First Break Automatic Picking with Deep Semi-Supervised Learning Neural Network

Kuo Chun Tsai, University of Houston; Wenyi Hu, Advanced Geophysical Technology, Inc;*
Xuqing Wu, Jiefu Chen, Zhu Han, University of Houston

SUMMARY

Although seismic industry has been investigating decades on solving the first break picking problems automatically, there are still enormous challenges during the investigation. Even till today, there are not solid solutions to avoid human labors to manually pick data by geophysicists. With the raise of deep learning and powerful hardware, many of those challenges can be overcome. In this work, we propose a deep semi-supervised neural network to achieve automatic picking for the first break in seismic data. The network is designed to perform with both unlabeled data and a limited amount of real data with labels. Initial feature representation is learning in a discriminative unsupervised manner on real datasets without labels. Since no assumptions are made with regard to the difference of underlying distributions between the synthetic and real data, our model has more marginal gain to compensate for the distribution drifting compare to the supervised learning models. In addition, the network is capable of updating itself through continuous learning. The system is able to identify labeling anomalies onsite and update the model through active learning. In simulation, we show our proposed deep semi-supervised neural network can achieve high accuracy on first break picking. Comparing with the supervised neural networks, our proposed network shows the advantage on using both labeled and unlabeled data set to achieve higher accuracy.

INTRODUCTION

Seismic imaging is a difficult inverse problem, requiring initialization from prior knowledge and extensive manual interventions by domain experts, in a process that can take months to complete. As the seismic industry moves toward denser source and receiver coverage, high-density data requires more efficient computing tools (AlTheyab et al., 2016). Seismic processing technology has for the most part been able to keep up with these data volume increase as the capacity and the speed of the computer hardware are rapidly improving. However, there are still some areas in seismic data processing which remain labor intensive and have not fully exploited latest advances in computer technology.

The first break picking or so-called first-arrival picking is the task of determining, given a set of seismic traces, the onsets of the first signal arrivals that are associated with the leading energy pulse received by a receiver in response to a seismic shot (Sabbione et al., 2010). The accurate determination of the location of the first-arrivals is the first step toward high quality velocity model building and depth imaging in seismic data processing. Although critical for detecting anomalies, removing inconsistencies, and setting initial values based on previous geological studies, manual picking and pattern identification jobs that depend on visual inspection are labor intensive and typically take up 20% to 50% of the total seismic processing time (McCormack et al., 1993). Seismic data volumes have grown exponentially over the last three decades as the seismic exploration industry increases the coverage of the survey. Various automatic picking approaches exist with varying degrees of success (Kahrizi et al., 2014). Nonetheless, the uncertainty involved in these tools is still too high to replace heavy human efforts in visual inspection and picking.

Nowadays, deep learning is a popular technique in many fields where the deep neural networks are able to solve many complex problems. In short, deep learning is an extension from machine learning which is inspired by the combination of the mysterious on information processing between human brain neurons and the robust calculation ability of the computers. This brilliant idea was proposed by Warren McCulloch and Walter Pitts in 1943. Unfortunately, this concept was not been well recognized till late 20th century due to the computer hardware developing bottleneck where the hardware were unable to handle the massive calculation of the learning process. With the rapid development of the computer hardware started in late 20th century, engineers has developed multi-cores CPUs and GPUs. In 2016, Google announced the tensor processing unit (TPU) which is the chip that specifically designed for the Google's TensorFlow framework. According to Google, TPU provides 71X performance compared with the CPU for CNN1 application (Jouppi et al., 2017).

Despite the fact that artificial intelligent was introduced into the seismic exploration industry in the early 1990s' (Kahrizi et al., 2014) and various picking software packages have been developed to tackle these problems, the industry still relies on human experts and treats those software approaches as assisting tools. Existing picking tools used by seismic data processing jobs fail to deliver consistent results on various datasets and require frequent human intervention and parameter tuning. One of the reasons is due to the shallow learning structure used by current approaches is unable to absorb a large amount of data and underestimate the variance of the distribution of the data. Unlike its shallow predecessors, in order to develop a machine vision enabled picking and identification tool to replace human vision inspection for processing the seismic data, the deep learning network requires a huge amount of training data with labels. However, data collection in the seismic exploration industry is a quite challenging task since most of the sources are proprietary data and processing results are kept as top business secrets.

In our paper, we propose a deep semi-supervised neural network (DSSNN) model for automatically picking the first break in seismic data which can reduce the human labor for the seismic data processing. The model is designed to perform with a limited amount of real data with labels which is perfectly fit in the scenario for the seismic data lacking of labels. The proposed network is capable of updating itself through continuous

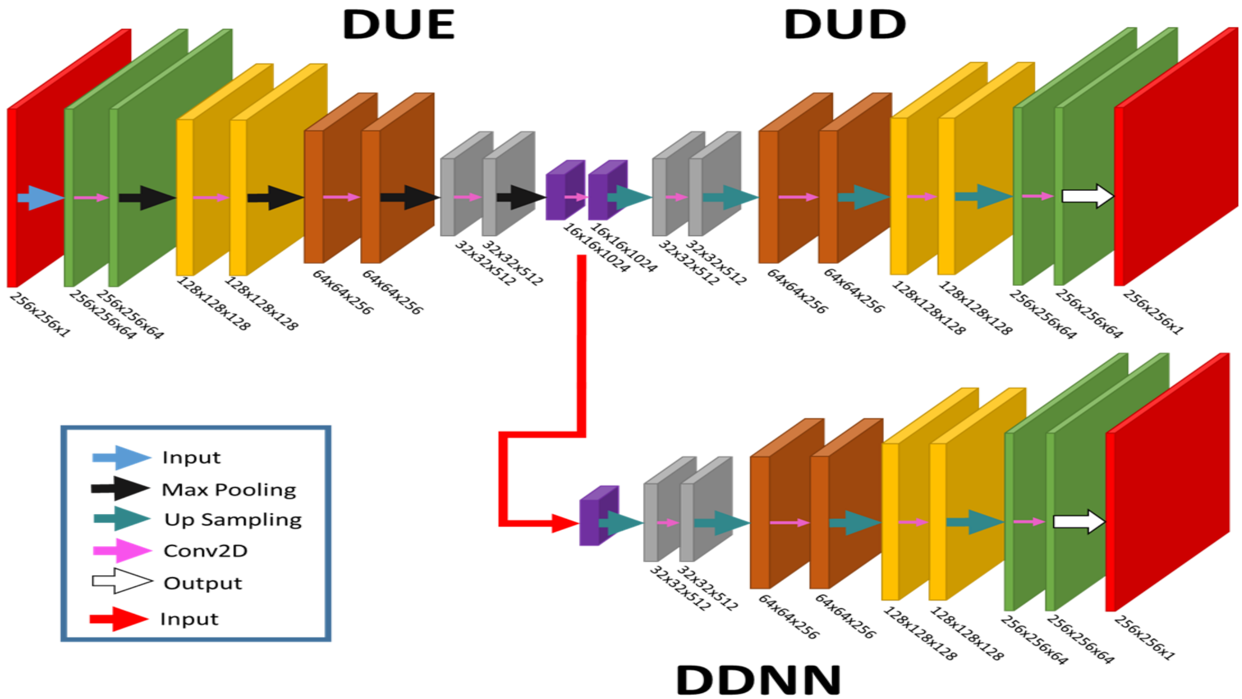


Figure 1: An overview of DSSNN model where it is constructed by the DUE, the DUD, and the DDNN.

learning from both labeled and unlabeled data. The system is able to identify labeling anomalies on-site and update the model through active learning.

The rest of the paper is organized as follows. We first discuss the details of our proposed method which includes the pre-processing process of the data and two deep neural networks within our model. Second, we give our simulation steps and the results between different neural networks. Finally, we will be concluding our work.

METHOD

Machine learning can break into three different types learning – supervised learning, unsupervised learning, and semi-supervised learning. First, supervised learning is where the model is given an input data **x** and a corresponding target output Y. In the area of machine learning, we usually refer Y as the label of **x**. The model will learn through the error between the predict label Y' and target label Y. For the second type, unsupervised learning model will learn from input **x** without corresponding label. This type of learning model is usually used to solve clustering or association problems. Finally, semi-supervised learning is the combination of supervised learning and unsupervised learning where only some of the inputs are labeled. The model has to learn from both labeled data and unlabeled data using both supervised learning and unsupervised learning techniques.

In this section, we will first discuss the overall picture of our DSSNN model since the structure of the model contains two different neural networks as mentioned before. Second, we shows the necessary data preprocessing steps for the input data before feeding into the DSSNN. In the third part, we start to discuss the structure of unsupervised neural network in our DSSNN and the training process for the network. Next, we will present the DSSNN model and the training process. Finally, we reconstruct the predict data from the DSSNN since we will be modifying the data during the data preprocessing process.

Deep Semi-supervised Neural Network Overview

Our DSSNN model is based on the idea of the autoencoder model where the autoencoder can be split into encoder and decoder parts and each part only contains one layer of the network. Different from the traditional autoencoder, not only our DSSNN has multiple layers which is more robust than the basic autoencoder on feature extraction but also has the ability of data label prediction. The DSSNN is constructed by one deep unsupervised autoencoder (DUA) and one deep deconvolutional neural network (DDNN). In addition, DUA is formed by one deep unsupervised encoder (DUE) and one deep unsupervised decoder (DUD) (see Figure 1). We will first train the DUA for feature extraction purpose where the extracted feature by the DUE will be the input of the DDNN. The training detail will be discussed in the later sections.

Data Preprocessing

The seismic data for each shot is constructed by the signals received by each receiver throughout the time steps. Therefore, each shot can be represented by a matrix $\mathbf{S_i}$ where each column represents the signals received by each receiver and the rows are corresponds to the time steps. In order to reduce data redundancy, we preform data normalization by mapping each

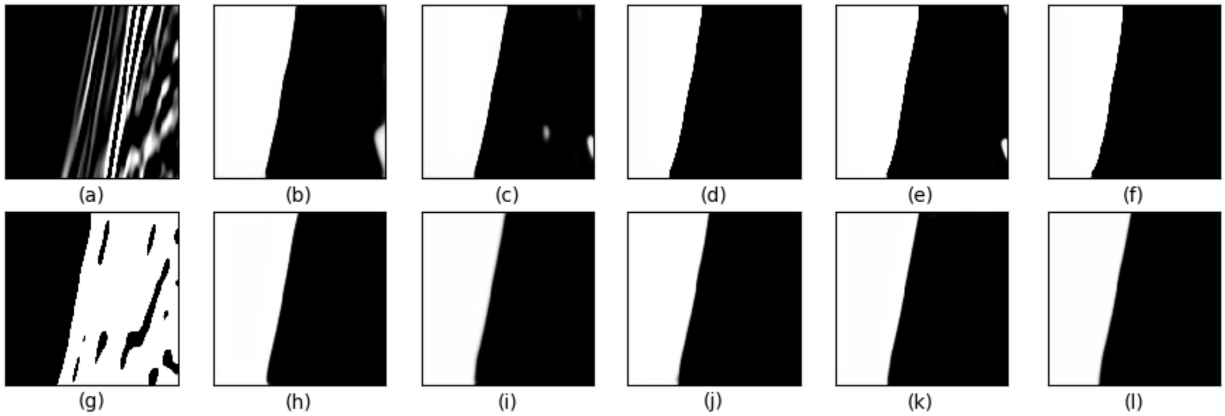


Figure 2: Networks input and output data. (a) is the input data for both DSNN and DSSNN. (g) is the output of the DUA. The predicted outputs of the DSNN [(b), (c), (d), (e), and (f)] with respect to the training sample size of 100 to 500 where the black area represents the earth. Similarly, (h), (i), (j), (k), and (l) are the outputs of the DSSNN with training sample size of 100 to 500.

elements in matrix $\mathbf{S_i}$ into the range of 0 to 255, i.e.,

$$\mathbf{S'}_i = \begin{cases} \frac{255}{max-min} \cdot (\mathbf{S}_i + min), & \text{if } max > min, \\ 0, & \text{otherwise.} \end{cases} \quad (1)$$

where max is the maximum value of $\mathbf{S_i}$, min is the minimum value of $\mathbf{S_i}$, and $\mathbf{S'}_i$ is the normalized matrix. Since each shot may contains thousands of millions of signals, it is not likely to input $\mathbf{S'}_i$ as one piece of data into the neural network due to the hardware memory size limitation of modern technologies. Thus, we crop each $\mathbf{S'}_i$ into small matrices $\mathbf{s}^i_k$, i.e., $\mathbf{S'}_i = \{\mathbf{s}^i_0, ..., \mathbf{s}^i_n\}$ where n is the total number of cropped matrices of each shot.

For some $\mathbf{S_i}$, there is a corresponding ground truth binary matrix $\mathbf{G}_i$ where the elements in $\mathbf{G}_i$ are been split into two groups. In $\mathbf{G}_i$, all the elements below the first break line have the value of 0 and the rest elements will have the value of 1. As we mentioned before of the cropping process for the matrix $\mathbf{S'}_i$, we perform the same task for $\mathbf{G}_i$ where $\mathbf{G}_i = \{\mathbf{g}^i_0, ..., \mathbf{g}^i_n\}$. It is very important to keep in mind that matrix $\mathbf{G}_i$ has to be cropped exactly the same way as the corresponding shot matrix $\mathbf{S'}_i$ since we will use $\mathbf{g}^i_k$ as the label of $\mathbf{s}^i_k$ for our DDNN during the supervised learning process.

Unsupervised Neural Network

In our DSSNN, the DUA is trained on unlabeled data. In more specific, we use the input data as the label of itself to train the DUA. As we mentioned previously, DUA is constructed by DUE and DUD. The DUE has the structure of the deep convolutional neural network and the DUA has the structure of the deep deconvolutional neural network. With the label been the same as the input data, the DUE is able to determine the most significant features of the input data and produce a compressed representation of those features (Q. V. Le, 2013). The idea in back of DUA is that since the target label is the input data itself, the DUE must extracts some features that is significant enough for the DUD to recognize in order to construct the output that is closed to the input data.

Supervised Neural Network

For the supervised learning part, we designed a supervised DDNN where the input depends on the output of the DUE that we discussed in the previous section. In other words, we extract the DUE from the DUA with fixed weights of the DUA after training. The DUE acts as the data generator for the DDNN where $\mathbf{s}^i_k$ is feed into DUE and the DUE will generate the compressed feature representation $\mathbf{r}^i_k$. Here is where the ground truth of the first break matrix kicked in. Since $\mathbf{g}^i_k$ is the label of $\mathbf{s}^i_k$, it is also the label of $\mathbf{r}^i_k$. With this concept, the DDNN will be learning from the input $\mathbf{r}^i_k$ and tried to construct the output matrix $\mathbf{y}^i_k$ that is as close as the target label $\mathbf{g}^i_k$.

Data Reassemble

In the data pre-processing section, we cropped $\mathbf{S'}$ into small pieces. Thus, we will need to reassemble the outputs from the DSSNN back to its original size. Before reassemble all the predicted matrices $\mathbf{y}^i_k$ back to its associate area, we will first find the first break line in each of the matrices. To do so, for each matrix $\mathbf{y}^i_k$, we perform the column wise outlier removal process. The elements in the column c that are predicted as earth will be placed in an array with the value of its row index in the column c. We then remove the elements that are two standard deviations away from the mean according to Gaussian distribution. Once the outliers are removed, we get the minimum value of the array as the row index of column c. We replace value of this row in column c to 1 and the rest of the rows in column c will be set 0. In the special case where the entire column has the same value, we will replace the entire column to zeros. Once the above process is completed for all $\mathbf{y}^i_k$, we then construct a matrix $\mathbf{Y}_i$ with the size of original shot matrix $\mathbf{S}_i$ filled with zeros. The way we reassemble the predicted matrices $\mathbf{y}^i_k$ is perform the elements wise addition according the cropped area of the input matrix $\mathbf{s}^i_k$ on $\mathbf{Y}_i$. Once addition process is done, we then perform column wise filtering on $\mathbf{Y}_i$ where we find the element of the maximum value of the column with the smallest row index number and replace it by 1. The rest of the elements in that column will be padded to 0. At the end, matrix $\mathbf{Y}_i$ will have at most one value in each column which is the first break of the corresponding receiver.

EXAMPLES

In our simulation, a synthetic seismic dataset is adopted to illustrate the performances of the neural networks. We will be comparing the result for each neural networks which includes the proposed DSSNNs and the deep supervised neural network (DSNN). For both of the networks, we adjust the size of the training data set from 100 samples to 500 samples with step size of 100 samples. The DSNN has the same structure as the DUA in the DSSNN and since it is a supervised learning model, the training data set will have a corresponding label set. In the simulation, we will be using the GPU to accelerate the training speed. Due to the hardware memory limitation on the GPU, we will input a batch size of 20 samples at once for training the neural networks. Each batch will run 50 epochs before input another set of samples.

In our simulation, the cropped input data (Figure 2 (a)) will go through the DUA and the output (Figure 2 (g)) is the compressed feature representation output from the DUE. As the result, Figure 2 (h), (i), (j), (k), and (l) show the predictions from the DSSNN with training sample size of 100, 200, 300, 400, and 500, respectively. The results from the DSNN are shown in Figure 2 (b), (c), (d), (e), and (f) as well. We can clearly see that the results of the DSSNN is better than results of the DSNN. From Figure 3, we can see that the DSSNN has higher accuracy for each test on different number of training sample. This shows that by using the semi-supervised learning, we are able to overcome the challenge of the limitation on the number of the labeled data. Keep in mind that Figure 2 only shows the preprocessed input data but not the entire final shot prediction. Finally, we reassembled the predicted outputs from the DSSNN and the DSNN and shows the final outputs in Figure 4 where the red line represents the ground truth of the first break line and the light blue circles represents the prediction of the first break line. As we can see, there is a huge gap between the prediction and the ground truth for the DSNN (Figur 4 (a)) in the left of the figure. On the other hand, the prediction from the DSSNN (Figure 4 (b)) is almost identical to the ground truth.

CONCLUSION

In this paper, we have proposed the DSSNN model to solve the labeled data limitation problem in the first break picking process. With the limited labeled seismic data and the huge amount of unlabeled data, we are able to use the unlabeled data to compress the significant features into another representation. By training the model with the small amount of labeled data for training the neural network, the DSSNN model is able to determine the first break from the input data. Simulation results have shown that our proposed DSSNN model has as significant higher effectiveness than other approaches in literature.

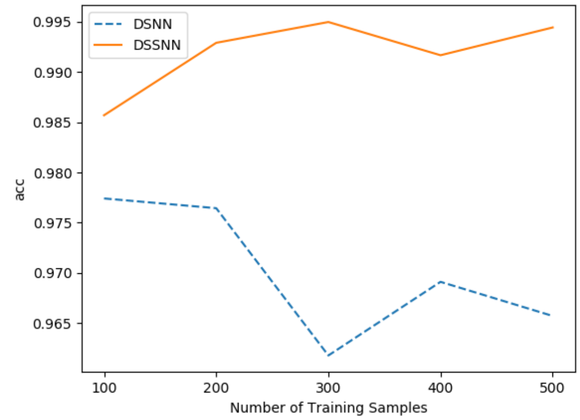
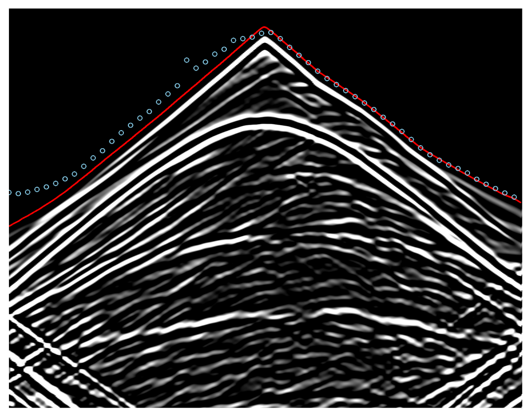
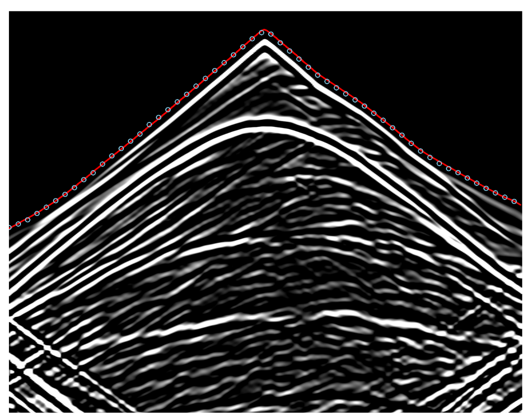


Figure 3: Accuracies of DSNN and DSSNN with training sample size of 100 to 500.



(a)



(b)

Figure 4: First break picking results of the DSNN (a) and the DSSNN (b) where the red line indicates the ground truth of the first break and the light blue circles indicates the predicted first break.

ACKNOWLEDGMENTS

This material is based upon work supported by National Science Foundation under Grant No. 1746824.