Assignment 3, Liping TANG, 217019012

2018年12月5日 20:02

1. $h_t = \tanh(W_{xh} x_t + W_{hh} h_{t-1} + b_h)$ $\quad V_t = W_{xh} x_t + W_{hh} h_{t-1} + b_h$

$z_t = \mathrm{softmax}(W_{hz} h_t + b_z)$ $\quad U_t = W_{hz} h_t + b_z$

$L(x,y) = \sum_t L_t = \sum_t -\log z_{t,y_t}$

Suppose $x_t \in \mathbb{R}^d$, $z_t \in \mathbb{R}^c$, then $h_t \in \mathbb{R}^d$, $W_{xh}, W_{hh} \in \mathbb{R}^{d\times d}$, $W_{hz} \in \mathbb{R}^{c\times d}$.

$$\frac{\partial L}{\partial L_t} = 1, \quad \frac{\partial L}{\partial z_t} = \frac{\partial L}{\partial L_t}\cdot\frac{\partial L_t}{\partial z_t} = \frac{\partial L_t}{\partial z_t} = \begin{bmatrix} 0 \\ \vdots \\ -\frac{1}{z_{t,y_t}} \\ 0 \\ \vdots \\ 0 \end{bmatrix} \leftarrow \text{index} = y_t$$

i.e. $\frac{\partial L}{\partial z_{t,i}} = 0$, for $i \neq y_t$, $\quad \frac{\partial L}{\partial z_{t,i}} = -\frac{1}{z_{t,i}}$ for $i = y_t$

①
$$\begin{aligned}
\frac{\partial L}{\partial W_{hz,ij}} &= \sum_t \frac{\partial L}{\partial z_t}\cdot\frac{\partial z_t}{\partial W_{hz,ij}} \\
&= \sum_{t,k} \frac{\partial L}{\partial z_{t,k}} \frac{\partial z_{t,k}}{\partial W_{hz,ij}} \\
&= \sum_{t,k,l} \frac{\partial L}{\partial z_{t,k}}\cdot\frac{\partial z_{t,k}}{\partial U_{t,l}}\cdot\frac{\partial U_{t,l}}{\partial W_{hz,ij}} \\
&= \sum_{t,l} \frac{\partial L}{\partial z_{t,y_t}}\cdot\frac{\partial z_{t,y_t}}{\partial U_{t,l}}\cdot\frac{\partial U_{t,l}}{\partial W_{hz,ij}} \\
&= \sum_t \frac{\partial L}{\partial z_{t,y_t}}\cdot\frac{\partial z_{t,y_t}}{\partial U_{t,i}}\cdot h_{t,j} \\
&= \sum_t -\frac{h_{t,j}}{z_{t,y_t}}\cdot\frac{\partial z_{t,y_t}}{\partial U_{t,i}}
\end{aligned}$$

where
$$\frac{\partial z_{t,y_t}}{\partial U_{t,i}} = \begin{cases} -z_{t,y_t}\cdot z_{t,i} & \text{for } i \neq y_t \\ z_{t,y_t}(1-z_{t,y_t}). & \text{for } i = y_t \end{cases}$$

②
$$\frac{\partial L}{\partial W_{hh}} = \sum_t \frac{\partial L}{\partial h_t}\frac{\partial h_t}{\partial W_{hh}}$$

$$\frac{\partial L}{\partial h_t} = \frac{\partial L}{\partial h_{t+1}}\cdot\frac{\partial h_{t+1}}{\partial h_t} + \frac{\partial L}{\partial z_t}\cdot\frac{\partial z_t}{\partial h_t}$$

$h_t = \tanh(V_t)$.

$$\frac{\partial h_{t,k}}{\partial V_{t,i}} = \begin{cases} 1 - h_{t,k}^2 & \text{for } k = i \\ 0 & \text{for } k \neq i \end{cases}$$

$$\frac{\partial h_{t+1,i}}{\partial h_{t,k}} = \sum_l \frac{\partial h_{t+1,i}}{\partial V_{t,l}}\cdot\frac{\partial V_{t,l}}{\partial h_{t,k}} = \frac{\partial h_{t+1,i}}{\partial V_{t,i}}\cdot\frac{\partial V_{t,i}}{\partial h_{t,k}}$$
$$= \frac{\partial h_{t+1,i}}{\partial V_{t,i}} W_{hh,ik}$$

$$\frac{\partial V_{t,l}}{\partial W_{hh,ij}} = \frac{\partial((W_{xh}x_t)_l + (W_{hh}h_{t-1})_l + b_{h,l})}{\partial W_{hh,ij}} = \begin{cases} h_{t-1,j} & \text{for } i = l \\ 0 & \text{for } i \neq l \end{cases}$$

Assignment 3

2018年12月5日 20:39

$$\frac{\partial L}{\partial h_{t,k}} = \frac{\partial L}{\partial h_{t+1}} \cdot \frac{\partial h_{t+1}}{\partial h_{t,k}} + \frac{\partial L}{\partial z_t} \cdot \frac{\partial z_t}{\partial h_{t,k}}$$

$$= \sum_i \frac{\partial L}{\partial h_{t+1,i}} \frac{\partial h_{t+1,i}}{\partial h_{t,k}} + \sum_i \frac{\partial L}{\partial z_{t,i}} \cdot \frac{\partial z_{t,i}}{\partial h_{t,k}}$$

$$= \frac{\partial L}{\partial h_{t+1,k}} \cdot \frac{\partial h_{t+1,k}}{\partial h_{t,k}} + \sum_{i,l} \frac{\partial L}{\partial z_{t,i}} \cdot \frac{\partial z_{t,i}}{\partial u_{t,l}} \frac{\partial u_{t,l}}{\partial h_{t,k}}$$

$$= \frac{\partial L}{\partial h_{t+1,k}} \cdot (1 - h_{t+1,k}^2) W_{hh,kk} + \sum_l \frac{\partial L}{\partial z_{t,y_t}} \cdot \frac{\partial z_{t,y_t}}{\partial u_{t,l}} \cdot \frac{\partial u_{t,l}}{\partial h_{t,k}}$$

$$= \frac{\partial L}{\partial h_{t+1,k}} (1 - h_{t+1,k}^2) W_{hh,kk} + \sum_l -\frac{1}{z_{t,y_t}} \cdot \frac{\partial z_{t,y_t}}{\partial u_{t,l}} \cdot W_{hz,lk} \qquad ①$$

where $\frac{\partial z_{t,y_t}}{\partial u_{t,i}} = \begin{cases} -z_{t,y_t} \cdot z_{t,i} & \text{for } i \neq y_t \\ z_{t,y_t}(1 - z_{t,y_t}) & \text{for } i = y_t \end{cases}$

$$\frac{\partial L}{\partial W_{hh,ij}} = \sum_t \frac{\partial L}{\partial h_t} \cdot \frac{\partial h_t}{\partial W_{hh,ij}} = \sum_{t,k} \frac{\partial L}{\partial h_{t,k}} \cdot \frac{\partial h_{t,k}}{\partial W_{hh,ij}} = \sum_{t,k,l} \frac{\partial L}{\partial h_{t,k}} \frac{\partial h_{t,k}}{\partial V_{t,l}} \cdot \frac{\partial V_{t,l}}{\partial W_{hh,ij}}$$

$$= \sum_{t,k} \frac{\partial L}{\partial h_{t,k}} \cdot \frac{\partial h_{t,k}}{\partial V_{t,i}} \cdot h_{t-1,j}$$

$$= \sum_t \frac{\partial L}{\partial h_{t,i}} \cdot (1 - h_{t,i})^2 \cdot h_{t-1,j} \qquad ②$$

Put ① in ②. We'll obtain a recursive term which contains $\frac{\partial L}{\partial h_{t,k}}$ and $\frac{\partial L}{\partial h_T} = 0$, where T is the last step.

2 1) $h_t = F_\theta(h_{t-1}, x_t)$

$$= F_\theta(F_\theta(h_{t-2}, x_{t-1}), x_t)$$

$$= F_\theta(F_\theta(F_\theta(h_{t-3}, x_{t-2}), x_{t-1}), x_t)$$

$$= F_\theta(F_\theta(F_\theta(h_{t-4}, x_{t-3}), x_{t-2}), x_{t-1}), x_t)$$

$$\vdots$$

Finally, $h_2 = F_\theta(F_\theta(h_0, x_1), x_2)$, $h_1 = F_\theta(h_0, x_1)$

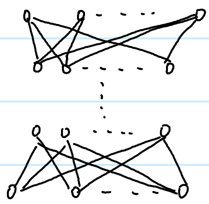
2) ① RNN shares parameters: the same weights are used for different instances at different time steps. This allows to apply the network to input sequences of different length and hence improve the generalization property of the model

② RNN maps an arbitrary-length sequence $(x_1, \dots, x_t)$ to a fixed-length vector h_t, which avoids the exponential growth of model complexity.

Assignment 3

2018年12月5日 15:12

3\. 1)



c nodes

n nodes } L layers

⋮

n nodes

d nodes

the number of parameters:

$$dn + n^2 \cdot (L-1) + cn$$

2) Suppose we have m nodes in layer i and n nodes in layer i+1.

From Layer i to layer i+1, we have $y = f(Wx)$, where $x \in \mathbb{R}^m$, $y \in \mathbb{R}^n$.

$$\frac{\partial J}{\partial W_{ij}} = \sum_k \frac{\partial J}{\partial y_k} \cdot \frac{\partial y_k}{\partial W_{ij}}$$

$$= \sum_k \frac{\partial J}{\partial y_k} \cdot f'((Wx)_k) \cdot \frac{\partial(\sum_l W_{kl} x_l)}{\partial W_{ij}}$$

$$= \frac{\partial J}{\partial y_i} f'((Wx)_i) \cdot x_j$$

$W \in \mathbb{R}^{n\times m}$. for each $\frac{\partial J}{\partial W_{ij}}$, we have to do 2 multiplications.

Finally, we need $2mn$ multiplications

$$\frac{\partial J}{\partial x_i} = \sum_k \frac{\partial J}{\partial y_k} \cdot \frac{\partial y_k}{\partial x_i} = \sum_k \frac{\partial J}{\partial y_k} \cdot f'((Wx)_k) \frac{\partial(\sum_l W_{kl} x_l)}{\partial x_i}$$

$$= \frac{\partial J}{\partial y_i} f'((Wx)_i) W_{ki}$$

$x \in \mathbb{R}^m$, for each $\frac{\partial J}{\partial x_i}$. we need 2 multiplication

Thus we need $2m$ multiplication for $\frac{\partial J}{\partial x}$.

The whole Back Propogation steps:

Output $\rightarrow$ Hidden L : for W : $2nc$, for neurons (x): $2n$

Hidden i+1 $\rightarrow$ Hidden i : for W : $2n^2$, for neurons (x): $2n$

Hidden$_1$ $\rightarrow$ Input : for W: $2nd$.

total number of multiplications: $2nc + 2n + (2n^2 + 2n)(L-1) + 2nd$

3) $$\frac{\partial u_N}{\partial u_i} = \sum_{\text{paths } u_{k1} \cdots u_{kn}} \prod_{j=2}^{N} \frac{\partial u_{kj}}{\partial u_{kj-1}}$$

For this term, we have $c\,n^{(N-i-1)}$ paths and $n-2$ multiplications for each path.

Output $\rightarrow$ Hidden$_L$: for W. $2cn$, for neurons: cn

Hidden$_{i+1}$ $\rightarrow$ Hidden$_i$: for W: $2n^2$, for neurons: $(L-i+1)\,c\,n^{L-i}$

Hidden$_1$ $\rightarrow$ Input : for W: $2dn$

total number of multiplications:

$$2cn + cn + 2n^2 + 2dn + \sum_{i=1}^{L-1} (L-i+1)\,cn^{L-i}$$

Assignment 3

2018年12月5日 17:48

4. If there's one linear hidden layer, we have

$$\tilde{x} = U_2 U_1 x.$$

where $U_2 \in \mathbb{R}^{d\times k}$, $U_1 \in \mathbb{R}^{k\times d}$ are the weights of the autoencoder, an $x \in \mathbb{R}^d$.

If we have N samples $\{x_n\}_{n=1}^N$, using the mean squared error criterion,

we have the cost $J = \frac{1}{N}\sum_{n=1}^{N} \|\tilde{x}_n - x_n\|^2$

We introduce a complete orthonormal set of d-dimensional basis vectors $\{u_i\}_{i=1}^d$

Then the data can be represented by a linear combination of these basis vectors:

$$x_n = \sum_{i=1}^{d} \alpha_{ni} u_i.$$

where $\alpha_{ni} = x_n^T u_i$.

W.l.o.g. we suppose that the first k basis vectors correspond to the representation from the k hidden units in autoencoder. Then

$$\tilde{x}_n = \sum_{i=1}^{k} z_{ni} u_i + \sum_{i=k+1}^{d} b_i u_i$$

To minimize the mean squared error, $J = \frac{1}{N}\sum_{n=1}^{N} \left\| \sum_{i=1}^{k} z_{ni} u_i + \sum_{i=k+1}^{d} b_i u_i - x_n \right\|^2$

Let $\frac{\partial J}{\partial z_{nj}} = \frac{2}{N}(x_n - \tilde{x}_n)^T u_j = \frac{2}{N}(x_n^T u_j - z_{nj}) = 0$

$$\Rightarrow z_{nj} = x_n^T u_j$$

$$\frac{\partial J}{\partial b_j} = \frac{2}{N}\sum_{n=1}^{N}(x_n - \tilde{x}_n)^T u_j = \frac{2}{N}\sum_{n=1}^{N}(x_n^T u_j - b_j) = 0$$

$\Rightarrow b_n = \bar{x} u_j$, where $\bar{x}$ is the sample mean.

Then $x_n - \tilde{x}_n = \sum_{i=1}^{d}(x_n^T u_i) u_i - \left(\sum_{i=1}^{k}(x_n^T u_i) u_i - \sum_{i=k+1}^{d}(\bar{x}^T u_i) u_i \right)$

$$= \sum_{i=k+1}^{d} \left[(x_n - \bar{x})^T u_i \right] u_i$$

$$J = \frac{1}{N}\sum_{n=1}^{N}\sum_{i=k+1}^{d}(x_n^T u_i - \bar{x} u_i)^2 = \sum_{i=k+1}^{d} u_i^T S u_i$$

where $S = \frac{1}{N}\sum_{n=1}^{N}(x_n^T - \bar{x})(x_n^T - \bar{x})^T$ is the data covariance matrix.

To minimize J, $\{u_i\}_{i=k+1}^{d}$ should be the eigenvectors that correspond to the least $d-k$ eigenvalues of S. i.e. $\{u_i\}_{i=1}^{k}$ are the eigenvectors that correspond to the largest k eigenvalues of S, which means the encoder is the k-dimensional PCA projection of x_n.