown in paper by Yang et al. (2015). In other words, FWI is also a specified machine learning process with a self-designed RNN framework, where the unknown subsurface parameters are treated as the trainable weights of neural network.

A key problem in deep learning training process is the hyperparameter tuning, which is still one of the toughest obstacles. In deep learning cases, the learning rate is usually determined by empirical analysis or some trials, which is in range of (0, 1]. However, the proposed RNN in this paper is not a typical architecture presented in literatures. Detailed analysis of hyperparameter tuning of such a RNN can be found in our report (Sun et al., 2018).

The synthetic example of Marmousi

To fully examine the capacity of casted RNN framework for velocity modeling, the 2D synthetic Marmousi (Figure 2) is employed to create synthetic short records as observed data, and a smoothed model is considered as initialization. For comparison, we also performed traditional FWI using nonlinear CG (Hu et al., 2011) and l-BFGS (Morales and Nocedal, 2011) algorithms, respectively.

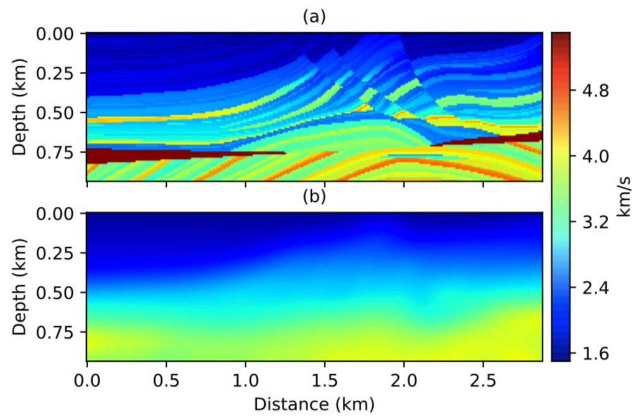

Figure 2. (a) True Marmousi. (b) initial model.

To effectively compare the efficiency of these algorithms, the predicted results versus the forward modeling iteration are plotted. In Figure 3, the RNN inversion results using non-linear CG method is delineated at [400, 800, 1420]th forward iterations. We can observe that the non-linear CG algorithm is able to retrieve the major structures of 2D Marmousi model, however it requires a tremendous computational cost because of its slow convergence speed. Figure 4 shows the RNN modelled velocity of 2D Marmousi using l-BFGS algorithm. Comparing to non-linear CG method, l-BFGS has much faster convergence speed and is able to obtain more precise and detailed results with less iterations. Finally, we implement the velocity building using the most popular deep learning algorithm, i.e., Adam (Kingma and Ba, 2014), through the designed RNN framework. Based on the numerical analysis of hyperparameter tuning (Sun et al., 2018), for velocity inversion, a suitable learning rate is in range of [10,100], which may provide fastest and

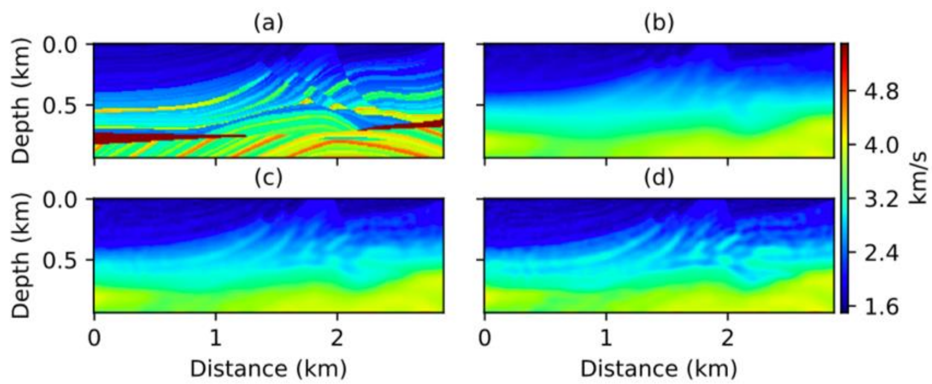

Figure 3. The inversion of Marmousi using CG algorithm. (a) True Marmousi. (b) at 400th iteration. (c) at 800th iteration. (d) at 1420th iteration.

stable convergence process. Furthermore, the analysis shows the effects of learning rate can be neglected as long as it is in the suitable range. In Figure 5, after 50 iterations, Adam obtained a better resolution of velocity model than non-linear CG and l-BFGS. With 100 iterations, a very precisely velocity model of Marmousi is built. The efficiency comparison is plotted in Figure 6. It shows that, Adam is capable of retrieving the detailed subsurface parameters with much less iterations.

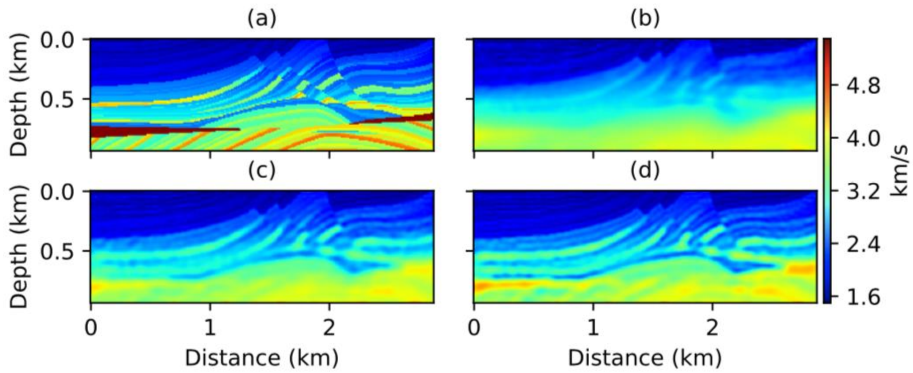

Figure 4. The inversion of Marmousi using L-BFGS algorithm. (a) True Marmousi. (b) at 200th iteration. (c) at 600th iteration. (d) at 1000th iteration.

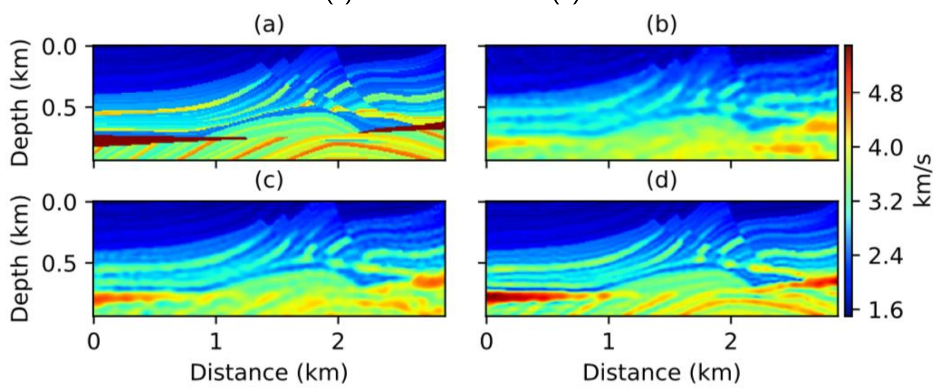

Figure 5. The inversion of Marmousi using Adam algorithm. (a) True Marmousi. (b) at 25th iteration. (c) at 50th iteration. (d) at 100th iteration.

Conclusion

Deep learning has been tremendously improved and widely applied in different fields. To benefits from the well-developed deep learning community and open-source libraries, we proposed a self-designed recurrent neural network which allows us to cast the forward modeling of seismic wave propagation into the forward propagation of RNN framework. As a consequence, the geophysical inversion problem is also turned into a training process of the presented RNN framework, where full wavefield information are involved. The gradient derivation using chain rule in deep learning perspective shows that the training process of the RNN is equivalent to a FWI problem. In other words, it is proven that FWI is also a specific machine learning case. A 2D acoustic Marmousi model is employed to examine the full capacity of RNN using nonlinear CG, l-BFGS and Adam algorithms, respectively.

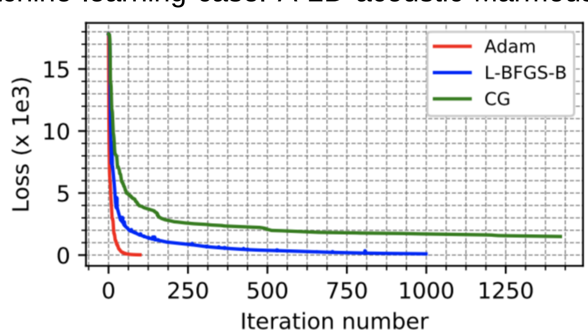

Figure 6. The convergence comparison of non-linear CG , L-BFGS, and Adam algorithms on 2D Marmousi model.

Acknowledgements

We thank the sponsors of CREWES for continued support. This work was funded by CREWES industrial sponsors and NSERC (Natural Science and Engineering Research Council of Canada) through the grant CRDPJ 461179-13.

References

Carcione, J. M., Herman, G. C., and Ten Kroode, A., 2002, Seismic modeling: Geophysics, 67, No. 4, 1304–1325.

Richardson, A., 2018. Seismic Full-Waveform Inversion Using Deep Learning Tools and Techniques. arXiv preprint arXiv:1801.07232.

Hu, W., Abubakar, A., Habashy, T., and Liu, J., 2011, Preconditioned non-linear conjugate gradient method for frequency domain full-waveform seismic inversion: Geophysical Prospecting, 59, No. 3, 477–491.

Kingma, D. P., and Ba, J., 2014, Adam: A method for stochastic optimization: arXiv preprint arXiv:1412.6980.

Morales, J. L., and Nocedal, J., 2011, Remark on algorithm 778: L-bfgs-b: Fortran subroutines for large-scale bound constrained optimization: ACM Transactions on Mathematical Software (TOMS), 38, No. 1, 7.

Sun, J., Niu, Z., Innanen, K., Li, J., and Trad, D., 2018, A deep learning perspective of the forward and inverse problems in exploration geophysics: CREWES Research report.

Tarantola, A., 1984, Inversion of seismic reflection data in the acoustic approximation: Geophysics, 49, No. 8, 1259–1266.

Virieux, J., and Operto, S., 2009, An overview of full-waveform inversion in exploration geophysics: Geophysics, 74, No. 6, WCC1–WCC26.