Updated on: 2$^{\text{nd}}$ March, 2023

Question (a)

Note that

$$\boldsymbol{y}_w = \begin{cases} 1 & \text{if } w = o \\ 0 & \text{if } w \neq o \end{cases}$$

Therefore,

$$\begin{aligned} -\sum_{w \in \text{Vocab}} \boldsymbol{y}_w \log(\hat{\boldsymbol{y}}_w) &= -(1)\log(\hat{\boldsymbol{y}}_{\boldsymbol{o}}) \\ &= -\log(\hat{\boldsymbol{y}}_{\boldsymbol{o}}) \end{aligned}$$

Question (b)

Part (i)

$$\begin{aligned} J_{naive-softmax}(\boldsymbol{v}_c, o, \boldsymbol{U}) &= -\log(\hat{\boldsymbol{y}}_{\boldsymbol{o}}) \\ &= -\log\left(\frac{\exp(\boldsymbol{u}_o^\top \boldsymbol{v}_{\boldsymbol{c}})}{\sum_{k \in |\text{Vocab}|} \exp(\boldsymbol{u}_k^\top \boldsymbol{v}_{\boldsymbol{c}})}\right) \\ &= -\log\big(\exp(\boldsymbol{u}_o^\top \boldsymbol{v}_{\boldsymbol{c}})\big) + \log\left(\sum_{k \in |\text{Vocab}|} \exp(\boldsymbol{u}_k^\top \boldsymbol{v}_{\boldsymbol{c}})\right) \\ &= -\boldsymbol{u}_o^\top \boldsymbol{v}_{\boldsymbol{c}} + \log\left(\sum_{k \in |\text{Vocab}|} \exp(\boldsymbol{u}_k^\top \boldsymbol{v}_{\boldsymbol{c}})\right) \end{aligned}$$

$$\begin{aligned} \frac{\partial J_{naive-softmax}(\boldsymbol{v}_c, o, \boldsymbol{U})}{\partial \boldsymbol{v}_{\boldsymbol{c}}} &= \frac{\partial}{\partial \boldsymbol{v}_{\boldsymbol{c}}}\left[-\boldsymbol{u}_o^\top \boldsymbol{v}_{\boldsymbol{c}} + \log\left(\sum_{k \in |\text{Vocab}|} \exp(\boldsymbol{u}_k^\top \boldsymbol{v}_{\boldsymbol{c}})\right)\right] \\ &= -\boldsymbol{u}_{\boldsymbol{o}} + \frac{\sum_{k \in |\text{Vocab}|} \exp(\boldsymbol{u}_k^\top \boldsymbol{v}_{\boldsymbol{c}})\boldsymbol{u}_{\boldsymbol{k}}}{\sum_{k \in |\text{Vocab}|} \exp(\boldsymbol{u}_k^\top \boldsymbol{v}_{\boldsymbol{c}})} \\ &= -\boldsymbol{u}_{\boldsymbol{o}} + \sum_{k \in |\text{Vocab}|}\left(\frac{\exp(\boldsymbol{u}_k^\top \boldsymbol{v}_{\boldsymbol{c}})}{\sum_{k \in |\text{Vocab}|} \exp(\boldsymbol{u}_k^\top \boldsymbol{v}_{\boldsymbol{c}})}\right)\boldsymbol{u}_{\boldsymbol{k}} \\ &= -\boldsymbol{u}_{\boldsymbol{o}} + \sum_{k \in |\text{Vocab}|}(\hat{\boldsymbol{y}}_{\boldsymbol{k}})\,\boldsymbol{u}_{\boldsymbol{k}} \\ &= \sum_{k \in |\text{Vocab}|} \boldsymbol{y}_{\boldsymbol{k}}(-\boldsymbol{u}_{\boldsymbol{k}}) + \sum_{k \in |\text{Vocab}|}(\hat{\boldsymbol{y}}_{\boldsymbol{k}})\,\boldsymbol{u}_{\boldsymbol{k}} \\ &= \sum_{k \in |\text{Vocab}|}(\hat{\boldsymbol{y}}_{\boldsymbol{k}} - \boldsymbol{y}_{\boldsymbol{k}})\,\boldsymbol{u}_{\boldsymbol{k}} \\ &= (\hat{\boldsymbol{y}}^\top - \boldsymbol{y}^\top)\boldsymbol{U}^\top \qquad \text{(a row vector)} \\ &= \boldsymbol{U}(\hat{\boldsymbol{y}} - \boldsymbol{y}) \qquad \text{(a colume vector)} \end{aligned}$$

Note that a column vector is left as the answer to follow the shape convention.

Part (ii)

If the gradient computed in (b)(i) is equal to zero, i.e. the equation $\boldsymbol{U}(\hat{\boldsymbol{y}} - \boldsymbol{y}) = 0$ is satisfied, then the vector $(\hat{\boldsymbol{y}} - \boldsymbol{y})$ lies in the nullspace of the matrix $\boldsymbol{U}$. Since the zero vector is always in the nullspace, a trival solution to the equation is $\hat{\boldsymbol{y}} - \boldsymbol{y} = 0$, i.e. the predicted word vector is correct.

Part (iii)

$$\begin{aligned}\frac{\partial J_{naive-softmax}(\boldsymbol{v}_c, o, \boldsymbol{U})}{\partial \boldsymbol{v_c}} &= \boldsymbol{U}(\hat{\boldsymbol{y}} - \boldsymbol{y}) \\ &= \boldsymbol{U}\hat{\boldsymbol{y}} - \boldsymbol{U}\boldsymbol{y} \\ &= \text{predicted word vector} - \text{true word vector}\end{aligned}$$

When the gradient is subtractedf from the word vecter $\boldsymbol{v_c}$, $\boldsymbol{v_c}$ is improved towards the true word vector since the true word vector is added to $\boldsymbol{v_c}$. On the other hand, the updated vector is moved away from the predicted word vector since the predicted word vector is subtracted from $\boldsymbol{v_c}$.

Part (iv)

Consider two word vectors $\mathbf{u_x}$ and $\mathbf{u_y}$ of the two different words $\mathbf{x}$ and $\mathbf{y}$, where $\mathbf{u_x} = \alpha\mathbf{u_y}$ and α is some scalar.

If the two words are opposite in meanings, it is expected that these two word vectors are in opposite direction, i.e. $\alpha < 0$. In such case, since the L2 normalization preserves the direction of the word vectors, one can still distinguish the two word vectors with their directions.

In the case where the two words have the same sign ($\alpha > 0$) but with different extent (e.g. $\mathbf{x}$ is much positive than $\mathbf{y}$), one would expect such information is revealed from the vector magnitudes. However, the L2 normalization produces vectors with the same magnitudes. Information of word vectors carried by the vector magnitude is lost after performing L2 normalization.

Question (c)

$$J_{naive-softmax}(\boldsymbol{v}_c, o, \boldsymbol{U}) = -\boldsymbol{u}_o^\top \boldsymbol{v_c} + \log\left(\sum_{k\in|\text{Vocab}|} \exp\left(\boldsymbol{u}_k^\top \boldsymbol{v_c}\right)\right)$$

When $w \neq o$,

$$\begin{aligned}\frac{\partial J_{naive-softmax}(\boldsymbol{v}_c, o, \boldsymbol{U})}{\partial \boldsymbol{u_{w\neq o}}} &= \frac{\partial}{\partial \boldsymbol{u_{w\neq o}}}\left[-\boldsymbol{u}_o^\top \boldsymbol{v_c} + \log\left(\sum_{k\in|\text{Vocab}|} \exp\left(\boldsymbol{u}_k^\top \boldsymbol{v_c}\right)\right)\right] \\ &= 0 + \frac{\exp\left(\boldsymbol{u}_{w\neq o}^\top \boldsymbol{v_c}\right)\boldsymbol{v_c}}{\sum_{k\in|\text{Vocab}|} \exp\left(\boldsymbol{u}_k^\top \boldsymbol{v_c}\right)} \\ &= \left(\frac{\exp\left(\boldsymbol{u}_{w\neq o}^\top \boldsymbol{v_c}\right)}{\sum_{k\in|\text{Vocab}|} \exp\left(\boldsymbol{u}_k^\top \boldsymbol{v_c}\right)}\right)\boldsymbol{v_c} \\ &= \hat{\boldsymbol{y}}_{\boldsymbol{w\neq o}}\boldsymbol{v_c}\end{aligned}$$

When $w = o$,

$$\begin{aligned}
\frac{\partial J_{naive-softmax}(\boldsymbol{v}_c, o, \boldsymbol{U})}{\partial \boldsymbol{u_{w=o}}} &= \frac{\partial}{\partial \boldsymbol{u_{w=o}}}\left[-\boldsymbol{u}_o^\top \boldsymbol{v_c} + \log\left(\sum_{k\in|\mathsf{Vocab}|} \exp\left(\boldsymbol{u}_k^\top \boldsymbol{v_c}\right)\right)\right] \\
&= -\boldsymbol{v_c} + \frac{\exp\left(\boldsymbol{u}_{w=o}^\top \boldsymbol{v_c}\right)\boldsymbol{v_c}}{\sum_{k\in|\mathsf{Vocab}|}\exp\left(\boldsymbol{u}_k^\top \boldsymbol{v_c}\right)} \\
&= -\boldsymbol{v_c} + \left(\frac{\exp\left(\boldsymbol{u}_{w=o}^\top \boldsymbol{v_c}\right)}{\sum_{k\in|\mathsf{Vocab}|}\exp\left(\boldsymbol{u}_k^\top \boldsymbol{v_c}\right)}\right)\boldsymbol{v_c} \\
&= -\boldsymbol{v_c} + \boldsymbol{\hat{y}_{w=o}}\boldsymbol{v_c} \\
&= (\boldsymbol{\hat{y}_{w=o}} - 1)\boldsymbol{v_c}
\end{aligned}$$

Therefore,

$$\frac{\partial J_{naive-softmax}(\boldsymbol{v}_c, o, \boldsymbol{U})}{\partial \boldsymbol{u_w}} = \begin{cases} \boldsymbol{\hat{y}_w}\boldsymbol{v_c} & \text{if } w \neq o \\ (\boldsymbol{\hat{y}_w} - 1)\boldsymbol{v_c} & \text{if } w = o \end{cases}$$

Question (d)

$$\frac{\partial J_{naive-softmax}(\boldsymbol{v}_c, o, \boldsymbol{U})}{\partial \boldsymbol{U}} = \begin{pmatrix} \frac{\partial J(\boldsymbol{v}_c, o, \boldsymbol{U})}{\partial \boldsymbol{u_1}} & \frac{\partial J(\boldsymbol{v}_c, o, \boldsymbol{U})}{\partial \boldsymbol{u_2}} & \cdots & \frac{\partial J(\boldsymbol{v}_c, o, \boldsymbol{U})}{\partial \boldsymbol{u}_{|\mathsf{Vocab}|}} \end{pmatrix}$$

Question (e)

$$\begin{aligned}
f(x) &= \max(\alpha x, x) \\
&= \begin{cases} \alpha x & \text{if } x < 0 \\ x & \text{if } x > 0 \end{cases} \\
\frac{df(x)}{dx} &= \begin{cases} \alpha & \text{if } x < 0 \\ 1 & \text{if } x > 0 \end{cases}
\end{aligned}$$

Question (f)

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

Question (g)

Part (i)

$$\begin{aligned}\frac{d}{dx}\log\sigma(x) &= \frac{1}{\sigma(x)}\left(\frac{d}{dx}\sigma(x)\right)\\ &= \frac{1}{\sigma(x)}\left(\sigma(x)\left(1-\sigma(x)\right)\right)\\ &= 1-\sigma(x)\end{aligned}$$

$$J_{\text{negative-sample}}(\boldsymbol{v}_c, o, \boldsymbol{U}) = -\log(\sigma(\boldsymbol{u}_o^\top \boldsymbol{v_c})) - \sum_{s=1}^{K}\log(\sigma(-\boldsymbol{u}_{w_s}^\top \boldsymbol{v_c}))$$

$$\begin{aligned}\frac{\partial J_{\text{negative-sample}}(\boldsymbol{v}_c, o, \boldsymbol{U})}{\partial \boldsymbol{v_c}} &= (\sigma(\boldsymbol{u}_o^\top \boldsymbol{v_c})-1)\frac{\partial}{\partial \boldsymbol{v_c}}(\boldsymbol{u}_o^\top \boldsymbol{v_c}) + \sum_{s=1}^{K}(\sigma(-\boldsymbol{u}_{w_s}^\top \boldsymbol{v_c})-1)\frac{\partial}{\partial \boldsymbol{v_c}}(-\boldsymbol{u}_{w_s}^\top \boldsymbol{v_c})\\ &= (\sigma(\boldsymbol{u}_o^\top \boldsymbol{v_c})-1)\boldsymbol{u}_o - \sum_{s=1}^{K}(\sigma(-\boldsymbol{u}_{w_s}^\top \boldsymbol{v_c})-1)\boldsymbol{u}_{w_s}\end{aligned}$$

$$\begin{aligned}\frac{\partial J_{\text{negative-sample}}(\boldsymbol{v}_c, o, \boldsymbol{U})}{\partial \boldsymbol{u_o}} &= (\sigma(\boldsymbol{u}_o^\top \boldsymbol{v_c})-1)\frac{\partial}{\partial \boldsymbol{u_o}}(\boldsymbol{u}_o^\top \boldsymbol{v_c}) + 0\\ &= (\sigma(\boldsymbol{u}_o^\top \boldsymbol{v_c})-1)\boldsymbol{v}_c\end{aligned}$$

$$\begin{aligned}\frac{\partial J_{\text{negative-sample}}(\boldsymbol{v}_c, o, \boldsymbol{U})}{\partial \boldsymbol{u_{w_s}}} &= 0 + (\sigma(-\boldsymbol{u}_{w_s}^\top \boldsymbol{v_c})-1)\frac{\partial}{\partial \boldsymbol{u_{w_s}}}(-\boldsymbol{u}_{w_s}^\top \boldsymbol{v_c})\\ &= -(\sigma(-\boldsymbol{u}_{w_s}^\top \boldsymbol{v_c})-1)\boldsymbol{v}_c\end{aligned}$$

Part (ii)

$$\boldsymbol{U_{o,\{w_1,\ldots,w_K\}}} = [\boldsymbol{u_o}, -\boldsymbol{u_{w_1}}, \cdots, -\boldsymbol{u_{w_K}}]$$

Denote

$$\begin{aligned}\boldsymbol{\lambda} &= \sigma\left(\boldsymbol{U}_{\boldsymbol{o},\{\boldsymbol{w_1},\cdots,\boldsymbol{w_K}\}}^\top \boldsymbol{v_c}\right) - \boldsymbol{1}\\ &= \begin{pmatrix}\sigma(\boldsymbol{u}_o^\top \boldsymbol{v_c})-1\\ \sigma(-\boldsymbol{u}_{w_1}^\top \boldsymbol{v_c})-1\\ \vdots\\ \sigma(-\boldsymbol{u}_{w_K}^\top \boldsymbol{v_c})-1\end{pmatrix}\\ &= \begin{pmatrix}\lambda_o\\ \lambda_1\\ \vdots\\ \lambda_K\end{pmatrix}\end{aligned}$$

Each element of $\boldsymbol{\lambda}$ is used in computing the partial derivatives in part (g)(i). The derivatives are simplified as follows:

$$\begin{aligned}\frac{\partial J_{\text{negative-sample}}(\boldsymbol{v}_c, o, \boldsymbol{U})}{\partial \boldsymbol{v_c}} &= \lambda_o\boldsymbol{u}_o - \sum_{s=1}^{K}\lambda_s\boldsymbol{u}_{w_s}\\ \frac{\partial J_{\text{negative-sample}}(\boldsymbol{v}_c, o, \boldsymbol{U})}{\partial \boldsymbol{u_o}} &= \lambda_o\boldsymbol{v}_c\\ \frac{\partial J_{\text{negative-sample}}(\boldsymbol{v}_c, o, \boldsymbol{U})}{\partial \boldsymbol{u_{w_s}}} &= -\lambda_s\boldsymbol{v}_c\end{aligned}$$

Part (iii)

The computation of $J_{\text{naive-softmax}}$ requires computing the dot products between all word vectors with the input vector. The time complexity of the computation is $O(|\text{Vocab}|)$. On the other hand, the computation of $J_{\text{negative-sample}}$ involves only the dot products between K negative sample word vectors and the input vectors, where the time complexity of computation is $O(K)$. Hence, computing the negative sampling loss function is much more efficient than the naive-softmax function.

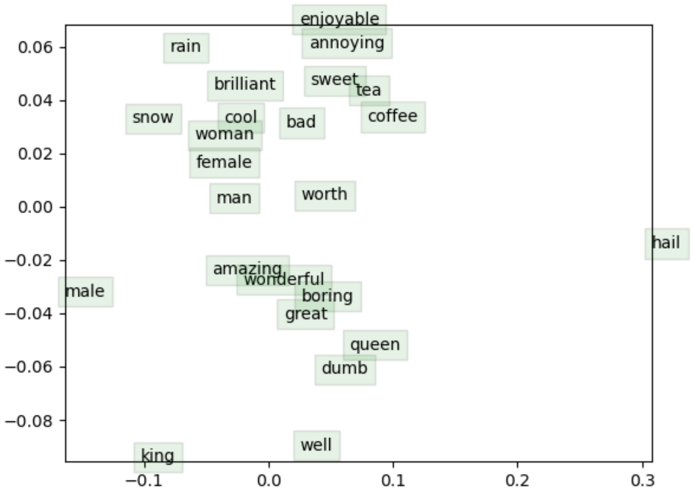
Question (h)

$$\begin{aligned}
\frac{\partial J_{\text{negative-sample}}(\boldsymbol{v}_c, o, \boldsymbol{U})}{\partial \boldsymbol{u}_{\boldsymbol{w_s}}} &= \frac{\partial}{\partial \boldsymbol{u}_{\boldsymbol{w_s}}}\left(-\log(\sigma(\boldsymbol{u}_o^\top \boldsymbol{v}_{\boldsymbol{c}})) - \sum_{k=1}^{K} \log(\sigma(-\boldsymbol{u}_{w_k}^\top \boldsymbol{v}_{\boldsymbol{c}}))\right) \\
&= \frac{\partial}{\partial \boldsymbol{u}_{\boldsymbol{w_s}}}\left(-\log(\sigma(\boldsymbol{u}_o^\top \boldsymbol{v}_{\boldsymbol{c}})) - \sum_{\substack{k=1 \\ w_k \neq w_s}}^{K} \log(\sigma(-\boldsymbol{u}_{w_k}^\top \boldsymbol{v}_{\boldsymbol{c}})) - \sum_{\substack{k=1 \\ w_k = w_s}}^{K} \log(\sigma(-\boldsymbol{u}_{w_k}^\top \boldsymbol{v}_{\boldsymbol{c}}))\right) \\
&= 0 - 0 - \frac{\partial}{\partial \boldsymbol{u}_{\boldsymbol{w_s}}}\left(\sum_{\substack{k=1 \\ w_k = w_s}}^{K} \log(\sigma(-\boldsymbol{u}_{w_k}^\top \boldsymbol{v}_{\boldsymbol{c}}))\right) \\
&= -\sum_{\substack{k=1 \\ w_k = w_s}}^{K} (\sigma(-\boldsymbol{u}_{w_s}^\top \boldsymbol{v}_{\boldsymbol{c}}) - 1)\boldsymbol{v}_c
\end{aligned}$$

Question (i)

$$\begin{aligned}
\frac{\partial J_{\text{skip-gram}}(\boldsymbol{v}_c, w_{t-m}, \cdots, w_{t+m}, \boldsymbol{U})}{\partial \boldsymbol{U}} &= \sum_{\substack{-m \leq j < m \\ j \neq 0}} \frac{\partial J(\boldsymbol{v}_c, w_{t+j}, \boldsymbol{U})}{\partial \boldsymbol{U}} \\
\frac{\partial J_{\text{skip-gram}}(\boldsymbol{v}_c, w_{t-m}, \cdots, w_{t+m}, \boldsymbol{U})}{\partial \boldsymbol{v}_{\boldsymbol{c}}} &= \sum_{\substack{-m \leq j < m \\ j \neq 0}} \frac{\partial J(\boldsymbol{v}_c, w_{t+j}, \boldsymbol{U})}{\partial \boldsymbol{v}_{\boldsymbol{c}}} \\
\frac{\partial J_{\text{skip-gram}}(\boldsymbol{v}_c, w_{t-m}, \cdots, w_{t+m}, \boldsymbol{U})}{\partial \boldsymbol{v}_{\boldsymbol{w}}} &= 0 \qquad \text{where} \quad w \neq c
\end{aligned}$$

Coding: Implementing word2vec Part (c)



In the plot, some synonyms ("amazing", "wonderful", "great") and antonyms ("enjoyable", "annoying") are cluster together. Terms with similar usage such as ("tea", "coffee"), ("rain", "snow") and ("female", "woman") also group together. Lastly, the linear relationships of word vectors, i.e. vec("king") − vec("male") + vec("female") ≈ vec("queen"), is observed.