Title

Generative Adversarial Net

Introduction

Before GAN was proposed, the most eye-catching results of deep learning in the field of computer vision were basically discriminative models, such as image classification, target recognition, and so on. But in fact, the other half of the story is Deep Generative Models. The influence of the generated model has been small, mainly because of the tricky probability calculation problem when using maximum likelihood estimation for deep neural networks (such as CNN), and GAN's proposal subtly bypasses this problem.

Content

This research proposes a new model of deep learning and is one of the most promising methods for unsupervised learning. Generative Adversarial Networks (GANs) is a generative model promoted by Goodfellow Ian of the University of Montreal in 2014, which has attracted wide attention and research in the industry. The discriminative model has achieved great success in deep learning and even machine learning. Its essence is to map the feature vector of the sample into the corresponding label; the generative model requires a large amount of prior knowledge for the modeling of the reality, and the choice of prior distribution directly influences the outcomes of the model. Therefore, people have paid more attention to the discriminant model method.

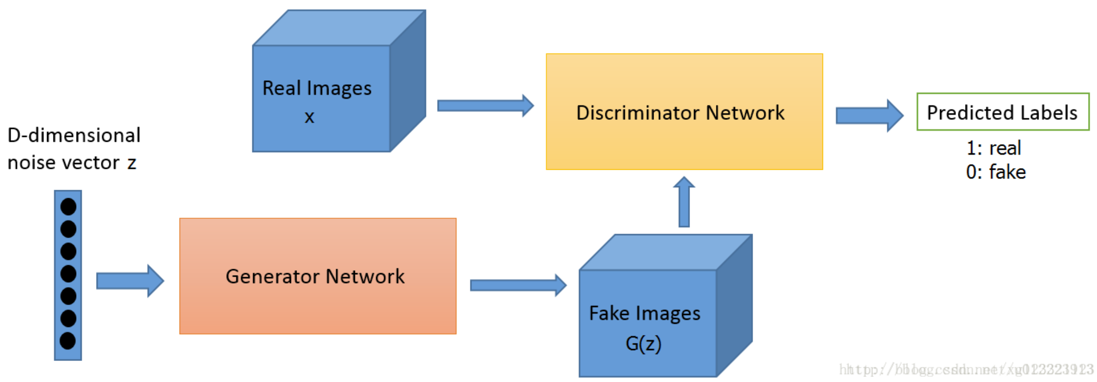

Figure 1.

The ultimate goal of GAN is to learn a high-quality generator G, which achieves a high-quality generator G by introducing a discriminator D. The purpose of G in the training process is to generate as realistic a picture as possible so that the discriminator can't judge whether the picture is a real picture or a fake photo. The purpose of D in the training process is to distinguish between true and false pictures as much as possible. G is hoping that D's error rate is maximized, while D is hoping that his mistake rate is minimized. The two are adversarial and progress together in the competition. In theory, this relationship can reach a balance point, the so-called Nash equilibrium, that is to say, the probability that the image D generated by G discriminates it as real data is 0.5, that is, the discriminator can no longer distinguish the image generated by the generator. True or false, then the purpose of the generator is to achieve the real thing. The optimization objective function of GAN in Paper is as follows:

$$\min_G \max_D V(D,G) = E_{x\sim p_{data}(x)}[log(D(x))] + E_{z\sim p_z(z)}[log(1-D(G(z)))]$$

This is a maximum and minimum optimization problem because it involves the largest and smallest problems, it is difficult to optimize together. Therefore, the author adopts an alternate training strategy, first training the discriminator network D, optimizing the parameters of the discriminator, and then training the generator network G. The training steps are as follows:

Algorithm 1 Minibatch stochastic gradient descent training of generative adversarial nets. The number of steps to apply to the discriminator, k, is a hyperparameter. We used $k = 1$, the least expensive option, in our experiments.

for number of training iterations **do**

  for k steps **do**

- Sample minibatch of m noise samples $\{\boldsymbol{z}^{(1)}, \ldots, \boldsymbol{z}^{(m)}\}$ from noise prior $p_g(\boldsymbol{z})$.
- Sample minibatch of m examples $\{\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(m)}\}$ from data generating distribution $p_{\text{data}}(\boldsymbol{x})$.
- Update the discriminator by ascending its stochastic gradient:

$$\nabla_{\theta_d} \frac{1}{m} \sum_{i=1}^{m} \left[\log D\left(\boldsymbol{x}^{(i)}\right) + \log\left(1 - D\left(G\left(\boldsymbol{z}^{(i)}\right)\right)\right)\right].$$

  end for

- Sample minibatch of m noise samples $\{\boldsymbol{z}^{(1)}, \ldots, \boldsymbol{z}^{(m)}\}$ from noise prior $p_g(\boldsymbol{z})$.
- Update the generator by descending its stochastic gradient:

$$\nabla_{\theta_g} \frac{1}{m} \sum_{i=1}^{m} \log\left(1 - D\left(G\left(\boldsymbol{z}^{(i)}\right)\right)\right).$$

end for

The gradient-based updates can use any standard gradient-based learning rule. We used momentum in our experiments.

The author explained in the paper that the step=K of the training D network is a hyperparameter. In the experiment, the author takes K=1. In theory, when the update speed of the G network is slow enough, the D network training can be saved more fully. In turn, the performance of the G network is better. However, there is also a problem in the above training generator network. The author also pointed out in the original text that in the early stage of training, the performance of G is poor, and the image generated by G is quite different from the original data. It is easy for D network to judge that it is false. The picture, that is, D(G($z^{(i)}$)) is small, log(1- D(G($z^{(i)}$))) is close to 0, which causes the gradient to disappear, which will cause the G network training to be insufficient, so the author is optimizing the G network. At the same time, the loss function of the G network is adjusted accordingly to become minimize-log(D(G($z^{(i)}$))). The author also proves through the theory that the existence of the global optimal solution, at this time D network LOSS=1.38, the accuracy of D for G generated data should be 50%.

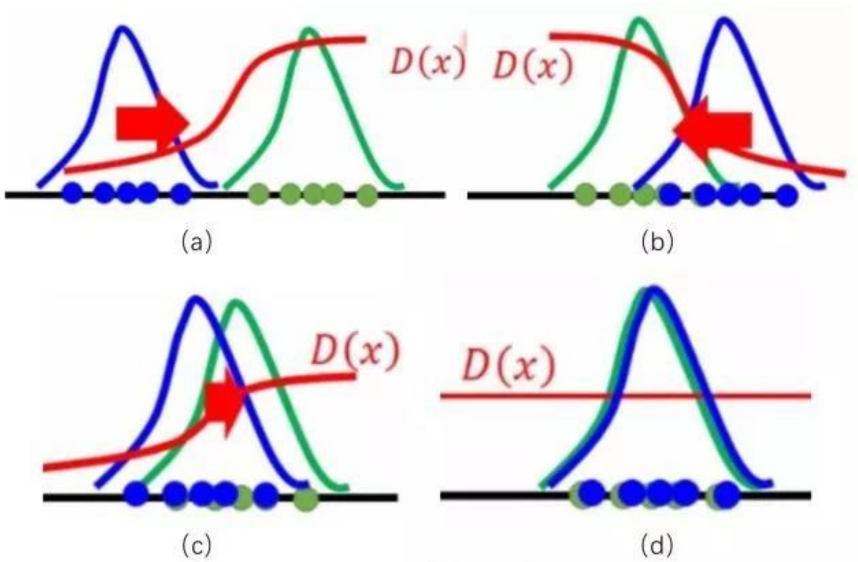

Figure 2.

Blue represents the distribution of the generated model, green represents the actual distribution, and D is the discriminator. According to the probability distribution, GAN pushes the generated distribution through D to the actual distribution and then optimizes D until it is as shown in Fig. 2. (d) indicates that the Nash equilibrium point is reached and the generator distribution coincides with the actual distribution.

Innovation

GAN is important to the generation model development. As a proficient method, GAN solves the troubles of generating natural interpretation data effectively. Especially for the generation of high-dimensional data, the generation does not limit the neural network structure used. This dimension greatly extends the range of generated data samples. The neural network structure used can synthesize different loss functions and improve the freedom of design.

GAN's training process innovatively uses the confrontation of two neural networks as a training criterion as well as can be trained using backpropagation. The process of training does not require a less efficient Markov chain method and does not require various approximate inferences. The lower bound of the complex variation greatly improves the training difficulty and the generated model training efficiency. The GAN generation process does not require cumbersome sampling sequences, and can directly sample and infer new samples, enhancing the efficiency of the new sample generation. This approach discards direct copying or averaging of real data, enhancing the diversity of samples generated. In the practice of generating a sample of Gan, the generated sample is easy for human understanding. For example, it is possible to creatively generate a very sharp image for a generation. Human meaningful data provide a possible solution.

In addition to the contribution to the generative model, GAN is also instructive for semi-supervised learning. GAN does not require data tags in the learning process. Although the purpose of GAN is not semi-supervised learning, the process of GAN training can be used to achieve unsupervised semi-supervised learning. The data preprocessing process of the model. Specifically, the GAN is trained with unlabeled data, based on the learned GAN's understanding of the data, and a small number of tagged data training discriminators are used for traditional classification and regression tasks.

Technology quality

GAN was originally designed to avoid Markov chains because of the high computational cost of the latter. Compared to the Boltzmann machine, the GAN limit is much smaller (only a few probability distributions apply to Markov chain sampling). Traditionally generated models generally require Markov chain inference as well as sampling, and GANs avoids this computationally complex process, directly sampling and inferring, thereby improving the application efficiency of GANs. The design framework of GANs is very flexible. Different types of loss functions can be designed for different tasks. Especially when the probability density of data is uncalculated, some traditional models that depend upon the natural interpretation of data will not work, but the adversarial training of GANs. The mechanism can still be used in this case. In addition to the computational advantages, the GAN has some statistical advantages, that is, the generation network G is not directly updated by the data samples, but only the gradient signals output by the D network are used to update the corresponding weight parameters of the G network. This means that the parameters that generate the network G are not directly derived from the input data.

Simultaneously, GAN also has some disadvantages. The first difficulty is that it is difficult to train and is prone to collapse. The GAN model is defined as a minimally large problem. Without a loss function, it is difficult to distinguish whether progress is being made during the training process.

The second difficulty is that the model is too free and uncontrollable without prior modeling. Compared with other generative models, GAN's competitive approach no longer requires a hypothetical data distribution. It uses a distribution to directly sample the sampling so that it can completely approximate the real data theoretically, which is the greatest advantage of GAN. However, the disadvantage of this method that does not require pre-modeling is that it is too free. For larger images, more pixel cases are less controllable based on the simple GAN approach.

Application and X-Factor

Since the birth of GANs, it has been widely used in many fields such as images, videos, and texts, and is still expanding. Some representative application examples are described below.

Super Resolution

Super-resolution represents the generation of corresponding high-resolution images from a given low-resolution image and has important application value in the satellite image, monitoring, medical imaging, and other fields. Traditional methods generally use interpolation, but they create blur. In September 2016, Twitter published a research project to develop a new loss function that uses a network of 16 residual blocks to parameterize the generated model. The discriminant model uses the VGG network, enabling the GANs to recover sharply down sampled images into high-resolution, clear images with rich detail.

Data synthesis (Apple)

In December 2016, Apple's first AI paper built a generative adversarial network for synthesizing a batch of tagged, real image data sets. Training machine learning with composite images and video can reduce time and labor costs. Composite images are already tagged and annotated and can be customized.

In my opinion, GAN is a generative model that employs model learning for the estimation of its potential distribution while generating new samples of the same distribution, rather than directly estimating the distribution of data samples. This is a model that is worth discussing and researching. Besides, the theoretical inference about the convergence of GAN as well as the existence of equilibrium points is also an essential research topic in the future.

Presentation

The generative adversarial framework proposed in this paper is a kind of "game theory" idea to realize the evaluation process of the generative model. The GAN consists of a set of adversarial neural networks (called generators and discriminators) that attempt to generate generative samples that can be mistaken for real samples by the discriminator. Compared with other generative models, the significant difference of GAN is that the implementation method of GAN is to let D and G play the game, and the two models are enhanced at the same time by competing in the training process. Because of the existence of the discriminative model D, G can learn to approximate the real data without a large amount of prior knowledge and prior distribution, and finally let the data generated by the model achieve the effect of false.

The author gives an example to help us understand the model. The generator G can be likened to a counterfeit coin maker team, trying to produce counterfeit money that cannot detect the authenticity; the discriminator D can be likened to the police, trying to distinguish the real currency. And counterfeit money. In the process of competition, both parties constantly improve their methods, which ultimately leads to the inability to distinguish between counterfeit currency and genuine products. Explain that we got a very good generator G.

Compared with the traditional generation model, GAN does not need to rely on the Markov chain, and the network can be directly trained using the backpropagation algorithm, unlike the Boltzmann machine, which requires a large number of approximate estimates. However, some shortcomings are difficult to training and the model is too free and uncontrollable.

In conclusion, GAN's solution meets the application as well as the research needs of many fields and also injects new development momentum into these areas.

Reference

1. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A. & Bengio, Y. 2014, 'Generative adversarial nets', Advances in neural information processing systems, pp. 2672-80.
2. Radford, A., Metz, L. & Chintala, S. 2015, 'Unsupervised representation learning with deep convolutional generative adversarial networks', arXiv preprint arXiv:1511.06434.
3. Creswell, A., White, T., Dumoulin, V., Arulkumaran, K., Sengupta, B. & Bharath, A.A. 2018, 'Generative adversarial networks: An overview', IEEE Signal Processing Magazine, vol. 35, no. 1, pp. 53-65.