

Chapter09 Tips for Deep Learning

1. Bad Results on Training Data?

a) New activation function

i) Vanishing Gradient Problem

In the layers near input, gradients are small, learn very slow, and the parameters are almost random, whereas in the layers near output, gradients are large, learn very fast, and the parameters are almost converged

Intuitive reason: Sigmoid function: large input, small output

ii) Alternative method:

ReLU:

$$x < 0: y = 0; \quad x > 0: y = x$$

1. Fast to compute
2. Biological reason
3. Handle the problem of vanishing gradient

Leaky ReLU:

$$x < 0: y = 0.01x; \quad x > 0: y = x$$

Parametric ReLU:

$$x < 0: y = \alpha x; \quad x > 0: y = x$$

α is learned by gradient descent

Maxout:

1. ReLU is a special case of Maxout
2. Learnable activation function
 - a) Activation function in Maxout network can be any piecewise linear convex function
 - b) How many pieces depending on how many elements in a group

b) Adaptive learning rate

i) Adagrad

$$1) \quad w^{t+1} = w^t - \frac{\eta}{\sqrt{\sum_{i=0}^t (g^i)^2}} g^t$$

2) Use first derivative to estimate second derivative

ii) RMSProp

1) Error surface can be very complex when training DNN

$$2) \quad w^1 = w^0 - \frac{\eta}{\sigma^0} g^0 \quad \sigma^0 = g^0$$

$$w^2 = w^1 - \frac{\eta}{\sigma^1} g^1 \quad \sigma^1 = \sqrt{\alpha(\sigma^0)^2 + (1 - \alpha)(g^1)^2}$$

$$w^{t+1} = w^t - \frac{\eta}{\sigma^t} g^t \quad \sigma^t = \sqrt{\alpha(\sigma^{t-1})^2 + (1 - \alpha)(g^t)^2}$$

- iii) Momentum
 - 1) Movement not just based on gradient, but previous movement
 - 2) Movement = momentum + negative of $\partial L / \partial w$
- iv) Adam
RMSProp + Momentum

2. Bad Results on Testing Data?

- a) Early Stopping
- b) Regularization
 - i) L1 Regularization
 - 1) $L'(\theta) = L(\theta) + \lambda \|\theta\|_1$ $\|\theta\|_1 = |w_1| + |w_2| + \dots$
 - 2) $\frac{\partial L'(\theta)}{\partial w} = \frac{\partial L(\theta)}{\partial w} + \lambda \text{sgn}(w)$
 - 3) $w^{t+1} = w^t - \eta \frac{\partial L}{\partial w} - \eta \lambda \text{sgn}(w^t)$
 - ii) L2 Regularization
 - 1) $L'(\theta) = L(\theta) + \frac{1}{2} \lambda \|\theta\|_2$ $\|\theta\|_2 = (w_1)^2 + (w_2)^2 + \dots$
 - 2) $\frac{\partial L'(\theta)}{\partial w} = \frac{\partial L(\theta)}{\partial w} + \lambda w$
 - 3) $w^{t+1} = (1 - \eta \lambda) w^t - \eta \frac{\partial L}{\partial w}$
 - iii) Weight Decay
- c) Dropout
 - i) Training:
 - 1) Each time before updating the parameters, each neuron has p% to dropout, the structure of the network is changed
 - 2) Using the new network for training
 - ii) Testing:
 - 1) No Dropout
 - 2) If the dropout rate at training is p%, all the weights times 1-p%
 - iii) Dropout is a kind of ensemble
 - 1) Train a bunch of networks with different structures
 - 2) Using one mini-batch to train one network
 - 3) Some parameters are shared in the network