CS 224n Assignment #2: word2vec

Notation:

- $\mathbf{v}$: 'center' column vector
- $\mathbf{u}$: 'outside' column vector
- o : The index of the desired context (outside) word.
- w : The w -th word in the vocabulary
- c : The index of the center word.
- $\tilde{y}$: The predicted distribution, row vector.

In **word2vec**, the conditional probability distribution is given by taking vector dot-products and applying the softmax function:

$$\tilde{y} = P(O = o \mid C = c) = \frac{\exp\left(\boldsymbol{u}_o^\top \boldsymbol{v}_c\right)}{\sum_{w \in \text{Vocab}} \exp\left(\boldsymbol{u}_w^\top \boldsymbol{v}_c\right)}$$

$$\boldsymbol{J}_{\text{naive-softmax}}\ (\boldsymbol{v}_c, o, \boldsymbol{U}) = -\log P(O = o \mid C = c)$$

(a) Show that the naive-softmax loss is the same as the cross-entropy loss between y and $\tilde{y}$; i.e. show that:

$$-\sum_{w \in Vocab} y_w \log\left(\hat{y}_w\right) = -\log\left(\hat{y}_o\right)$$

Solution: The true empirical distribution y is a one-hot vector with a 1 for the true outside word o, and 0 everywhere else. The sum on the LHS breaks down as follows:

$$-\sum_{w \in Vocab} y_w \log\left(\hat{y}_w\right) = -\left(y_1 \log\left(\hat{y}_1\right) + \ldots + y_o \log\left(\hat{y}_o\right) + \ldots + y_{|V|} \log\left(\hat{y}_{|V|}\right)\right) = -\log\left(\hat{y}_o\right)$$

(b) Compute the partial derivative of $\boldsymbol{J}_{\text{naive-softmax}}\ (\boldsymbol{v}_c, o, \boldsymbol{U})$ with respect to $\mathbf{v_c}$.

Solution:

$$
\begin{aligned}
\frac{\partial J_{naive-softmax}}{\partial \mathbf{v_c}} &= \frac{\partial}{\partial \mathbf{v_c}}\left[-\log\left(\hat{y}_o\right)\right] \\
&= -\mathbf{u_o} + \sum_{x \in V \text{ ocab}} \frac{e^{\mathbf{u_x}^T \mathbf{v_c}}}{\sum_{w \in \text{Vocab}} e^{\mathbf{u_w}^T \mathbf{v_c}}} \mathbf{u_x} \\
&= -\mathbf{u_o} + \sum_{x \in V \text{ ocab}} \tilde{y_x} \mathbf{u_x} \\
&= \mathbf{U}(\tilde{y} - y)^T
\end{aligned}
$$

This says that the slope of the loss funciton w.r.t the center word is equal to the difference between the observed representation of the outside word and the expected context word according to our model.

(c) Compute the partial derivative of $\boldsymbol{J}_{\text{naive-softmax}}\ (\boldsymbol{v}_c, o, \boldsymbol{U})$ with respect to each of the 'outside' word vectors, $\mathbf{u_w}$'s. There will be two cases: when $w = o$, the true 'outside' word vector, and $w \neq o$, for all other words.

Solution:

Case 1 - the outside word vector is the true context word vector

$$
\begin{aligned}
\frac{\partial J_{naive-softmax}}{\partial u_{w=o}} &= \frac{\partial}{\partial u_{w=o}}\left[-\log\left(\hat{y}_o\right)\right] \\
&= \frac{\partial}{\partial u_{w=o}}\left[-\log\left(\frac{e^{\mathbf{u_0}^T \mathbf{v_c}}}{\sum_{w \in \text{Vocab}} e^{\mathbf{u_w}^T \mathbf{v_c}}}\right)\right] \\
&= -\frac{\partial}{\partial u_{w=o}}\left[\mathbf{u_o}^T \mathbf{v_c}\right] + \frac{\partial}{\partial u_{w=o}}\left[\log\left(\sum_{w \in \text{Vocab}} e^{\mathbf{u_w}^T \mathbf{v_c}}\right)\right] \\
&= -\left(\mathbf{v_c}\right) + \left(\frac{1}{\sum_{w \in \text{Vocab}} e^{\mathbf{u_w}{}^T \mathbf{v_c}}}\left(e^{\mathbf{u_o}{}^T \mathbf{v_c}} \cdot \mathbf{v_c}\right)\right) \\
&= \mathbf{v_c}\left(\hat{y_o} - 1\right)
\end{aligned}
$$

Case 2 - the outside word vector is any context word but the true one

$$\begin{aligned}
\frac{\partial J_{\text{naive}-\text{softmax}}}{\partial \mathbf{u_{w\neq o}}} &= \frac{\partial}{\partial \mathbf{u_{w\neq o}}}\left[-\log\left(\hat{y}_o\right)\right] \\
&= \frac{\partial}{\partial \mathbf{u_{w\neq o}}}\left[-\log\left(\frac{e^{\mathbf{u_o}^T\mathbf{v_c}}}{\sum_{w\in\text{Vocab}} e^{\mathbf{u_w}^T\mathbf{v_c}}}\right)\right] \\
&= -\frac{\partial}{\partial \mathbf{u_{w\neq o}}}\left[\mathbf{u_o}^T\mathbf{v_c}\right] + \frac{\partial}{\partial \mathbf{u_{w\neq o}}}\left[\log\left(\sum_{w\in\text{Vocab}} e^{\mathbf{u_w}^T\mathbf{v_c}}\right)\right] \\
&= 0 + \left(\frac{1}{\sum_{w\in Vocab} e^{\mathbf{u_w}^T\mathbf{v_c}}}\cdot e^{u_{w\neq o}^T\mathbf{v_c}}\cdot\mathbf{v_c}\right) \\
&= \mathbf{v_c}\cdot\hat{y}_{w\neq o}
\end{aligned}$$

Generally,

$$\frac{\partial J_{\text{naive}-\text{softmax}}}{\partial \mathbf{u_w}} = \mathbf{v_c}(\tilde{y}-y)$$

(d) Compute the derivate of the sigmoid $\sigma(\mathbf{x}) = \frac{1}{1+e^{-\mathbf{x}}} = \frac{e^{\mathbf{x}}}{e^{\mathbf{x}}+1}$ w.r.t. $\mathbf{x}$, where $\mathbf{x}$ is a vector.

Solution:

$$\begin{aligned}
\frac{d\sigma}{d\mathbf{x}} &= \frac{d}{d\mathbf{x}}\left[\frac{1}{1+e^{-\mathbf{x}}}\right] \\
&= \frac{d}{d\mathbf{x}}\left[\left(1+e^{-\mathbf{x}}\right)^{-1}\right] \\
&= \left[-\left(1+e^{-\mathbf{x}}\right)^{-2}\right]\left[-e^{-\mathbf{x}}\right] \\
&= \frac{e^{-\mathbf{x}}}{\left(1+e^{-\mathbf{x}}\right)^2} \\
&= \sigma(\mathbf{x})(1-\sigma(\mathbf{x}))
\end{aligned}$$

(e) Compute partial derivatives of $\boldsymbol{J}_{\text{neg-sample}}\left(\boldsymbol{v}_c, o, \boldsymbol{U}\right)$ w.r.t. $\mathbf{v_c}, \mathbf{u_o}, \mathbf{u_k}$, where $k\in[1,K]$. Why this loss function is much more efficient to compute than the naive-softmax loss.

$$\boldsymbol{J}_{\text{neg-sample}}\left(\boldsymbol{v}_c, o, \boldsymbol{U}\right) = -\log\left(\sigma\left(\boldsymbol{u}_o^{\top}\boldsymbol{v}_c\right)\right) - \sum_{k=1}^{K}\log\left(\sigma\left(-\boldsymbol{u}_k^{\top}\boldsymbol{v}_c\right)\right)$$

Solution:

$$\frac{\partial J_{neg-sample}}{\partial \mathbf{v}_c} = -\mathbf{u_o}\left(1-\sigma\left(\mathbf{u_o}^T\mathbf{v_c}\right)\right) + \sum_{k=1}^{K}\mathbf{u_k}\left(1-\sigma\left(-\mathbf{u_k}^T\mathbf{v_c}\right)\right)$$

$$\frac{\partial J_{neg-sample}}{\partial \mathbf{u}_o} = -\mathbf{v_c}\left(1 - \sigma\left(\mathbf{u_o}^T \mathbf{v_c}\right)\right)$$

$$\frac{\partial J_{neg-sample}}{\partial \mathbf{u}_k} = \mathbf{v_c}\left(1 - \sigma\left(-\mathbf{u_k}^T \mathbf{v_c}\right)\right)$$

This loss function is much more efficient to compute than the naive-softmax loss because it takes into account jusk K more sample word vectors $(O(K))$ whereas in the naive-softmax loss we must normalize the probabilities, requiring that we look at all the word vectors in the entire vocabulary $O(|V|)$