In these two weeks, I mainly learned the basic idea of deep learning, the training process of deep learning, and the commonly used models of deep learning.

1. The basic idea of Deep Learning

　　Suppose we have a system S with n layers, whose input is I and output is O. It is visually represented as: I=>S1=>S2=> Sn=>0, if output O is equal to input I, that is, there is no information loss after the input I undergoes this system change, it can be proved that the mutual information between a and c does not exceed the mutual information between a and b by assuming that the processing of a information yields b, and then the processing of b yields c. This indicates that information processing does not add information, and most processing will lose information. Of course, if you lose useless information, it would be great. It remains unchanged, which means that input I passes through each layer of Si without any information loss. That is, at any layer of Si, it is another representation of the original information. Now back to our topic Deep Learning, we need to automatically learn features. Suppose we have a pile of input I, and suppose we design a system S. By adjusting the parameters in the system so that its output is still input I, we can automatically obtain a series of hierarchical features to input I, namely S1,..., Sn.

For deep learning, the idea is to stack multiple layers, which means that the output of one layer is used as the input of the next layer. In this way, it is possible to achieve hierarchical representation of input information on a 7/13 basis. In addition, the previous assumption is that the output is strictly equal to the input. This restriction is too strict, and we can slightly relax this restriction. For example, we can only make the difference between the input and output as small as possible. This relaxation will lead to a different type of Deep Learning method. The above is the basic idea of deep learning.

二、Deep learning training process

If you train all layers simultaneously, the time complexity will be too high. If you train one layer at a time, the deviation will be transmitted layer by layer. This can lead to the most severe under fitting. Hinton proposed an effective method for building a multi-layer neural network on unsupervised data. To put it simply, there are two steps: one is to train a layer of network each time, and the other is to tune it. The high-level representation r generated from the original representation x up and the high-level representation r generated from the high-level representation r down should be as consistent as possible. The method is:

(1) First, we construct single layer neurons layer by layer, so that each time we train a single layer network.

(2) After all layers have been trained, Hinton uses the walk sleep algorithm for tuning.

Change the weights between layers other than the topmost layer to bidirectional, so that the topmost layer is still a single layer neural network, while the other layers become graph models. The upward weight is used for "cognition", and the downward weight is used for "generation". Then use the Wake-Sleep algorithm to adjust all the weights. To achieve agreement between cognition and generation is to ensure that the generated top-level representation can restore the underlying nodes as accurately as possible. For example, if a node on the top layer represents a face, then the image of all faces should activate this node, and the resulting downward generated image should be able to represent a rough face image. Wake-Sleep algorithm is divided into two parts: waking and sleeping.

1) Wake stage: cognitive process, which generates an abstract representation of each layer through external features and upward weights, and uses gradient descent to modify the downward weights between layers. "If the reality is different from what I imagined, changing my weight makes what I imagined look like this.".

2) Sleep stage: the generation process, which generates the status of the bottom layer through top-level representation and downward weighting, while modifying the upward weighting between layers. "If the scene in a dream is not a corresponding concept in my brain, changing my cognitive weight makes this scene seem to me to be this concept.". The detailed process of deep learning training is as follows:

(1) Unsupervised learning from the bottom up

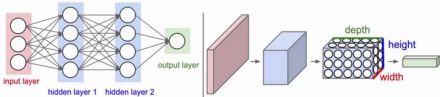
Start at the bottom and train one layer at a time to the top. Using uncalibrated data (with or without calibrated data) to hierarchically train the parameters of each layer can be seen as an unsupervised training process, which is also the biggest difference from traditional neural networks and can be seen as a feature learning process. Specifically, first train the first layer with uncalibrated data, and first learn the parameters of the first layer during training. This layer can be seen as obtaining a hidden layer of a three-layer neural network that minimizes the difference between output and input. Due to model capacity constraints and sparsity constraints, the resulting model can learn the structure of the data itself, thereby obtaining features that are more expressive than the input; After learning to obtain the n-l layer, the output of the n-l layer is used as the input of the n-th layer, and the n-th layer is trained to obtain the parameters of each layer.

(2) Top down supervised learning

It is to train through labeled data, transmit errors from top to bottom, and fine-tune the network. Further tuning the parameters of the entire multi-layer model based on the parameters of each layer obtained in the first step is a supervised training process. The first step is similar to the random initialization initial value process of neural networks. Since the first step is not random initialization, but is obtained by learning the structure of the input data, this initial value is closer to the global optimal value, which can achieve better results. Therefore, the good effect of deep learning is largely attributed to the first step of the feature learning process.

三、Common models for deep learning

Convolutional neural network model

[](https://baike.baidu.com/pic/%E6%B7%B1%E5%BA%A6%E5%AD%A6%E4%B9%A0/3729729/0/bd3eb13533fa828b269a8a0ff31f4134960a5aa4?fr=lemma&fromModule=lemma_content-image&ct=single)

Before the advent of unsupervised pre training, it was often very difficult to train deep neural networks, and one special case was convolutional neural networks. Convolutional neural networks are inspired by the structure of the visual system. The first convolutional neural network computing model was proposed in Fukushima's neurocognitive machine. Based on local connections between neurons and hierarchical tissue image conversion, neurons with the same parameters are applied to different positions in the previous neural network to obtain a translation invariant neural network structure. Later, Le Cun et al. based on this idea, designed and trained convolutional neural networks using error gradients, achieving superior performance in some pattern recognition tasks. Up to now, the pattern recognition system based on convolutional neural networks is one of the best implementation systems, especially in handwritten character recognition tasks, showing extraordinary performance.