CS224d Assignment 1

Jeremy Irvin

October 24, 2016

1.
(a)

$$\begin{aligned}
\text{softmax}(\mathbf{x}+c)_i &= \frac{e^{\mathbf{x}_i+c}}{\sum_j e^{\mathbf{x}_j+c}} \\
&= \frac{e^c e^{\mathbf{x}_i}}{e^c \sum_j e^{\mathbf{x}_j}} \\
&= \frac{e^{\mathbf{x}_i}}{\sum_j e^{\mathbf{x}_j}} \\
&= \text{softmax}(\mathbf{x})_i
\end{aligned}$$

2.
(a)

$$\begin{aligned}
\nabla_x \sigma(x) &= \nabla_x \frac{1}{1+e^{-x}} \\
&= \frac{1}{(1+e^{-x})^2} \cdot e^{-x} \\
&= \frac{e^{-x}}{1+e^{-x}} \cdot \frac{1}{1+e^{-x}} \\
&= (1-\sigma(x))\sigma(x)
\end{aligned}$$

(b) Assuming the kth element of $\boldsymbol{y}$ is 1 and all other elements are 0, then

$$\begin{aligned}
CE(\boldsymbol{y}, \hat{\boldsymbol{y}}) &= \left(-\sum_i y_i \log(\hat{y}_i)\right) \\
&= (-y_k \log(\hat{y}_k)) \\
&= -\log(\text{softmax}(\theta)_k) \\
&= -\log\left(\frac{e^{\theta_k}}{\sum_j e^{\theta_j}}\right) \\
&= -\theta_k + \log\left(\sum_j e^{\theta_j}\right)
\end{aligned}$$

Therefore

$$\begin{aligned}
\frac{\partial}{\partial \theta_k} CE(\boldsymbol{y}, \hat{\boldsymbol{y}}) &= -1 + \frac{e^{\theta_k}}{\sum_j e^{\theta_j}} \\
&= \frac{e^{\theta_k}}{\sum_j e^{\theta_j}} - 1 \\
&= \hat{y}_k - 1
\end{aligned}$$

and for all $i \neq k$,

$$\begin{aligned}
\frac{\partial}{\partial \theta_i} CE(\boldsymbol{y}, \hat{\boldsymbol{y}}) &= \frac{e^{\theta_i}}{\sum_j e^{\theta_j}} \\
&= \hat{y}_i
\end{aligned}$$

Thus $\frac{\partial}{\partial\theta}CE(\boldsymbol{y},\hat{\boldsymbol{y}}) = \hat{\boldsymbol{y}} - \boldsymbol{y}$.

(c) Say $\boldsymbol{W}_1 \in \mathbb{R}^{D\times H},\ \boldsymbol{W}_2 \in \mathbb{R}^{H\times O}$. Then $\frac{\partial J}{\partial \boldsymbol{x}} \in \mathbb{R}^D$. Hence

$$\begin{aligned}
\frac{\partial J}{\partial \boldsymbol{x}} &= \frac{\partial}{\partial\theta}CE(\boldsymbol{y},\hat{\boldsymbol{y}})\frac{\partial\theta}{\partial\boldsymbol{x}},\ \theta = \boldsymbol{h}\,\boldsymbol{W}_2 + \boldsymbol{b}_2 \\
&= (\hat{\boldsymbol{y}} - \boldsymbol{y})\,\frac{\partial\theta}{\partial\boldsymbol{x}} \\
&= (\hat{\boldsymbol{y}} - \boldsymbol{y})\,\frac{\partial\theta}{\partial\boldsymbol{h}}\frac{\partial\boldsymbol{h}}{\partial\boldsymbol{x}},\ \boldsymbol{h} = \sigma(\boldsymbol{x}\,\boldsymbol{W}_1 + \boldsymbol{b}_1) \\
&= (\hat{\boldsymbol{y}} - \boldsymbol{y})\,\frac{\partial\theta}{\partial\boldsymbol{h}}\frac{\partial\boldsymbol{h}}{\partial z}\frac{\partial z}{\partial\boldsymbol{x}},\ z = \boldsymbol{x}\,\boldsymbol{W}_1 + \boldsymbol{b}_1 \\
&= (\hat{\boldsymbol{y}} - \boldsymbol{y})\,\boldsymbol{W}_2^T\sigma'(z)\,\boldsymbol{W}_1^T
\end{aligned}$$

(d) There are $D_x \cdot H + H + H \cdot D_y + D_y$ parameters.

3.
(a) Let $\boldsymbol{y}$ and $\hat{\boldsymbol{y}}$ be row vectors. Then assuming the oth element of $\boldsymbol{y}$ is 1 and the other elements are 0,

$$\begin{aligned}
\frac{\partial}{\partial\boldsymbol{v}_c}J_{softmax-CE}(\boldsymbol{o},\boldsymbol{v}_c,\boldsymbol{U}) &= \frac{\partial}{\partial\boldsymbol{v}_c}CE(\boldsymbol{y},\hat{\boldsymbol{y}}) \\
&= \frac{\partial}{\partial\boldsymbol{v}_c}\left(-\log\left(\frac{\exp(\boldsymbol{u}_o^T\boldsymbol{v}_c)}{\sum_{w=1}^W\exp(\boldsymbol{u}_w^T\boldsymbol{v}_c)}\right)\right) \\
&= \frac{\sum_{x=1}^W\exp(\boldsymbol{u}_x^T\boldsymbol{v}_c)\boldsymbol{u}_x^T}{\sum_{w=1}^W\exp(\boldsymbol{u}_w^T\boldsymbol{v}_c)} - \boldsymbol{u}_o^T \\
&= \sum_{x=1}^W p(\boldsymbol{x}|\boldsymbol{c})\boldsymbol{u}_x^T - \boldsymbol{u}_o^T \\
&= \sum_{x=1}^W \hat{\boldsymbol{y}}_x\boldsymbol{u}_x^T - \boldsymbol{u}_o^T \\
&= (\hat{\boldsymbol{y}} - \boldsymbol{y})U
\end{aligned}$$

(b)

$$\begin{aligned}
\frac{\partial}{\partial\boldsymbol{u}_w}J_{softmax-CE}(\boldsymbol{o},\boldsymbol{v}_c,\boldsymbol{U}) &= \frac{\partial}{\partial\boldsymbol{u}_w}\left(-\log\left(\frac{\exp(\boldsymbol{u}_o^T\boldsymbol{v}_c)}{\sum_{x=1}^W\exp(\boldsymbol{u}_x^T\boldsymbol{v}_c)}\right)\right) \\
&= \begin{cases} \frac{\exp(\boldsymbol{u}_o^T\boldsymbol{v}_c)\boldsymbol{v}_c^T}{\sum_{x=1}^W\exp(\boldsymbol{u}_x^T\boldsymbol{v}_c)} - \boldsymbol{v}_c^T, & \text{if } w = o \\ \frac{\exp(\boldsymbol{u}_w^T\boldsymbol{v}_c)\boldsymbol{v}_c^T}{\sum_{x=1}^W\exp(\boldsymbol{u}_x^T\boldsymbol{v}_c)}, & \text{otherwise} \end{cases} \\
&= \begin{cases} \hat{\boldsymbol{y}}_w\boldsymbol{v}_c^T - \boldsymbol{v}_c^T, & \text{if } w = o \\ \hat{\boldsymbol{y}}_w\boldsymbol{v}_c^T, & \text{otherwise} \end{cases}
\end{aligned}$$

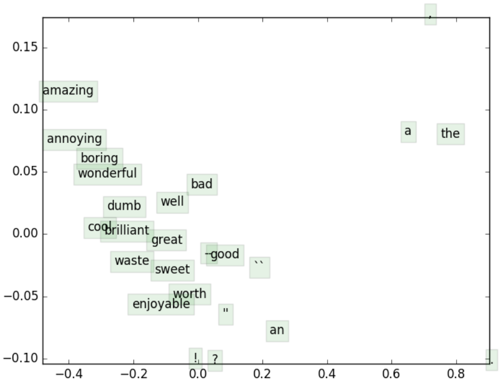
(c) Now viewing the vectors as column vectors,

$$\begin{aligned}
\frac{\partial}{\partial\boldsymbol{v}_c}J_{neg-sample}(\boldsymbol{o},\boldsymbol{v}_c,\boldsymbol{U}) &= \frac{\partial}{\partial\boldsymbol{v}_c}\left(-\log(\sigma(\boldsymbol{u}_o^T\boldsymbol{v}_c)) - \sum_{k=1}^K\log(\sigma(-\boldsymbol{u}_k^T\boldsymbol{v}_c))\right) \\
&= (\sigma(\boldsymbol{u}_o^T\boldsymbol{v}_c) - 1)\boldsymbol{u}_o - \sum_{k=1}^K(\sigma(-\boldsymbol{u}_k^T\boldsymbol{v}_c) - 1)\boldsymbol{u}_k
\end{aligned}$$

$$\frac{\partial}{\partial\boldsymbol{u}_w}J_{neg-sample}(\boldsymbol{o},\boldsymbol{v}_c,\boldsymbol{U}) = \begin{cases} (\sigma(\boldsymbol{u}_o^T\boldsymbol{v}_c) - 1)\boldsymbol{v}_c, & \text{if } w = o \\ -(\sigma(\boldsymbol{u}_k^T\boldsymbol{v}_c) - 1)\boldsymbol{v}_c, & \text{if } w = k \text{ for } k = 1, ..., K \end{cases} \tag{1}$$

The gradients of the softmax-CE loss take $O(Wd)$ to compute the normalizing term (where d is the size of the word vectors) whereas the gradients of the negative sampling loss function only take $O(Kd)$ to compute, where $K << W$.

(d)



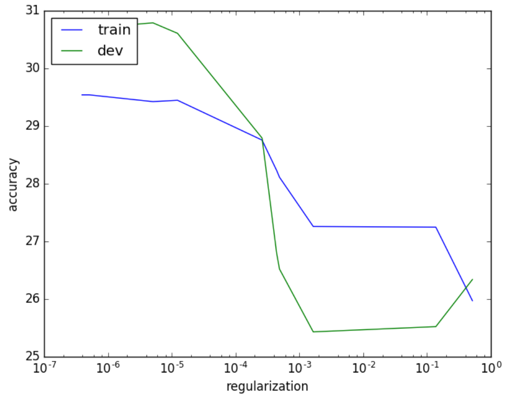
(g)

All of the adjectives are near each other, the articles are near eachother, and the punctuation are near eachother. The positive adjectives tend to be closer to eachother (and same for the ngative adjectives).

4.

(b) To prevent overfitting and thus improve generalization to new data.

(c) The regularization parameter chosen was 5.076209E-06. This yielded train accuracy 29.424157%, dev accuracy 30.790191%, and test accuracy 28.506787%. The search methodology was simply to sample uniformly on a log scale (10^{-8} to 10^{1}) and choose the regularization parameter which yielded the best dev set accuracy.



(d)

The dev accuracy drops off once a certain regularization parameter becomes too large, and also decreases when it becomes too small (ie overfitting the training set).