Deep Learning at SEG 2020

Machine learning dominated the technical program at the Society of Exploration Geophysicists (SEG) conference this year; with Deep Learning applications being most popular. A consideration of the mechanics of convolutional neural networks (CNNs) and generative adversarial networks (GANs) leads to a comparison of several complementary efforts to resolve the most obvious challenge to such pursuits in the geosciences (the lack of real training data), and several complementary efforts to resolve one of the key weaknesses in Full Waveform Inversion (the lack of very low frequency signal in the recorded data). Despite the hype that can accompany this broad topic, encouraging progress is being made towards geoscientists being able to make better informed decisions, using more (all) data, and in less time.

A Virtual Event

The entirely virtual SEG 2020 conference was held 11-16 October, and consisted of pre-recorded oral presentations from the technical program, with live-streamed Zoom broadcasts of the post-convention workshop program. Machine learning (ML) topics formed the largest component of the overall program, and are the basis of the discussion below. Deep Learning pursuits based upon various forms of neural networks were prevalent, so I briefly summarize the comparative ambitions of convolutional neural networks (CNNs) and generative adversarial neural networks (GANs), before sharing a few emerging applications to the geosciences in the SEG 2020 program.

Machine Learning in the Geosciences

ML applications to seismic pursuits are typically used to optimize and simplify repeatable processes, or to replicate aspects of human-intensive tasks such as seismic data interpretation. In principle, if comprehensive 'labeled' data are available from several complementary information sources, ML may provide better-informed decisions and forecasts—uncorrupted by the fatigue or inconsistency that affects humans applied to onerous and time-consuming tasks—although we are a long way from matching human ingenuity and experience-based instinct. As summarized in the next sections, all modern Deep Learning pursuits take a modular approach to building deep neural networks that abstract operations into layers, which can be configured into flexible input and output configurations. This necessity to abstract the operations being simulated is a fundamental limitation on the scope of applications for Deep Learning, and a timely reminder that there are already several decades of other ML pursuits such as Monte Carlo-based, Support Vector Machine (SVM) and tree-based Random Forest statistical classifications that can robustly satisfy a vast range of practical optimization ambitions. An example application to one of the largest bottlenecks in seismic imaging workflows—velocity model building—is the automated PGS hyperModel approach that generates accurate models on a vast scale in only a few days; with statistical uncertainties if desired. All discussion below, however, concerns Deep Learning presentations.

Convolutional Neural Networks (CNNs)

As the name suggests, a convolutional neural network (CNN) is an artificial neural network that features one or more convolutional layers. This layer configuration enables a deep learning model to efficiently process spatial patterns, which makes convolutional layers especially appropriate to applications to the field of Computer Vision. Beyond image classification; applications include object detection, wherein the algorithm is tasked with isolating objects (or anomalies) within an image; and image segmentation, wherein several overlapping events within an image can be discriminated. Semantic segmentation is important in seismic interpretation, and early published applications of CNNs have included salt body detection, fault and event interpretation, clinoform interpretation, facies classification, the prediction of lithology and fluid properties, and so on.

As discussed below, CNNs can be trained to statistically estimate filters in seismic imaging workflows rather than explicitly formulate them in the traditional manner. **Figure 1** is taken from one example application by PGS wherein the convolutional filters within each layer were iteratively adjusted during the training step to remove targeted noise features and produce 'clean' outputs from 'noisy' inputs. Once trained, the CNN model was then applied to denoise seismic images that correspond to recorded field data. This two-step approach is common to Deep Learning pursuits: computational models are built that use inference and pattern recognition instead of explicit sets of rules.

A computational task such as classification, regression, or clustering is improved by conditioning of the model on a training data set. In a manner to seismic experts familiar with inversion-based pursuits, the performance of the model is measured with regard to a loss, which quantifies the performance of the model on the provided data. A good model eventually generalizes to data not used for model training, but with common characteristics and goals, on the same task the model was trained to perform.

U-Net is a popular form of CNNs applied to seismic data analysis, and was originally created for the purpose of segmenting biomedical images. The U-Net model, such as illustrated in **Figure 1**, consists of a fully convolutional architecture, and consists of three parts: the encoder (contraction; left branch), the bottleneck (bottom) and the decoder (expansion; right branch). These two paths—the contracting and expanding paths—are symmetrical; forming a 'U' shape. The contracting path serves to allow the model to learn high-resolution features from the image, and these high-resolution features are handed directly to the expanding path. By the end of the expanding path, we expect the model to have localized these features with the final image dimensions. After concatenating the feature maps from the contracting path onto the expanding path, a subsequent convolutional layer allows the network to learn to assemble and localize these features precisely. The final result is a network that is highly adept both at identifying features and at locating these features within multi-dimensional space. Computationally, the U-Net architecture utilizes several shortcuts in an encoder-decoder architecture to efficiently achieve stable segmentation results.

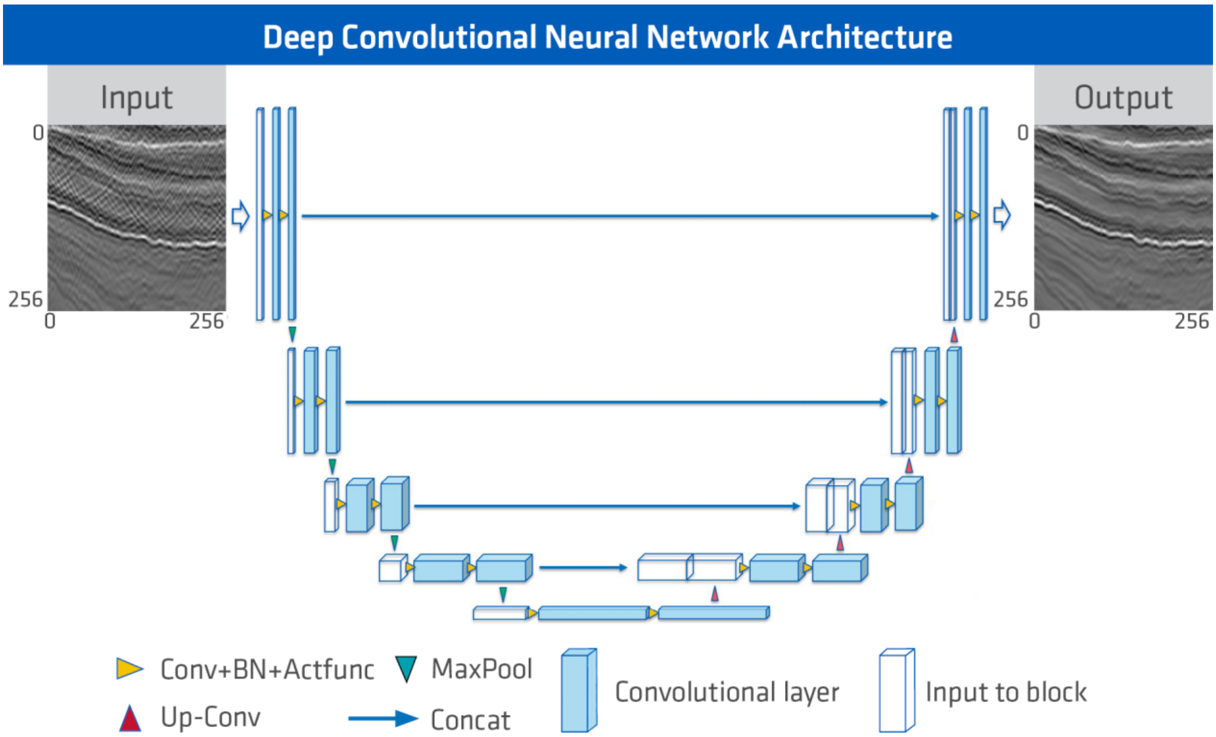

Figure 1. U-Net architecture used to attenuate migration artifacts from seismic images, courtesy of Elena Klochikhina, PGS.

Generative Adversarial Networks (GANs)

GANs provide an interesting alternative to CNNs, and are behind the 'Deep Fake' videos that are starting proliferate through the online space. At the highest level, a GAN involves two deep learning networks pitted against each other in an adversarial relationship, that implicitly learn a latent, low-dimensional representation of arbitrarily high-dimensional data. One network is the generator which produces samples (e.g. images or time series) in a certain style from an input, and the other network is a discriminator that attempts to distinguish whether input samples were produced by the generator or not, i.e. real or fake image from the training data. Training a GAN requires two opposing (i.e. adversarial) processes, wherein discriminator training alternates with generator training. In discriminator training, the generator produces output samples by forward-propagating some vector through the network, batches of the desired sample style are mixed into these outputs, and these collective samples are then input to the discriminator. A discriminator outputs a prediction that each output is real or fake, and a cross-entropy cost is back-propagated to train the discriminator model to correctly determine the validity of the input samples. During generator training, the generator receives a random noise vector as input, and produces a fake sample. All output samples from the generator are labeled as real, and the discriminator outputs predictions regarding whether the inputs were real or fake. The cross-entropy cost is used via back-propagation to train the generator to become better at fooling the discriminator. At the conclusion of overall training, the discriminator is discarded and the generator is the final product. Random noise input to the generator will then output samples that match the style of the samples the adversarial network was trained on.

The relative merits of (deep) CNNs vs. GANs are being tested. Like CNNs, early published applications of GANs to the geosciences have included the automation seismic interpretation and feature/geobody extraction, rock physics and digital rock modeling, generating seismic data, geostatistical inversion, and either augmenting or attempting to replace traditional velocity inversion methods such as FWI.

Two Example Themes from SEG 2020

Rather than provide an exhaustive technical summary of all the related presentations at SEG 2020, I decided to highlight a few interesting aspects of Deep Learning applications. One of the notable challenges to training deep neural networks in geoscience is the frequent lack of real data for training. Several presentations demonstrated how to robustly **generate and apply synthetic data to augment network training**. For example, Fraunhofer ITWM used synthetic seismic gathers to train a U-Net for automated trim statics and demultiple of post-migration data; Total created synthetic data from pseudo-randomly-generated velocity and resistivity models, and trained a network to jointly invert (2D) seismic and electromagnetic data for reconstructing salt bodies; Chevron created a global AVA database from their well data, and trained models to predict lithology, porosity and fluid type from selected sets of features or seismic attributes; the University of Science and Technology of China randomly generated subsurface models, generated synthetic data, and trained CNNs to interpret 3D meandering channel features; Stanford University simulated facies from an object-based geostatistical model and forward model seismic data to train a seismic inversion solution for reservoir facies; Shell generated synthetic data using 4D reservoir and geomechanical models, and trained ML models for each producing field of interest to estimated 4D timeshifts; and KAUST generated a large dataset of random subsurface models to train a CNN for velocity model building.

On the topic of **velocity model building**, full waveform inversion (FWI) was the second-most popular topic at SEG 2020, and several ML presentations investigated ideas to augment aspects of the traditional FWI workflow. For example, FWI benefits from very low frequency (< 3Hz) input data to obtain initial model updates without cycle-skipping challenges, but such frequencies are often not available in the recorded data. Polytechnique Montreal used BP synthetic model data to demonstrate the application of a recursive CNN (RNN) to denoise and generate artificial low frequency data from high frequency data. Each pass of the network halves the frequency content of a gather (refer to **Figure 2**), and the dominant frequency can be lowered by a factor of 64 with relative stability.

Note that the RNN abbreviation is ambiguously used to denote both recurrent and recursive neural networks. Recurrent neural networks are particularly suited to sequential tasks in seismic imaging such as time series analysis. A recursive neural network is more like a hierarchical network where there is really no explicit sequential aspect to the input sequence but the input has to be processed hierarchically in a tree fashion.

In other efforts to extrapolate low frequencies, Schlumberger generated training data by forward modeling with two different (lower and higher frequency) wavelets, and trained a CNN to build a relation between high and low frequency data. A CNN-predicted low frequency data were then used to invert a low wavenumber model using FWI, and this initial estimate was used in an iterative workflow to help mitigate cycle-skipping effects when low frequencies were unavailable in the real data. Advanced Geophysical Technology similarly described a self-learning method to exploit underlying physical relations between high and low frequency components of the data, and thereby extended the bandwidth of real data to lower frequencies. Equinor trained a U-Net on real seismic data with relatively high frequencies labeled and without the relatively low frequency components. The trained network was applied to down-sampled data to generate lower frequency components. Application to real data extended the low frequencies from about 5 to 2.5 Hz. To complete this brief cross-section of alternate approaches, MIT used synthetic Marmousi model data to pursue low-frequency extrapolation of multi-component data as the input to elastic FWI. By training a CNN twice, once with a dataset of horizontal components and once with a dataset of vertical components, they improved the extrapolation of the low frequencies in contrast to only using an acoustic training dataset. At a more sophisticated level, other presentations also sought to augment the way the FWI objective function is computed, or improve the rate of convergence, but this level of pursuit is not considered here.

Some authors also contemplated how to extend the target domain from velocity to the full RTM (reverse time migration) image, and although such studies remain conceptual, long term ambitions include real-time processing, fast-track generation of 'migrated' products, rapid salt model scenario testing, and so on.

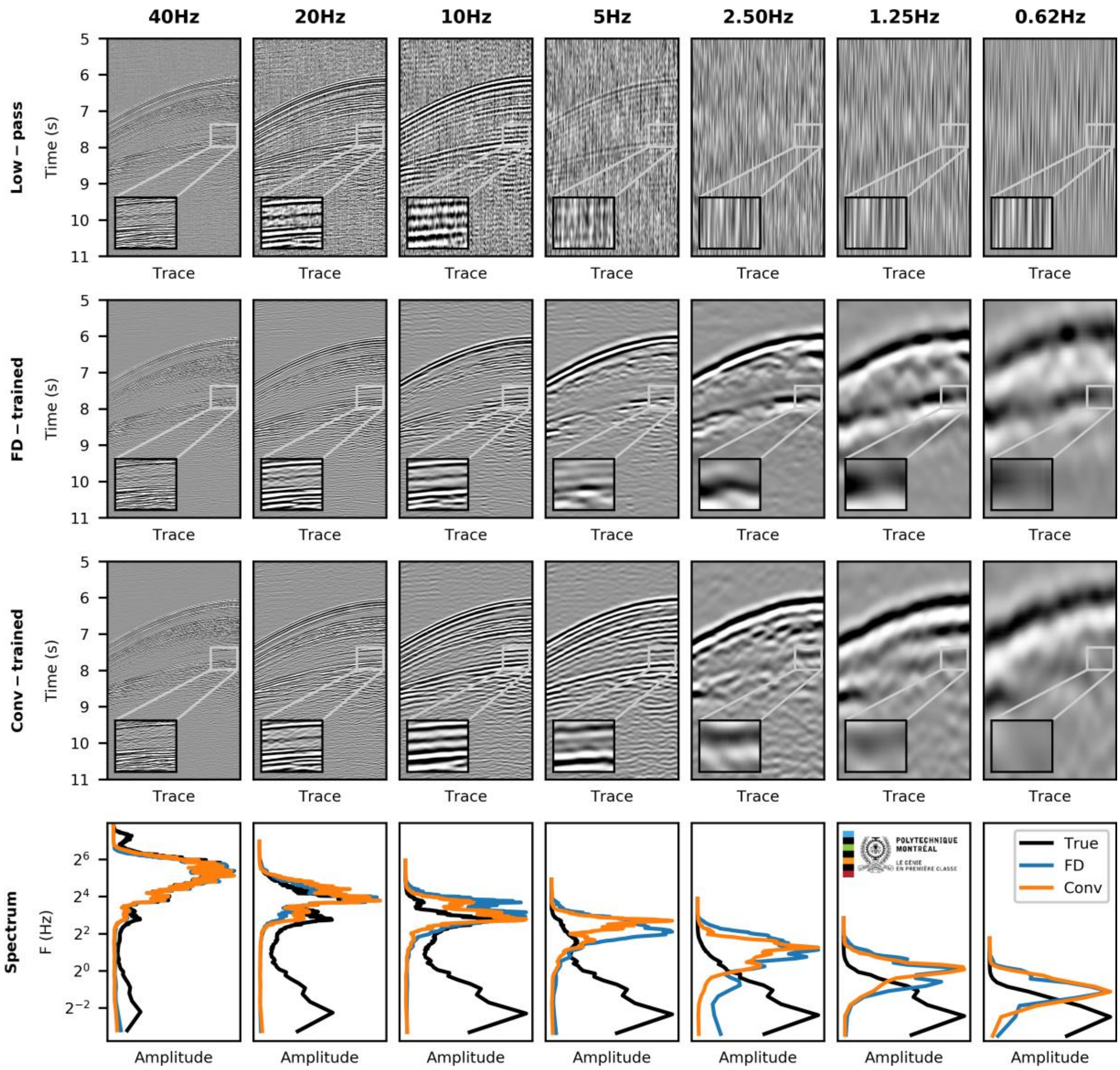

Figure 2. Low frequency predictions for the 2004 BP benchmark model, courtesy of Gabriel Fabien-Ouellet, Polytechnique Montréal. The first row shows the labeled (real) data after low-pass filtering, and the second and third rows show the predictions from two versions of a CNN, respectively. The fourth row shows the mean normalized amplitude spectra. Each pass of the network halves the lowest useful frequency content of a gather.

Summary

In summary, the applications of Deep Learning and other forms of Machine Learning (ML) to the geosciences are increasingly diverse, but commercial solutions are generally confined to various forms of data interpretation and characterization. Nevertheless, it is clear that significant global research efforts are helping ML become both useful and flexible components of seismic imaging, expanding our toolbox of solutions. While a high-profile ambition has been to automate and replace human contributions, thereby greatly accelerating the delivery of products to assist decisions, the bigger picture also includes the mitigation of risk and uncertainty. The contribution of ML to geoscientists is therefore to make **better informed decisions**, **using more (all) data**, and **in less time**.

Further Reading Material

- Deep Learning by Goodfellow, I., Bengio, Y., and Courville, A., 2016, MIT Press.
- Leveraging Deep Learning for Seismic Image Denoising by Klochikhina, E., Crawley, S., Frolov, S., Chemingui, N., and Martin, T., 2020, First Break.