Assignment 2 for CS224d

Lifu Huang

March 2016

0 Warmup: Boolean Logic

(a)

$$XOR(x, y) = NOT(x\ AND\ y)\ AND\ (x\ OR\ y) \tag{1}$$

(b)

$$h_{AND}(x, y) = \theta(x + y - 1.5) \tag{2}$$
$$h_{OR}(x, y) = \theta(x + y - 0.5) \tag{3}$$
$$h_{NOT}(x) = \theta(-x + 0.5) \tag{4}$$
$$\tag{5}$$

(c)

See part0-XOR.ipynb

1 Deep Networks for Named Entity Recognition

(a)

$$
\begin{aligned}
\delta^{(2)} &= \nabla_{z^{(2)}} J && (6)\\
&= \hat{y} - y && (7)\\
\frac{\partial J}{\partial U} &= \frac{\partial J}{\partial z^{(2)}} \frac{\partial z^{(2)}}{\partial U} && (8)\\
&= (\delta^{(2)})^T \left[\frac{\partial z_i^{(2)}}{\partial U} \right] && (9)\\
&= (\delta^{(2)})^T \left[\frac{\partial}{\partial U} (\epsilon_i^T (Uh + b_2)) \right] && (10)\\
&= (\delta^{(2)})^T \left[h(\epsilon_i)^T \right] && (11)\\
&= h(\delta^{(2)})^T && (12)\\
\nabla_U J &= \delta^{(2)} h^T && (13)\\
\text{similarly,} & && (14)\\
\nabla_{b2} J &= \delta^{(2)} && (15)\\
\delta^{(1)} &= (1 - h^2) \circ (U^T \delta^{(2)}) && (16)\\
\nabla_W &= \delta^{(1)} x^{(t)} && (17)\\
\nabla_{b_1} &= \delta^{(1)} && (18)\\
\nabla_L &= W^T \delta^{(1)} && (19)
\end{aligned}
$$

(b)

$$
\begin{aligned}
\nabla_U J_{full} &= \nabla_U J + \nabla_U J_{reg} && (20)\\
&= \delta^{(2)} h^T + \lambda U && (21)\\
\nabla_W J_{full} &= \nabla_W J + \nabla_W J_{reg} && (22)\\
&= \delta^{(1)} (x^{(t)})^T + \lambda W && (23)
\end{aligned}
$$

(b*)

Proof

$$
\begin{aligned}
P(\theta|\Sigma,\mu) &= \frac{1}{(2\pi)^{n/2}|\Sigma|^{\frac{1}{2}}} exp(-\frac{1}{2}(\theta-\mu)^T\Sigma^{-1}(\theta-\mu)) \\
&= \frac{1}{(2\pi)^{n/2}} exp(-\frac{1}{2}\theta^T\theta) \\
\log P(\theta|\Sigma,\mu) &= \log \frac{1}{(2\pi)^{n/2}} exp(-\frac{1}{2}\theta^T\theta) \\
&= -\log(2\pi)^{n/2} - \frac{1}{2}||\theta||^2 \\
\arg\max_\theta \log P(\theta|\Sigma,\mu) &= \arg\max_\theta \frac{1}{2}||\theta||^2
\end{aligned}
$$

(c), (d), (e), (f)

Please see specific files for solution to these questions.

1.1 Deep Networks: Probing Neuron Responses

Please see specific files for solution to this question.

2 Recurrent Neural Networks: Language Modeling

(a)

Proof

$$
\begin{aligned}
PP^{(t)}(\hat{y}^{(t)}, y^{(t)}) &= \frac{1}{\hat{P}(x_{t+1}^{pred} = x_{t+1} | x_t, ..., x_1)} \\
&= \frac{1}{\sum_{j=1}^{|V|} y_j^{(t)} \hat{y}_j^{(t)}} \\
&= \frac{1}{\hat{y}_k^{(t)}} \\
&= \exp(-\log \hat{y}_k^{(t)}) \\
&= \exp(J^{(t)}) \\
\arg\min_\theta \left(\prod_{t=1}^{T} PP^{(t)}(\hat{y}^{(t)}, y^{(t)}) \right)^{\frac{1}{T}} &= \arg\min_\theta \left(\log \left(\prod_{t=1}^{T} PP^{(t)}(\hat{y}^{(t)}, y^{(t)}) \right)^{\frac{1}{T}} \right) \\
&= \arg\min_\theta \left(\frac{1}{T} \sum_{t=1}^{T} \log PP^{(t)}(\hat{y}^{(t)}, y^{(t)}) \right) \\
&= \arg\min_\theta \left(\frac{1}{T} \sum_{t=1}^{T} J^{(t)} \right)
\end{aligned}
$$

TED

Baseline

$$
\begin{aligned}
PP^{(t)}(\hat{y}^{(t)}, y^{(t)}) &= \frac{1}{\hat{y}_k^{(t)}} \\
&= \frac{1}{\frac{1}{|V|}} \\
&= |V| \\
CE_{2000} &= \log 2000 = 7.6 \\
CE_{10000} &= \log 10000 = 9.2
\end{aligned}
$$

(b)

$$\begin{aligned}
\delta^{(2)(t)} &= \hat{y}^{(t)} - y^{(t)} \\
\nabla_U J^{(t)} &= \delta^{(2)(t)}(h^{(t)})^T \\
\delta^{(1)(t)} &= h^{(t)} \circ (1 - h^{(t)}) \circ (W^T \delta^{(2)(t)}) \\
\nabla_{L_{x^{(t)}}} J^{(t)} &= \delta^{(1)(t)} \\
\nabla_H J^{(t)}\Big|_{(t)} &= \delta^{(1)(t)}(h^{(t-1)})^T \\
\nabla_{h^{(t-1)}} J^{(t)} &= H^T(\delta^{(1)(t)})
\end{aligned}$$

(c)

$$\begin{aligned}
\delta^{(1)(t-1)} &= h^{(t-1)} \circ (1 - h^{(t-1)}) \circ (H^T \delta^{(1)(t)}) \\
\nabla_{L_{x^{(t-1)}}} J^{(t)} &= \delta^{(1)(t-1)} \\
\nabla_H J^{(t)}\Big|_{(t-1)} &= \delta^{(1)(t-1)}(h^{(t-2)})^T
\end{aligned}$$

(d)

$$\begin{aligned}
\text{Forward for one step} &= O(|V|D_h) \\
\text{Backward for one step} &= O(|V|D_h) \\
\text{Backward for } \tau \text{ steps} &= O(|V|D_h)
\end{aligned}$$

if $|V| > D_h$, then the slow step is the computation of the output layer.

(e)(f)(g)

Please see specific files for solution to these questions.