IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received April 13, 2019, accepted May 13, 2019, date of publication May 22, 2019, date of current version June 5, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2918366

Semi-Supervised Learning Based on Generative Adversarial Network and Its Applied to Lithology Recognition

GUOHE LI[1,2,3], YINGHAN QIAO[1,2,3], YIFENG ZHENG[1,2,3,4,5], YING LI[1,2,3], AND WEIJIANG WU[1,2,3]

[1]Key Laboratory of Petroleum Data Mining, China University of Petroleum, Beijing 102249, China
[2]College of Information Science and Engineering, China University of Petroleum, Beijing 102249, China
[3]Laboratory of Oil and Gas Big Data, China University of Petroleum Beijing at Kelamayi, Xinjiang 834000, China
[4]School of Computer Science, Minnan Normal University, Zhangzhou 363000, China
[5]Key Laboratory of Data Science and Intelligence Applications, Fujian Province University, Zhangzhou 363000, China

Corresponding author: Yifeng Zheng (zyf@mnnu.edu.cn)

This work was supported in part by the Nature Science Foundation of China under Grant 60473125 and Grant 61701213, in part by the Innovation Foundation of CNPC under Grant 05E7013, in part by the National Key Project Foundation of Science under Grant G5800-08-ZS-WX, in part by the Science Foundation of China University of Petroleum-Beijing At Karamay under Grant RCYJ2016B-03-001, in part by the Cooperative Education Project of Nation Education Ministry under Grant 201702098015, in part by the Fujian Province Natural Science Funds under Grant 2018J01545, Grant 2018J01546, and Grant 2019J01748, and in part by the Research Funds for the Educational Department of Fujian Province under Grant JA15300.

ABSTRACT Lithology recognition is an essential part of reservoir parameter prediction. Compared to conventional algorithms, deep learning that needs a large amount of training data as support can extract features automatically. In the process of real data acquisition, the labeled data account for only a small portion due to high drilling cost, and it is difficult to achieve the data size required for deep learning training, resulting in a significant variance of the recognition model. In this paper, for this shortage, a semi-supervised algorithm based on generative adversarial network (GAN) with Gini-regularization is proposed, called SGAN_G, which takes borehole-side data as labeled data and seismic data as unlabeled data. First, the SGAN_G is trained by Adam (a method for stochastic optimization) algorithm and utilizes a discriminator to lithology recognition. And, we add the entropy regularization to the initial loss function which enhances the convergence speed and accuracy of the model. Eventually, we propose a novel sampling approach which employs multiple sampling points of seismic data as inputs to use the stratum information implicitly. Through the experimental comparison with a variety of supervised approaches, we can see that the SGAN_G can achieve higher prediction accuracy by using unlabeled data effectively.

INDEX TERMS Entropy regularization, generative adversarial network, lithology recognition, semi-supervised learning.

I. INTRODUCTION

Geophysical exploration is an essential part to the development and production of petroleum. It can help people understand the stratigraphic structure, lithology, and oil-bearing properties, as well as porosity by seismic data analysis through manual seismic, seismic wave detection and seismic data processing [1]. Oil and gas reservoir prediction is a critical component of geophysical exploration and is essential for assessing oil and gas reserves [2]. Its main tasks involve the use of seismic data for modeling and recognition of stratigraphy lithology (i.e., seismic inversion). Due to the anisotropy of the stratum and the complexity of the stratum caused by stratigraphic structure, lithology, interspace, fluids, etc., there is a complex nonlinear relationship between seismic waves (i.e., seismic-data) and stratigraphic parameters (including physical properties, lithology, oil-bearing properties, etc.). Therefore, it is complicated to establish inversion models between the seismic data and lithology. In addition to traditional seismic inversion, artificial intelligence

The associate editor coordinating the review of this manuscript and approving it for publication was Adam Czajka.

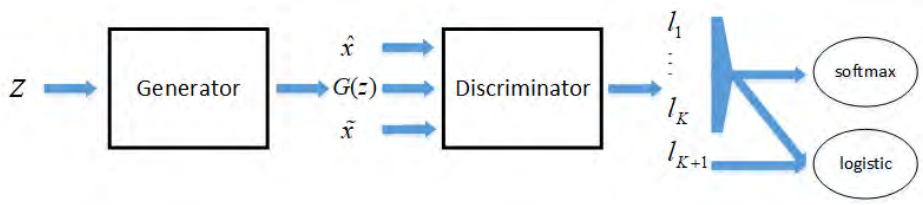


FIGURE 1. Semi-supervised learning based on GAN. The discriminator D outputs a $K+1$ dimensional vector L. The first K elements represent the score of the input belonging to each category. The last one represent the score of the input belonging to $G(z)$.

techniques are used to lithology recognition such as Random Forest [3]–[5], Deep Neural Network(DNN) [6]–[10] and Convolutional Neural Network(CNN) [11]–[14] etc. With the rapid development of technology, deep learning utilizes in seismic lithology inversion, such as lithology recognition based on deep belief network (DBN) [15]–[17]. Lithology recognition is a common task found in the petroleum exploration filed, which is actually a problem of classifying rock types, based on labeled data obtained from well drilling programs [18]. It has gained enormous benefits from the significant progress of intelligent modeling methods. Unfortunately, there are still three problems: (1) the model is prone to over-fitting when facing classification problem of strong noise data, such as RandomForest; (2)the model is not robust, such as neural network with random initial values will fall into local optimum easily during training; (3) there are only a few labeled data due to the expensive cost of drilling, and the number of different classes is incredibly imbalanced. Although it can be alleviated by training the network layer-by-layer with contrast divergence algorithm [19], [20] or adopting regularization methods, such as $L1$ or $L2$-regularization, Dropout [21] and Batchnormalization [22], it is still hard to improve the generalization ability of the model further under the limitation of supervised learning mode when the amount of labeled data is small and imbalanced. The above problems result in substantial error and wrong tendency to the majority class when utilizing the model to lithology recognition. In this paper, a semi-supervised algorithm based on GAN(Generative Adversarial Network) with Gini-regularization is proposed, called SGAN_G, which takes borehole-side data as labeled data and seismic data as unlabeled data. First, SGAN_G is trained by Adam(a method for stochastic optimization) algorithm [23]. And then we utilize the discriminator for lithology recognition. Finally, we employ multi-sampling points of seismic data as inputs to use the stratum information implicitly. The rest of the paper is organized as follows. The basics of GAN is introduced in Section II. The proposed novel algorithm is presented in Section III. Experimental results and evaluations are given in Section IV. Finally, the conclusion is given in Section V

II. PRELIMINARY KNOWLEDGE

In 2014, Goodfellow presented Generative Adversarial Network(GAN) [24] which learns by a game between two multi-layer networks, called the generator network G and discriminator network D. Given a generator network G with parameters $\theta(G)$ and a discriminator network D with parameters $\theta(D)$. Let z denotes random vectors sampled from some simple noise distribution $p_z(z)$(e.g., the uniform distribution or Gaussian distribution). G represents a map from the space of z to the space of data(like images): $G: G(z;\theta(G)) \rightarrow R^{|x|}$, where x is an image and $|\cdot|$ denotes the number of dimensions [25].

Then D maps from x to a scalar: $D: D(x;\theta(D)) \rightarrow (0, 1)$, which indicates the probability of x from the real data distribution $p_{data}(x)$, rather than the generator distribution $p_G(x)$. D is utilized to classify the input x as being from real data(close to 1) or from G(close to 0), and G is employed to deceive D into misclassifying its output as actual data. G and D alternately train their parameters based on game theory principles respectively, and finally reach a optimal-point—the generator distribution $p_G(x)$ matches the actual distribution $p_{data}(x)$ and D cannot distinguish the inputs from G. In other words, it can be summarized as a mini-max problem, with a loss function defined as follow:

$$\min_{\theta(G)}\max_{\theta(D)} L(D, G) = E_{x\sim p_{data}(x)} logD(x) + E_{z\sim p_z(z)} log(1 - D(G(z))) \quad (1)$$

III. SEMI-SUPERVISED LEARNING BASED ON GAN

A. MODEL PRINCIPLE

In 2016, Salimans and Goodfellow et al. proposed Semi-supervised Learning based on GAN (SGAN) [26], [27]. In the multi-classification problem, for a sample x, its outputs can express as a K-dimensional vector $L = \langle l_1, l_2, \cdots, l_k \rangle$, where l_k represents the score belonging to each category. To use unlabeled samples for training, the discriminator network D of SGAN solves a multi-classification task, instead of a two-category task. Since using the fake data generated by the generator network G, the discriminator network D also needs to judge whether the input is from G in addition to the category of the input. Therefore, the number of categories in the classification problem increase to $K+1$. The whole process is shown in FIGURE 1. Note that the discriminator network D includes three inputs: (1) the labeled data $\hat{x}$ from the actual dataset, (2) the unlabeled data $\tilde{x}$ from the real dataset, and (3) the fake data $G(z)$ from the generator network G.

In SGAN, the loss function for training the discriminator network D is redefined as follows:

$$L(D) = L(D\_supervised) + L(D\_unsupervised) \quad (2)$$

$$L(D\_supervised) = -E_{\hat{x},y\sim p_{data}(\hat{x},y)} log p_D\left(y|\hat{x}, y < K+1\right) \quad (3)$$

$$L(D\_unsupervised) = -E_{\tilde{x}\sim p_{data}(\tilde{x})} log\left[1 - p_D(y = K+1|\tilde{x})\right] - E_{z\sim p_z(z)} p_D\left(y = K+1|G\left(z\right)\right) \quad (4)$$

where $L\left(D\_supervised\right)$ denotes standard supervised cross-entropy loss function, $L\left(D\_unsupervised\right)$ represents the unsupervised loss function which is the objective function of conventional GAN.

To better approximate the distribution of actual data, the loss function for training the generator network G is defined as follow:

$$L\left(G\right) = \left\| E_{\tilde{x}\sim p_{data}(\tilde{x})} f\left(\tilde{x}\right) - E_{z\sim p_z(z)} f\left(G\left(z\right)\right) \right\|_2^2 \quad (5)$$

where $f\left(\cdot\right)$ denotes the output of the feature layer(the layer before $\langle l_1, l_2, \cdots, l_k, l_{k+1}\rangle$), and $\|\cdot\|_2^2$ represents the $L2-norm$.

B. PRINCIPLE VISUALIZATION

To illustrate the mechanism of SGAN carefully, we do experiments on the simulated dataset to visualize the process of model training. As shown in FIGURE 2, the outer circle of the dataset is consist of positive samples, and the inner circle is consist of negative samples (each circle contains 1000 samples). To reflect the characterization of semi-supervised learning, 50 samples are selected from the left (right) part of the positive (negative) samples as labeled data randomly, and the remaining are unlabeled data.

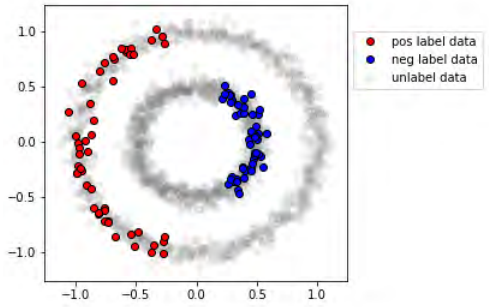


FIGURE 2. Toy dataset: positive and negative samples are distributed in the outer and inner circle respectively. Red and blue points are labeled data. The others are unlabeled.

For comparison, we train a Deep Neural Network(DNN) on the labeled data firstly. DNN consists of feature extraction layers(4×2) and a classifier layer (logistic regression). At first, the model maps the data to feature space through the feature extraction layers. And then, it uses the classifier to classify the data in the feature space. FIGURE 3(a) shows the distribution of data in the feature space of the DNN during training. It can be seen that the model has converged after 25 epochs (an epoch means one iteration over the entire input data). From the figure, the result illustrates that its generalization ability is too bad, especially in the unlabeled data. The error rate of the model on the unlabeled data is about 50%.

As mentioned above, we train the SGAN model on the labeled and unlabeled data, which consists of a generator network $G(2 \times 2)$ and a discriminator network D same as the DNN above. The random noise z is a one-dimensional vector sampled from standard normal distribution. The loss function of G can better approximate the distribution of actual data, while the unsupervised part of the discriminator loss function makes their distribution divisible. And the supervised one can establish the segmentation boundary between positive and negative samples in the feature space. As shown in FIGURE 3(b), during the adversarial training process, the fake data wraps positive and negative data respectively and pulls them to the opposite direction. From the result, compared to DNN, its error rate is 12.8% when the model converged.

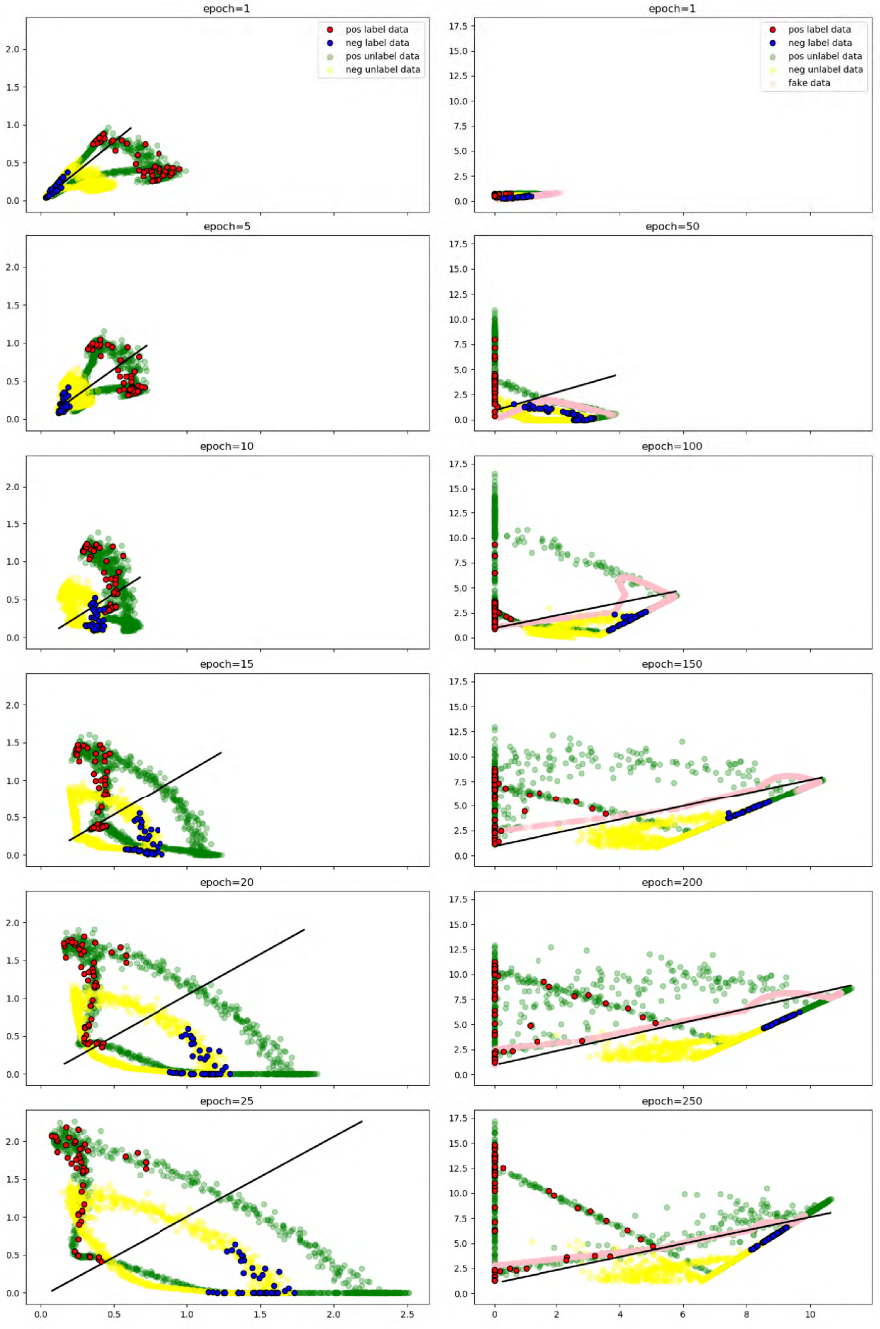
C. ENTROPY REGULARIZATION

In the information theory, entropy represents the uncertainty of a random variable. Assume that $X = \langle x_1, x_2, \cdots, x_n\rangle$ is a discrete random variable with n different values, and its probability distribution can be expressed as $p\left(X = x_i\right) = p_i\left(i = 1, 2, \cdots, n\right)$. Therefore, the entropy of X is defined as follows:

$$Entropy\left(X\right) = -\sum_{i=1}^{n} p_i log p_i$$
$$s.t. Entropy\left(X\right) \in [0, 1] \quad (6)$$

Obviously, the larger the entropy, the more uncertain the random variable is, and vice versa. The output L of the multi-classification model can be viewed as a random variable X that conform to a multi-classification probability distribution, and each output corresponding to an entropy value. The discrimination ability with respect to various types of samples is crucial to measure the performance of classification models. It means that the entropy of the output will be smaller, as shown in FIGURE 4. Therefore, to enhance the effectiveness of SGAN learning process, we improve the unsupervised part of the discriminator loss function by adopting an Entropy-regularization term, which can append additional constraints for the model training on unlabeled data.

When the probability $p \to 0$, $logp_i \to -\infty$, it may cause the overflow in computer calculations. To solve this problem, we utilize Gini-regularization to replace Entropy-regularization. The Gini-regularization(also known as Gini-impurity) is also an information measure which can reflect the uncertainty of random variable distribution. In essence, the Gini-impurity has similar distribution characteristics to Entropy. For $Entropy\left(X\right)$, if we expand the $logp_i$ by



(a) (b)

FIGURE 3. **Distrubution on feature space. The black line denotes the segmentation boundary.**

first-order Taylor series at $p = 1$, the $Gini(X)$ can be approximated to it.

$$\begin{aligned} Entropy(X) &= -\sum_{i=1}^{n} p_i log p_i \\ &= \sum_{i=1}^{n} p_i f(p_i) \\ &\approx \sum_{i=1}^{n} p_i \left[\frac{f(1)}{0!} + \frac{f'(1)}{1!}(p_i - 1) \right] \\ &= 1 - \sum_{i=1}^{n} p_i^2 \\ &= Gini(X) \end{aligned} \tag{7}$$

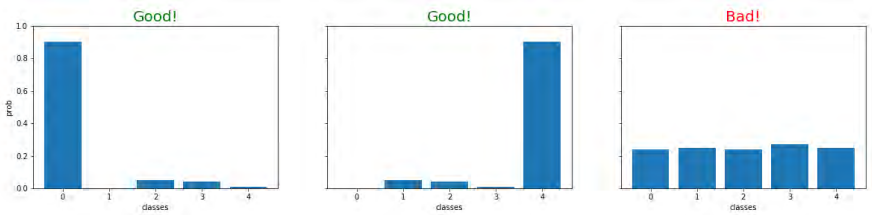


FIGURE 4. Predictions of multiple-class problems.

Algorithm 1 Minibatch Stochastic Gradient Descent Training of SGAN_G

Input: $p_z(z)$: noise prior, $p_{data}(x)$: data distribution, m: batch size, λ: coefficient of Gini_regularization
Output: discriminator D after training

1: **for** number of training iterations **do**
2: Sample minibatch of m noise samples $\left\{z^{(1)}, \cdots, z^{(m)}\right\}$ from $p_z(z)$.
3: Sample minibatch of m labeled examples $\left\{\left(\hat{x}^{(1)}, y^{(1)}\right), \cdots, \left(\hat{x}^{(m)}, y^{(m)}\right)\right\}$ from $p_{data}\left(\hat{x}, y\right)$.
4: Sample minibatch of m unlabeled examples $\left\{\tilde{x}^{(1)}, \cdots, \tilde{x}^{(m)}\right\}$ from $p_{data}(\tilde{x})$.
5: Update the discriminator D by descending its stochastic gradient:

$$\nabla_{\theta(D)} \frac{1}{m} \sum_{i=1}^{m} \left(-log p_D\left(y^{(i)} | \hat{x}^{(i)}, y^{(i)} < K+1\right) - \left[log\left(1 - p_D\left(y = K+1 | \tilde{x}^{(i)}\right)\right) + log p_D\left(y = K+1 | z^{(i)}\right)\right] + \lambda \left[1 - \sum_{k=1}^{K} p_D\left(y = k | \tilde{x}^{(i)}\right)\right]\right)$$

6: Update the generator G by descending its stochastic gradient:

$$\nabla_{\theta(G)} \frac{1}{m} \left\| f\left(\tilde{x}^{(i)}\right) - f\left(G\left(z^{(i)}\right)\right)\right\|_2^2$$

7: **end for**
8: The gradient-based updates can use any standard gradient-based learning algorithm. We used Adam in our experiments.

Therefore, the formula (4) can be redefined as follows:

$$L\left(D\_unsupervised\_with\_gini\_regularization\right) = L\left(D\_unsupervised\right) + \lambda \left[1 - \sum_{k=1}^{K} p_{model}\left(y = k | \tilde{x}\right)\right] \tag{8}$$

where λ denotes the coefficient of regularization term, and it can control the effect of the Gini-regularization term on the loss function, $\lambda \in (0, 1)$.

Therefore, the novel model proposed in this paper is also called SGAN_G (Semi-supervised GAN with Gini-regularization). The detailed process of SGAN_G is introduced as Alogrithm 1.

D. THEORETICAL PROOF

Because of relying on the gradient descent algorithm(SGD) in the training, we analyze the effect of the Gini-regularization from its influence on the gradient. To simplify the representation, we utilize p_j to represent the probability of the jth class:

$$p_j = p_D\left(y = j | \tilde{x}\right) = \frac{exp\left(l_j\right)}{\sum_{k=1}^{K} exp\left(l_k\right)} \tag{9}$$

According to the chain rule, we can obtain the gradient of Gini-regularization respect to the jth element $l_j\ (j = 1, 2, \ldots, K)$ in the L layer:

$$\begin{aligned} \frac{\partial Gini}{\partial l_j} &= \frac{\partial Gini}{\partial p_j} \cdot \frac{\partial p_j}{\partial l_j} + \sum_{k=1, k\neq j}^{K} \frac{\partial Gini}{\partial p_k} \cdot \frac{\partial p_k}{\partial l_j} \\ &= 2p_j^2\left(p_j - 1\right) + \sum_{k=1, k\neq j}^{K} 2p_k^2 p_j \\ &= 2p_j\left(-p_j + \sum_{k=1}^{K} p_k^2\right), \quad \left(2p_j \geq 0\right) \end{aligned} \tag{10}$$

where $-p_j + \sum_{k=1}^{K} p_k^2$ determines the direction of the gradient, can be expressed as $g = -p_j + \sum_{k=1}^{K} p_k^2$.

Theorem 1: Given p_j, under the constraint of $\sum_{k=1,k\neq j}^{K} = 1 - p_j$, for g, the lower bound can be expressed as $G\left(p_j\right) = inf\left(g\right) = \frac{\left(Kp_j - 1\right)\left(p_j - 1\right)}{K-1}$, and the upper bound is $H\left(p_j\right) = sup\left(g\right) = \left(2p_j - 1\right)\left(p_j - 1\right)$.

Proof: We introduce a Lagrangian multiplier β and study the Lagrange function as follows:

$$\mathcal{L}\left(p_1, \cdots, p_{j-1}, p_{j+1}, \cdots, p_K, \beta\right) = g + \beta\left(p_j - 1 + \sum_{k=1,k\neq j}^{K} p_k\right) \quad (11)$$

The first-order partial derivatives of each element in $\mathcal{L}$ can be calculated as follows:

$$\frac{\partial \mathcal{L}}{\partial p_i} = 2p_i + \beta \quad (12)$$

$$\frac{\partial \mathcal{L}}{\partial \beta} = -1 + \sum_{k=1}^{K} p_k \quad (13)$$

where $i = 1, \cdots, j-1, j+1, \cdots, K$.

Let (12) and (13) equal to zero, we can obtain the local extreme value of g, $G\left(p_j\right) = \frac{(Kp_j-1)(p_j-1)}{K-1}$, when $p_j = \frac{1-p_j}{K-1}$. $G\left(p_j\right)$ is the minimum value of g due to the Hessian matrix of $\mathcal{L}$ is a positive definite matrix. Since $0 \leq p_i \leq 1 - p_j$, g can take the maximum value $H\left(p_j\right) = \left(2p_j - 1\right)\left(p_j - 1\right)$ at the edge of the domain(i.e., one of the p_i equals to $1 - p_j$ and the others equal to zero). ■

Therefore, the gradient of Gini-regularization with respect to l_j is between $2p_j \cdot G(p_j)$ and $2p_j \cdot H(p_j)$, and their function curves are shown in FIGURE 5.

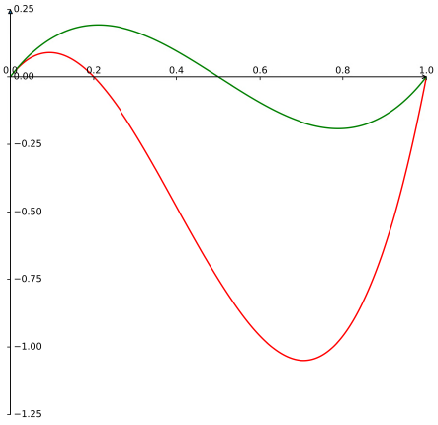


FIGURE 5. Function curves. The red line represent the lower bound of $\frac{\partial Gini}{\partial l_j}$, i.e., $2p_j \cdot G(p_j)$, when $K = 5$ and the green line represent the upper bound $2p_j \cdot H(p_j)$.

The ratio of $\frac{\partial Gini}{\partial l_j}$ is depend on the probability distribution of the rest $K - 1$ classes. When the distribution is balanced or concentrated to a certain class, $\frac{\partial Gini}{\partial l_j}$ achieves the lower bound or upper bound respectively. Once the distribution is determined, $\frac{\partial Gini}{\partial l_j}$ will have the consistent changing tendency and similar shape with the curves in FIGURE 5.

For example, when the probability distribution of the rest $K - 1$ $(K = 5)$ classes is balanced, $\frac{\partial Gini}{\partial l_j}$ achieves the lower bound, $2p_j \cdot G\left(p_j\right)$. If $p_j > \frac{1}{K}$ (indicates that the model has classified the sample as the jth class), we can see that $\frac{\partial Gini}{\partial l_j} < 0$ from FIGURE 5, which leads the model to update in the direction of keeping increasing l_j by *SGD* algorithm. That means the confidence of the model will enhance in classification. Furthermore, when $p_j \to \frac{1}{K} or 1$, it indicates that the model has little or great confidence to classify the sample as the jth class, the effect of Gini-regularization decrease. Similarly, when $p_j < \frac{1}{K}$, the model further denies that the sample belongs to the jth class. Therefore, the Gini-regularization term can adjust the learning rate reasonably and accelerate the convergence of the model.

However, the current predictions of the model is not necessarily correct, especially at the beginning of training. Thus, to avoid the model updating in the wrong direction, we add a coefficient λ to control the effect of Gini-regularization.

Consider the final segmentation plane in the feature space, as shown in FIGURE 3, the Gini-regularization term can also make the distance from the sample points to the segmentation plane larger. Just like the idea of the maximum-margin hyperplane in SVM, it can improve the generalization ability of the model.

IV. EXPERIMENTS

A. GENERAL DATASETS

To validate the effectiveness of our proposed algorithm, we modified the loss function based on the original SGAN. Moreover, the same datasets are utilized, includes the MNIST(consists of 700000 28 × 28 digits images) [28], CIFAR-10(consists of 600000 32 × 32 natural images in 10 classes) [29] and SVHN dataset(consists of over 600000 32×32 digit images in the natural scene) [30] in experiments. Meanwhile, we randomly select 100, 4000, and 1000 (average distributed in each category) samples from the training set as labeled data respectively, and the remaining as unlabeled data. The experimental hardware environment is a k80 graphics card with 11G memory. For the complex cifar-10 and svhn datasets, the time cost is so expensive that each epoch consume 10 minutes. The experiment results of test error curves are shown in FIGURE 6, $\lambda = 0$ represents the result of the original. It can see that the Gini-regularization term can help the model converge faster and better when the value λ is proper. Unfortunately, λ is data-dependent; therefore, it must go through many repeated experiments to find the optimal value.

B. LITHOLOGIC DATASETS

The post-stack seismic data is from a certain area of Huabei Oilfield, with a total of 1956 traces and 501 sampling points, and the sampling rate is 2ms. Since the amplitude data of different frequencies can record the stratum information, the low-frequency amplitude data reflect the general stratigraphic structure distribution better, and the high-frequency amplitude data indicates the stratum details better. Therefore, one full-frequency data and eight different frequency data

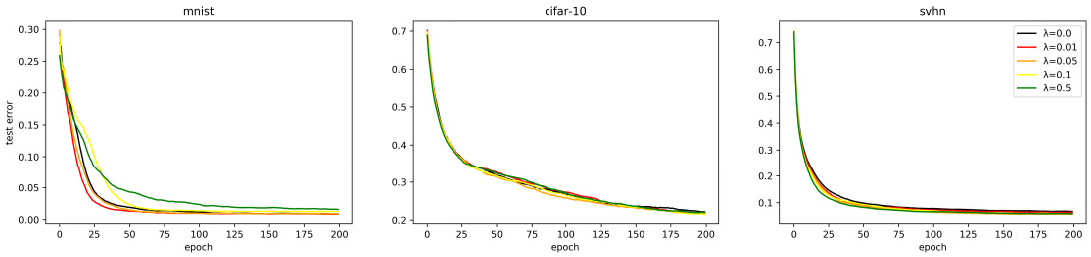


FIGURE 6. Test errors of different λ on MNIST, CIFAR-10 and SVHN. The black line represents the original SGAN model. The others are SGAN_G models with different λ.

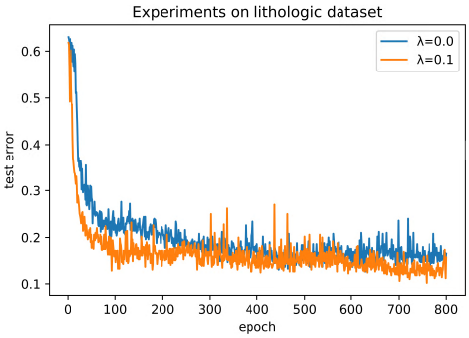


FIGURE 7. Test errors of SGAN and SGAN_G on lithologic dataset.

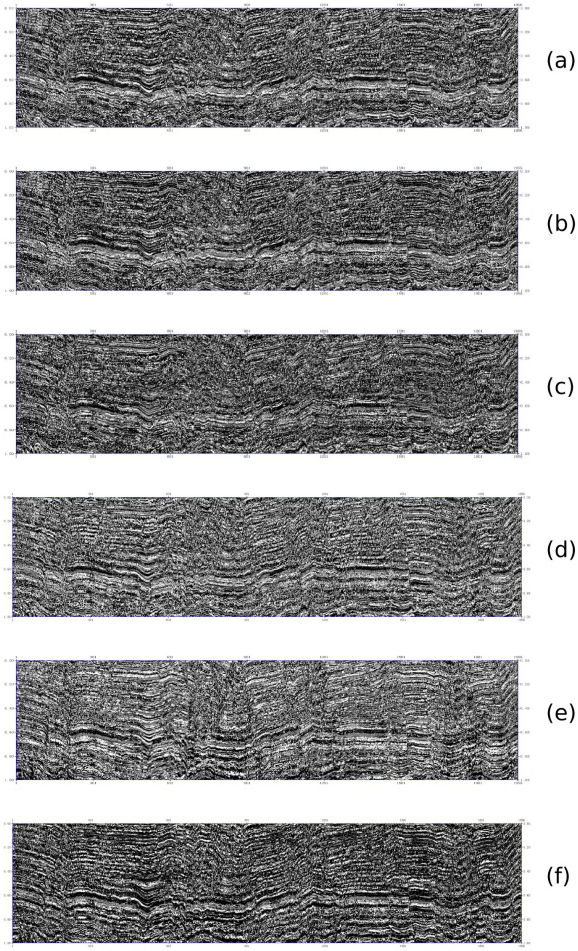


FIGURE 8. Lithologic profile predicted by sixe models. Black, white, and gray represent shale, sandstone, and other lithology. (a) Adaboost. (b) RandomForest. (C) DNN. (d) DBN. (e) SGAN. (f) SGAN_G.

of the seismic data are selected as input. To train the model better, the input data needs to be normalized.

The area contains a total of lithology data from seven well logs. The vertical resolution of the seismic data is lower than the well logging data. Therefore, the original lithology curve must be roughened. According to the porosity, the lithology of the formation is divided into shale, sand stone and other lithology, which are represented by 0, 1, and 2 respectively, as lithology identification output data. The dataset includes 2,000 labeled data samples which are divided into test and training datasets according to 1:3. Moreover, we introduce 900,000 unlabeled data for semi-supervised models.

To validate the effectiveness of SGAN_G, we compare with SGAN, DBN, BP neural network, Random Forest and AdaBoost models. In experiments, SGAN_G and DNN are implemented by using tensorflow in python, DBN is implemented by using python numpy library, and Random Forest and AdaBoost are calling python sklearn library. The optimal parameters of the model adopt minimizing test error according to grid search (as shown in Table 1). Note that the coefficient of $L2$-regularization term is set to 0.1 in the BP model, as well as the discriminator of SGAN_G and SGAN, to enhance the generalization ability.

The training error and the test error are two critical factors to measure classification performance. The training error illustrates the credibility of the model, and the test error represents the availability of the model (the degree of generalization). If the training error is small, and the test error is large, lead to the over-fitting, otherwise, lead to the under-fitting. Therefore, the test error and the training error is small simultaneous, can ensure the model with strong generalization ability.

From the analysis of the model errors (as shown in Table 1), we can see that the test error of the BP network is smaller than that of Random Forest and AdaBoost, but it is still larger. Compare to the BP network, the test error of the DBN network significantly reduced. It is worth noting that the test

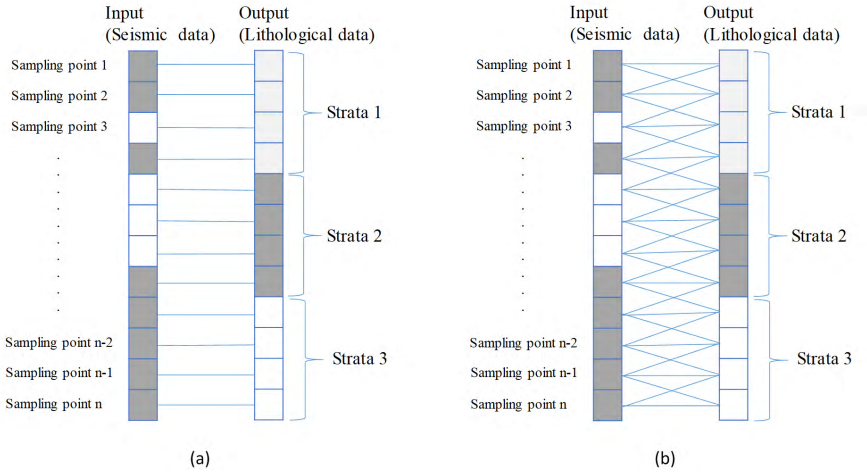


FIGURE 9. The input and output of models: (a) single-point sampling: the corresponding lithology is predicted according to each sample point's feature and (b) multiple-points sampling: the lithology of the center point is predicted according to multiple-points' feature.

TABLE 1. Comparison of experimental results.

Model	Network parameters	Training error rate	Test error rate
SGAN_G	Generator: 20*50 Discriminator: 25*100*50*20 λ : 0.1	0.005	0.139
SGAN	Generator: 20*50 Discriminator: 25*100*50*20	0.005	0.161
DBN	25*100*50	0.037	0.223
BP	25*100*50*20	0.093	0.300
Model	Number of tree	Training error rate	Test error rate
Random Forest	100	0	0.370
AdaBoost	70	0	0.326

error of SGAN model is reduced by 6.2% compared to the DBN network, which is a significant improvement of model performance. The SGAN_G model proposed in this paper, further reduced the test error by 2.2%. FIGURE 7 shows the test error curves of SGAN($\lambda = 0.0$) and SGAN_G($\lambda = 0.1$) model. It validates that the Gini-regularization term can improve the convergence speed and generalization ability of the model.

FIGURE 8(a-f) show the lithology profile identified by the sixe models, with black, white, and gray corresponding to shale, sandstone, and other lithology. The above experiments demonstrate that the lithology profile identified by SGAN_G is more detailed and clearer, and the horizon is more continuous. By contrast, the profiles identified by other models are relatively rough and some are very blurred. Either from the test error or identification of the lithology profile, SGAN_G is the best of the sixe models.

1) MULTI-SAMPLING POINTS

Due to the formation has continuity in the longitudinal direction, any sampling point associates with its adjacent points.

TABLE 2. The experiment of multi-sampling points.

SGAN_G	Training error rate	Test error rate
SGAN_G-1	0.005	0.139
SGAN_G-3	0.008	0.107
SGAN_G-5	0.004	0.058
SGAN_G-7	0.003	0.055
SGAN_G-8	0.002	0.064

Therefore, compared with using single sampling point of seismic trace(shown in FIGURE 9(a)), utilizing multiple adjacent points as input(shown in FIGURE 9(b)) take full advantage of "layer" information to achieve better recognition results [16]. Let k denotes the number of neighborhood sampling points. As shown in Table 2, with multi-sampling points method, the test error of SGAN_G further decrease to about 5% ($k = 7$), can obtain the optimal classification performance.

To compare the impact of multi-sampling points on lithology recognition in detailed, FIGURE 10 shows the recognition results of the lithology section with the time window from 0.2 to 1.0 seconds and traces from 300 to 1500. By comparison, the experiment results illustrate that the number of neighborhood sampling points is a critical factor for lithology recognition. When $k = 1$, the identified profile is delicate, though the test error is significant. With the k increases, the main horizon of the identified lithology profile becomes clearer. But when $k \geq 7$, because of the complex structure of underground media and the lithology changes frequently, the lithologic boundary becomes unclear. The excessive sampling points will affect the recognition of the target lithology. Therefore, in practical applications, the lithology profile of different resolutions can be identified by adjusting the size of k, and analyses the profile from various aspects.

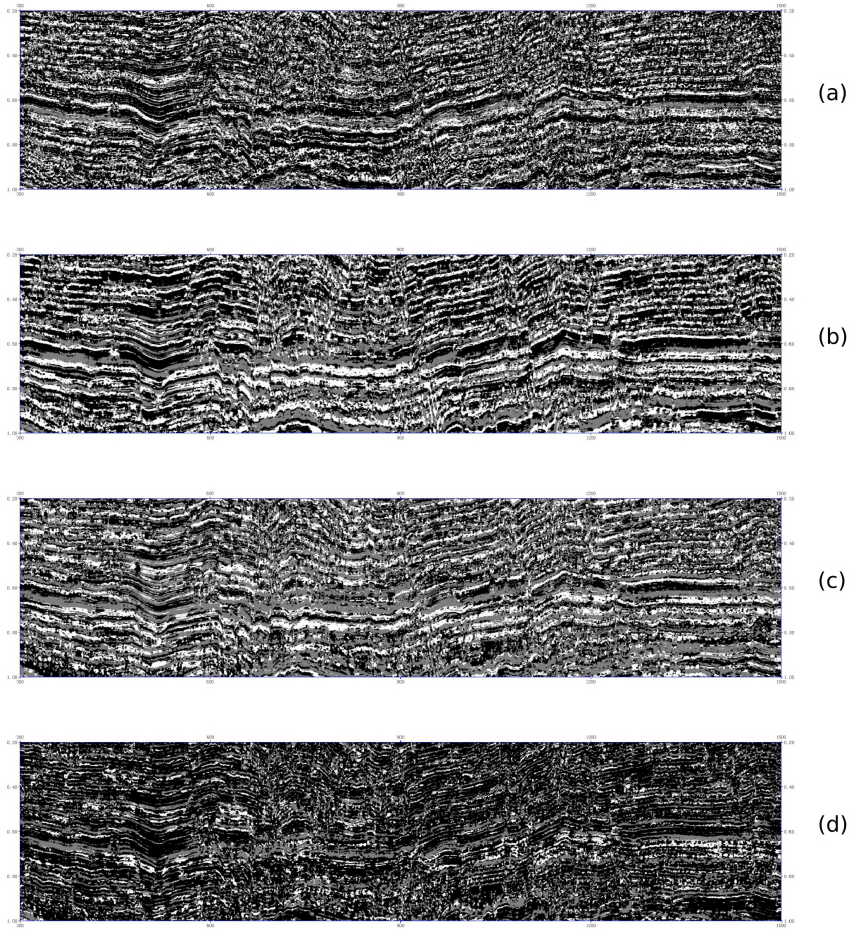


FIGURE 10. Lithologic profile predicted by multiple. Black, white, and gray represent shale, sandstone, and other lithology. (a) k = 1. (b) k = 3. (c) k = 5. (d) k = 7.

V. CONCLUSION AND FUTURE WORK

In petroleum exploration, machine learning approaches have been widely utilized as an effective means of modeling. With the advent of the era of big data, the effect of deep learning has surpassed the traditional machine learning method, and become a hot spot of research and application. In these, there are numerous studies on GAN. The semi-supervised learning based on GAN effectively employs a large amount of unlabeled data to establish a nonlinear mapping between input and output according to a small amount of labeled data. Since the information provided by the unlabeled data for model training is limited, therefore, we utilize the Gini-regularization term to the unsupervised part of the original discriminator loss function to help the model converge faster and better. In the field of lithology recognition, because logging data (labeled data) is minimal, supervised learning methods are challenging to achieve good results, and generalization ability. SGAN_G can employ a mass of seismic data as unlabeled data for training. Thus, the model can extract features better which are very consistent with the data characteristics in the field of lithology recognition. Through experimental comparison, the performance of lithology recognition enhance significantly. Moreover, using multi-sampling points as inputs, containing the information of ''layer'', further improves the recognition accuracy.

Notwithstanding the performance of the classifier could not degrade in the balanced general datasets(MNIST, CIFAR-10, and SVHN) or the unbalanced dataset(lithologic dataset) in our models. But the security of introducing unlabeled samples is essential and needs to be further analyzed and studied in future work. Besides, the selection of optimal value only relies on multiple repeated experiments. Therefore, how to obtain the proper λ more efficiently and automatically will be the focus of further research.

ACKNOWLEDGMENT

The authors would like to thank colleagues at the Data and Knowledge Engineering Center, School of Information Technology and Electrical Engineering, University of Queensland. They would also like to thank Prof. Xiaofang Zhou for his special suggestions and many interesting discussions.

REFERENCES

[1] J. Busby and C. Fgs, ''An introduction to geophysical exploration,'' *Geophys. J. Int.*, vol. 152, no. 2, p. 506, 2003.

[2] H. Zhang, L. Zou, and X. Shen, "The application if BP neural network in well lithology identification," *Geol. Prospecting*, vol. 38, no. 6, pp. 63–65, 2002.
[3] W. Zhihong, H. Lu, and Q. Lei, "Random forest classification method in the application of reservoir lithology recognition," (in Chinese), *J. Liaoning Tech. Univ. (Natural Science)*, vol. 34, no. 9, pp. 1083–1088, 2015.
[4] I. Priezzhev and E. Stanislav, "Application of machine learning algorithms using seismic data and well logs to predict reservoir properties," presented at the 80th EAGE Conf. Exhib., 2018.
[5] Y. Kim, R. Hardisty, E. Torres, and K. J. Marfurt, "Seismic-facies classification using random forest algorithm," in *Proc. SEG Tech. Program Expanded Abstr.*, Society of Exploration Geophysicists, 2018, pp. 2161–2165.
[6] G. Röth and A. Tarantola, "Neural networks and inversion of seismic data," *J. Geophys. Res., Solid Earth*, vol. 99, no. B4, pp. 6753–6768, 1994.
[7] L.-L. Cui, L.-Y. Qi, S.-F. Liu, and B.-L. Hu, "Litholoy prediction with multiple seismic attributes and neural network," *Acta Scientiarum Naturalium Univ. Nankaiensis*, vol. 40, no. 2, pp. 99–104, 2007.
[8] S. Yu, K. Zhu, and F. Diao, "A dynamic all parameters adaptive BP neural networks model and its application on oil reservoir prediction," *Appl. Math. Comput.*, vol. 195, no. 1, pp. 66–75, 2008.
[9] U. Iturrarán-Viveros and J. O. Parra, "Artificial neural networks applied to estimate permeability, porosity and intrinsic attenuation using seismic attributes and well-log data," *J. Appl. Geophys.*, vol. 107, pp. 45–54, Aug. 2014.
[10] A. A. Silva, I. A. L. Neto, R. M. Misságia, M. A. Ceia, A. G. Carrasquilla, and N. L. Archilha, "Artificial neural networks to support petrographic classification of carbonate-siliciclastic rocks using well logs and textural information," *J. Appl. Geophys.*, vol. 117, pp. 118–125, Jun. 2015.
[11] T. Zhao, "Seismic facies classification using different deep convolutional neural networks," in *Proc. SEG Tech. Program Expanded Abstr.*, Society of Exploration Geophysicists, 2018, 2046–2050.
[12] Y. Shi, X. Wu, and S. Fomel, "Automatic salt-body classification using a deep convolutional neural network," in *Proc. SEG Tech. Program Expanded Abstr.*, Society of Exploration Geophysicists, 2018, pp. 1971–1975.
[13] J. S. Dramsch and M. Lüthje, "Deep-learning seismic facies on state-of-the-art CNN architectures," in *Proc. SEG Tech. Program Expanded Abstr.*, Society of Exploration Geophysicists, 2018, pp. 2036–2040.
[14] A. U. Waldeland and A. Solberg, "Salt classification using deep learning," presented at the 79th EAGE Conf. Exhib., 2017.
[15] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, no. 5786, pp. 504–507, 2006.
[16] L. Liu, R. Lu, J. Li, and W. Yang, "Seismic lithofacies computation method based on deep learning," presented at the Int. Geophys. Conf., Qingdao, China, Apr. 2017.
[17] G.-H. Li, Y. Zheng, Y. Li, W.-J. Wu, Y.-F. Hong, and X.-M. Zhou, "Lithology recognition of multi-sampling points based on deep belief network," *Progr. Geophys.*, vol. 33, no. 4, pp. 1660–1665, 2018.
[18] R. O. V. Santos, M. M. B. R. Vellasco, F. A. V. Artola, and S. A. B. da Fontoura, "Neural net ensembles for lithology recognition," *Multiple Classifier Systems*. Berlin, Germany: Springer, 2003.
[19] G. E. Hinton, "Training products of experts by minimizing contrastive divergence," *Neural Comput.*, vol. 14, no. 8, pp. 1771–1800, 2002.
[20] T. Tieleman, "Training restricted Boltzmann machines using approximations to the likelihood gradient," presented at the ICML, 2008.
[21] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 1929–1958, 2014.
[22] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," 2015, *arXiv:1502.03167*. [Online]. Available: https://arxiv.org/abs/1502.03167
[23] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*. [Online]. Available: https://arxiv.org/abs/1412.6980
[24] I. J. Goodfellow *et al.*, "Generative adversarial networks," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 3. 2014, pp. 2672–2680.
[25] A. Creswell, T. White, V. Dumoulin, K. Arulkumaran, B. Sengupta, and A. A. Bharath, "Generative adversarial networks: An overview," *IEEE Signal Process.*, vol. 35, no. 1, pp. 53–65, Jan. 2017.
[26] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, X. Chen, and X. Chen, "Improved Techniques for Training GANs," in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 2234–2242.
[27] A. Odena, "Semi-supervised learning with generative adversarial networks," 2016, *arXiv:1606.01583*. [Online]. Available: https://arxiv.org/abs/1606.01583
[28] Y. LeCun. (1998). *The MNIST Database of Handwritten Digits*. [Online]. Available: http://yann.lecun.com/exdb/mnist/
[29] A. Krizhevsky and G. E. Hinton, "Learning multiple layers of features from tiny images," Univ. Toronto, Toronto, ON, Canada, Tech. Rep., 2009. [Online]. Available: http://www.cs.toronto.edu/~kriz/cifar.html
[30] Y. Netzer, T. Wang, A. Coates, A. Bissacco, B. Wu, and A. Y. Ng, "Reading digits in natural images with unsupervised feature learning," in *Proc. NIPS Workshop Deep Learn. Unsupervised Feature Learn.*, 2011, pp. 1–9. [Online]. Available: http://udl.stanford.edu/housenumbers/