CS224n Assignment #2: word2vec (44 Points)

1 Written: Understanding `word2vec` (26 points)

(a) (3 points) Show that the naive-softmax loss given in Equation (2) is the same as the cross-entropy loss between y and $\hat{y}$; i.e., show that

$$-\sum_{w \in \text{Vocab}} y_w \log \hat{y}_w = -\log \hat{y}_o$$

Your answer should be one line.

💡 **Answer:**

The true empirical distribution (i.e., the ground truth) y is a one-hot vector where $y_w = 1$ when $w = o$ and $y_w = 0$ when $w \neq o$. Mathematically,

$$y_w = \begin{cases} 1 \text{ if } w = o \\ 0 \text{ if } w \neq o \end{cases}$$

As such, considering the cross-entropy loss $J_{\text{cross-entropy}}$,

$$J_{\text{cross-entropy}}(y, \hat{y}) = -\sum_{w \in \text{Vocab}} y_w \log(\hat{y}_w)$$

$$= -[y_1 \log(\hat{y}_1) + ... + y_o \log(\hat{y}_o) + ... + y_w \log(\hat{y}_w)]$$

$$= -y_o \log(\hat{y}_o)$$

$$= -\log(\hat{y}_o)$$

$$= -\log P(O = o | C = c)$$

$$= \boxed{J_{\text{naive-softmax}}(v_c, o, U)}$$

(b) (5 points) Compute the partial derivative of $J_{\text{naive-softmax}}(v_c, o, U)$ with respect to v_c. Please write your answer in terms of y, $\hat{y}$, and U. Note that in this course, we expect your final answers to follow the shape convention. This means that the partial derivative of any function $f(x)$ with respect to x should have the same shape as x. For this subpart, please present your answer in vectorized form. In particular, you may not refer to specific elements of y, $\hat{y}$, and U in your final answer (such as y_1, y_2, ...).

💡 **Answer:**

$$\frac{\delta}{\delta v_c} J_{\text{naive-softmax}}(v_c, o, U) = -\frac{\delta}{\delta v_c} \log P(O = o | C = c)$$

$$= -\frac{\delta}{\delta v_c} \log \frac{\exp{(u_o^T v_c)}}{\sum_{w \in \text{Vocab}} \exp{(u_w^T v_c)}}$$

$$= -\frac{\delta}{\delta v_c} \log \exp{(u_o^T v_c)} + \frac{\delta}{\delta v_c} \log \sum_{w \in \text{Vocab}} \exp{(u_w^T v_c)}$$

$$= -\frac{\delta}{\delta v_c} u_o^T v_c + \frac{1}{\sum_{w \in \text{Vocab}} \exp{(u_w^T v_c)}} \frac{\delta}{\delta v_c} \sum_{w \in \text{Vocab}} \exp{(u_w^T v_c)}$$

$$= -u_o + \frac{1}{\sum_{w \in \text{Vocab}} \exp{(u_w^T v_c)}} \sum_{w \in \text{Vocab}} \frac{\delta}{\delta v_c} \exp{(u_w^T v_c)}$$

$$= -u_o + \frac{1}{\sum_{w \in \text{Vocab}} \exp{(u_w^T v_c)}} \sum_{w \in \text{Vocab}} \exp{(u_w^T v_c)} \frac{\delta}{\delta v_c} u_w^T v_c$$

$$= -u_o + \frac{\sum_{w \in \text{Vocab}} \exp{(u_w^T v_c)} u_w}{\sum_{w \in \text{Vocab}} \exp{(u_w^T v_c)}}$$

Since $y_w = \begin{cases} 1 \text{ if } w = o \\ 0 \text{ if } w \neq o \end{cases}$, we can write u_o as $\sum_{w \in \text{Vocab}} y_w u_w$. As such,

$$= -\sum_{w \in \text{Vocab}} y_w u_w + \sum_{w \in \text{Vocab}} \frac{\exp{(u_w^T)} v_c}{\sum_{k \in \text{Vocab}} \exp{(u_k^T v_c)}} u_w$$

Since $\frac{\exp{(u_w^T)} v_c}{\sum_{k \in \text{Vocab}} \exp{(u_k^T v_c)}}$ is the conditional probability distribution, $\hat{y}_w$,

$$= -\sum_{w \in \text{Vocab}} y_w u_w + \sum_{w \in \text{Vocab}} \hat{y}_w u_w$$

$$= \sum_{w \in \text{Vocab}} (-y_w u_w + \hat{y}_w u_w)$$

$$= \sum_{w \in \text{Vocab}} u_w (-y_w + \hat{y}_w)$$

$$= \sum_{w \in \text{Vocab}} u_w (\hat{y}_w - y_w)$$

Note that y is a one-hot vector with 1 at word o and 0 at all other positions. Vectorizing the above equation in terms of y, $\hat{y}$, and U gives us,

$$\boxed{\frac{\delta}{\delta v_c} J_{\text{naive-softmax}}(v_c, o, U) = U(\hat{y} - y)}$$

(c) (5 points) Compute the partial derivatives of $J_{\text{naive-softmax}}(v_c, o, U)$ with respect to each of the ‘outside’ word vectors, u_w’s. There will be two cases: when $w = o$, the true ‘outside’ word vector, and $w \neq o$, for all other words. Please write your answer in terms of y, $\hat{y}$, and v_c. In this subpart, you may use specific elements within these terms as well, such as (such as y_1, y_2, ...).

💡 **Answer:**

$$\frac{\delta}{\delta u_w} J_{\text{naive-softmax}}(v_c, o, U) = -\frac{\delta}{\delta u_w} \log P(O = o|C = c)$$

$$= -\frac{\delta}{\delta u_w} \log \frac{\exp u_o^T v_c}{\sum_{w \in \text{Vocab}} \exp(u_w^T v_c)}$$

$$= -\frac{\delta}{\delta u_w} \log \exp(u_o^T v_c) + \frac{\delta}{\delta u_w} \log \sum_{w \in \text{Vocab}} \exp(u_w^T v_c)$$

When $w \neq o:$

$$\frac{\delta}{\delta u_w} J_{\text{naive-softmax}}(v_c, o, U)$$

$$= \frac{\delta}{\delta u_w} \log \sum_{w \in \text{Vocab}} \exp(u_w^T v_c)$$

$$= \frac{1}{\sum_{k \in \text{Vocab}} \exp(u_k^T v_c)} \frac{\delta}{\delta u_w} \sum_{\substack{x \in \text{Vocab} \\ x \neq o}} \exp(u_w^T v_c)$$

$$= \frac{1}{\sum_{k \in \text{Vocab}} \exp(u_k^T v_c)} \frac{\delta}{\delta u_w} \exp(u_x^T v_c)$$

$$= \frac{\exp(u_w^T v_c)}{\sum_{k \in \text{Vocab}} \exp(u_k^T v_c)} v_c$$

Since $\frac{\exp(u_w^T v_c)}{\sum_{k \in \text{Vocab}} \exp(u_k^T v_c)}$ is the conditional probability distribution, $\hat{y}_w$,

$$\boxed{\frac{\delta}{\delta u_w} J_{\text{naive-softmax}}(v_c, o, U) = \hat{y}_w v_c}$$

When $w = o:$

$$\frac{\delta}{\delta u_o} J_{\text{naive-softmax}}(v_c, o, U)$$

$$= -\frac{\delta}{\delta u_o} \log \exp(u_o^T v_c) + \frac{\delta}{\delta u_o} \log \sum_{w \in \text{Vocab}} \exp(u_w^T v_c)$$

$$= -v_c + \frac{\exp(u_o^T v_c)}{\sum_{k \in \text{Vocab}} \exp(u_k^T v_c)} v_c$$

$$= (\frac{\exp(u_o^T v_c)}{\sum_{k \in \text{Vocab}} \exp(u_k^T v_c)} - 1) v_c$$

Since $\frac{\exp\left(u_o^T v_c\right)}{\sum_{k \in \text{Vocab}} \exp\left(u_k^T v_c\right)}$ is the conditional probability distribution, $\hat{y}_o$,

$$\boxed{\frac{\delta}{\delta u_o} J_{\text{naive-softmax}}(v_c, o, U) = (\hat{y}_o - 1)v_c}$$

Consolidating both cases,

$$\frac{\delta J}{\delta u_w} = \begin{cases} (\hat{y}_o - 1)v_c \text{ if } w = o \\ \hat{y}_w v_c \text{ otherwise} \end{cases}$$

(d) (1 point) Compute the partial derivative of $J_{\text{naive-softmax}}(v_c, o, U)$ with respect to U. Please write your answer in terms of $\frac{\delta J(v_c,o,U)}{\delta u_1}, \frac{\delta J(v_c,o,U)}{\delta u_2}, ..., \frac{\delta J(v_c,o,U)}{\delta u_{|Vocab|}}$. The solution should be one or two lines long.

💡 **Answer:**

The derivative of a scalar y by a matrix A is given by,

$$\frac{\delta y}{\delta A_{m \times n}} = \begin{bmatrix} \frac{\delta y}{\delta A_{11}} & \frac{\delta y}{\delta A_{12}} & \cdots & \frac{\delta y}{\delta A_{1n}} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\delta y}{\delta A_{m1}} & \frac{\delta y}{\delta A_{m2}} & \cdots & \frac{\delta y}{\delta A_{mn}} \end{bmatrix}$$

Given u_w represents the vector for "outside" word w, the derivative of $J_{\text{naive-softmax}}$ (which is a scalar) by U (which is a matrix) is,

$$\frac{\delta J_{\text{naive-softmax}}(v_c, o, U)}{\delta U} = \begin{bmatrix} \frac{\delta J}{\delta u_1} & \frac{\delta J}{\delta u_2} & \cdots & \frac{\delta J}{\delta u_{\text{Vocab}}} \end{bmatrix}$$

where,

$$\frac{\delta J}{\delta u_w} = \begin{cases} (\hat{y} - 1)v_c \text{ if } w = o \\ \hat{y} v_c \text{ if } w \neq o \end{cases}$$

(e) (3 Points) The sigmoid function is given by Equation 4:

$$\sigma(x) = \frac{1}{1 + e^{-x}} = \frac{e^x}{e^x + 1}$$

Please compute the derivative of $\sigma(x)$ with respect to x, where x is a scalar. Hint: you may want to write your answer in terms of $\sigma(x)$.

💡 **Answer**:

$$\frac{\delta \sigma}{\delta x} = \frac{\delta}{\delta x}\left[\frac{e^x}{e^x + 1}\right]$$

Using the quotient rule,

$$= \frac{e^x(e^x + 1) - e^{2x}}{(e^x + 1)^2}$$

$$= \frac{e^x}{(e^x + 1)^2}$$

$$= \frac{e^x}{e^x + 1}\frac{1}{e^x + 1}$$

$$= \sigma(x) \cdot \frac{1}{e^x + 1}$$

We can multiply with $\frac{e^{-x}}{e^{-x}}$ which is essentially equivalent to 1,

$$= \sigma(x) \cdot \left[\frac{1}{e^x + 1}\right] \times \frac{e^{-x}}{e^{-x}}$$

$$= \sigma(x) \cdot \frac{e^{-x}}{1 + e^{-x}}$$

$$= \sigma(x) \cdot \left[\frac{1 + e^{-x}}{1 + e^{-x}} - \frac{1}{1 + e^{-x}}\right]$$

$$= \boxed{\sigma(x)(1 - \sigma(x))}$$

(f) (4 points) Now we shall consider the Negative Sampling loss, which is an alternative to the Naive Softmax loss. Assume that K negative samples (words) are drawn from the vocabulary. For simplicity of notation we shall refer to them as $w_1, w_2, ..., w_K$ and their outside vectors as $u_1, ..., u_K$. For this question, assume that the K negative samples are distinct. In other words, $i \neq j$ implies $w_i \neq w_j$ for $i, j \in \{1, ..., K\}$. Note that $o \notin \{w_1, ..., w_K\}$. For a center word c and an outside word o, the negative sampling loss function is given by:

$$J_{\text{neg-sample}}(v_c, o, U) = -\log\left(\sigma(u_o^T v_c)\right) - \sum_{k=1}^{K} \log\left(\sigma(-u_k^T v_c)\right)$$

for a sample $w_1, ..., w_K$, where $\sigma(\cdot)$ is the sigmoid function.

Please repeat parts (b) and (c), computing the partial derivatives of $J_{\text{neg-sample}}$ with respect to v_c, with respect to u_o, and with respect to a negative sample u_k. Please write your answers in terms of the vectors u_o, v_c and u_k, where $k \in [1, K]$. After you've done this, describe with one sentence why this loss function is much more efficient to compute than the naive-softmax loss. Note, you should be able to use your solution to part (e) to help compute the necessary gradients here.

💡 **Answer:**

The loss function contains two terms:

(i) The first term is the log of the probability that the center word and true outside word came from the corpus.

(ii) The second term is the sum of the logs of the probabilities that the center word and outside context words did not come from the corpus.

1. Computing the partial derivatives of $J_{\text{neg-sample}}$ w.r.t v_c which is the word vector of the center word, c.

$$\frac{\delta J_{\text{neg-sample}}(v_c, o, U)}{\delta v_c} = \frac{\delta}{\delta v_c}\left[-\log\left(\sigma(u_o^T v_c)\right) - \sum_{k=1}^{K} \log\left(\sigma(-u_k^T v_c)\right)\right]$$

$$= \frac{\delta}{\delta v_c}\left[-log(\sigma(u_o^T v_c))\right] - \frac{\delta}{\delta v_c}\left[\sum_{k=1}^{K} \log\left(\sigma(-u_k^T v_c)\right)\right]$$

$$= \frac{\delta}{\delta v_c}\left[-log(\sigma(u_o^T v_c))\right] - \sum_{k=1}^{K}\left[\frac{\delta}{\delta v_c} \log\left(\sigma(-u_k^T v_c)\right)\right]$$

$$= \left(-\frac{1}{\sigma(u_o^T v_c)}\right)\sigma(u_o^T v_c)(1-\sigma(u_o^T v_c))u_o - \sum_{k=1}^{K} \frac{u_k \sigma(-u_k^T v_c)(1-\sigma(-u_k^T v_c))}{\sigma(-u_k^T v_c)}$$

$$= -u_o(1-\sigma(u_o^T v_c)) - \sum_{k=1}^{K} u_k(1-\sigma(-u_k^T v_c))$$

$$= \boxed{u_o(\sigma(u_o^T v_c)-1) - \sum_{k=1}^{K} u_k(\sigma(-u_k^T v_c)-1)}$$

2. Computing the partial derivatives of $J_{\text{neg-sample}}$ w.r.t u_o which is the word vector of the outside word, o.

$$\frac{\delta J_{\text{neg-sample}}(v_c, o, U)}{\delta u_o} = \frac{\delta}{\delta u_o}\left[-\log\left(\sigma(u_o^T v_c)\right) - \sum_{k=1}^{K} \log\left(\sigma(-u_k^T v_c)\right)\right]$$

$$= \frac{\delta}{\delta u_o}\left[-log(\sigma(u_o^T v_c))\right] - \frac{\delta}{\delta u_o}\left[\sum_{k=1}^{K} \log\left(\sigma(-u_k^T v_c)\right)\right]$$

Since $o \notin \{w_1, ..., w_K\}$, $\frac{\delta}{\delta u_o}\left[\sum_{k=1}^{K} \log\left(\sigma(-u_k^T v_c)\right)\right] = 0$. As such,

$$\frac{\delta J_{\text{neg-sample}}}{\delta u_o} = \frac{\delta}{\delta u_o}\left[-log(\sigma(u_o^T v_c))\right] - 0$$

Using part (e), $\frac{\delta \sigma}{\delta x} = \sigma(x)(1-\sigma(x))$. Hence,

$$\frac{\delta J_{\text{neg-sample}}(v_c, o, U)}{\delta u_o} = \left(-\frac{1}{\sigma(u_o^T v_c)}\right)\sigma(u_o^T v_c)(1 - \sigma(u_o^T v_c))v_c$$

$$= -v_c(1 - \sigma(u_o^T v_c))$$

$$= \boxed{= v_c(\sigma(u_o^T v_c) - 1)}$$

3. Computing the partial derivatives of $J_{\text{neg-sample}}$ w.r.t u_k which is the word vector for one of the K negative samples.

$$\frac{\delta J_{\text{neg-sample}}(v_c, o, U)}{\delta u_k} = \frac{\delta}{\delta u_k}\left[-\log\left(\sigma(u_o^T v_c)\right) - \sum_{k=1}^{K}\log\left(\sigma(-u_k^T v_c)\right)\right]$$

$$= \frac{\delta}{\delta u_k}\left[-log(\sigma(u_o^T v_c))\right] - \frac{\delta}{\delta u_k}\left[\sum_{k=1}^{K}\log\left(\sigma(-u_k^T v_c)\right)\right]$$

Since $o \notin \{w_1, ..., w_K\}$, $\frac{\delta}{\delta u_k}\left[-log(\sigma(u_o^T v_c))\right] = 0$. As such,

$$\frac{\delta J_{\text{neg-sample}}(v_c, o, U)}{\delta u_k} = 0 - \frac{\delta}{\delta u_k}\left[\sum_{k=1}^{K}\log\left(\sigma(-u_k^T v_c)\right)\right]$$

Since the derivative of a sum is the sum of derivatives,

$$\frac{\delta J_{\text{neg-sample}}(v_c, o, U)}{\delta u_k} = -\sum_{k=1}^{K}\left[\frac{\delta}{\delta u_k}\log\left(\sigma(-u_k^T v_c)\right)\right]$$

Now, derivatives of all the terms with u_w where $w \neq k$ are 0 while the derivative of the term with u_w where $w = k$ remains.

Using part (e), $\frac{\delta\sigma}{\delta x} = \sigma(x)(1 - \sigma(x))$. Hence,

$$\frac{\delta J_{\text{neg-sample}}(v_c, o, U)}{\delta u_k} = -\frac{v_c\sigma(-u_k^T v_c)(1 - \sigma(-u_k^T v_c))}{\sigma(-u_k^T v_c)}$$

$$= \boxed{v_c(\sigma(-u_k^T v_c) - 1), \forall k \in [1, K]}$$

By using the negative sampling loss, we only need to go through $O(K)$ samples, while the naive-softmax loss requires traversing through the whole vocabulary, which is is typically much greater than K.

(g) (2 point) Now we will repeat the previous exercise, but without the assumption that the K sampled words are distinct. Assume that K negative samples (words) are drawn from the vocabulary. For simplicity of notation we shall refer to them as $w_1, w_2, ..., w_K$ and their outside vectors as $u_1, ..., u_K$. In this question, you may not assume that the words are distinct. In other words, $w_i = w_j$ may be true when $i \neq j$ is true. Note that $o \notin \{w_1, ..., w_K\}$. For a center word c and an outside word o, the negative sampling loss function is given by:

$$J_{\text{neg-sample}}(v_c, o, U) = -\log\left(\sigma(u_o^T v_c)\right) - \sum_{k=1}^{K} \log\left(\sigma(-u_k^T v_c)\right)$$

for a sample $w_1, ..., w_K$, where $\sigma(\cdot)$ is the sigmoid function.

Compute the partial derivative of $J_{\text{neg-sample}}$ with respect to a negative sample u_k. Please write your answers in terms of the vectors v_c and u_k, where $k \in [1, K]$. Hint: break up the sum in the loss function into two sums: a sum over all sampled words equal to u_k and a sum over all sampled words not equal to u_k.

💡 **Answer:**

$$\frac{\delta J_{\text{neg-sample}}(v_c, o, U)}{\delta u_k} = \frac{\delta}{\delta u_k}\left[-\log\left(\sigma(u_o^T v_c)\right) - \sum_{k=1}^{K} \log\left(\sigma(-u_k^T v_c)\right)\right]$$

$$= \frac{\delta}{\delta u_k}\left[-log(\sigma(u_o^T v_c))\right] - \frac{\delta}{\delta u_k}\left[\sum_{k=1}^{K} \log\left(\sigma(-u_k^T v_c)\right)\right]$$

Since $o \notin \{w_1, ..., w_K\}$, $\frac{\delta}{\delta u_k}\left[-log(\sigma(u_o^T v_c))\right] = 0$. As such,

$$\frac{\delta J_{\text{neg-sample}}(v_c, o, U)}{\delta u_k} = 0 - \frac{\delta}{\delta u_k}\left[\sum_{k=1}^{K} \log\left(\sigma(-u_k^T v_c)\right)\right]$$

Note that K negative samples, $i \in [1, K]$, were drawn from the vocabulary which cannot be assumed to be distinct. As such, let's break up the sum in the loss function into two sums:

1. sum over all sampled words w_i equal to w_k and,
2. sum over all sampled words w_i not equal to w_k

Further, note that here we are iterating over the indices of the words w instead of indices of the vectors u.

$$\frac{\delta J_{\text{neg-sample}}(v_c, o, U)}{\delta u_k} = -\frac{\delta}{\delta u_k}\left[\sum_{i \in \{1,\dots,K\}: w_i = w_k} \log\left(\sigma(-u_i^T v_c)\right) + \sum_{i \in \{1,\dots,K\}: w_i \neq w_k} \log\left(\sigma(-u_{i;w_i \neq w_k}^T v_c)\right)\right]$$

$$= -\left[\sum_{i \in \{1,\dots,K\}: w_i = w_k} \frac{\delta}{\delta u_k} \log\left(\sigma(-u_i^T v_c)\right) + \sum_{i \in \{1,\dots,K\}: w_i \neq w_k} \frac{\delta}{\delta u_k} \log\left(\sigma(-u_{i;w_i \neq w_k}^T v_c)\right)\right]$$

Now, $\frac{\delta}{\delta u_k} \log\left(\sigma(-u_{i;w_i \neq w_k}^T v_c)\right) = 0$. Thus,

$$\frac{\delta J_{\text{neg-sample}}(v_c, o, U)}{\delta u_k} = -\left[\sum_{i \in \{1,\dots,K\}: w_i = w_k} \frac{\delta}{\delta u_k} \log\left(\sigma(-u_i^T v_c)\right)\right]$$

Or equivalently,

$$\frac{\delta J_{\text{neg-sample}}(v_c, o, U)}{\delta u_k} = -\left[\sum_{i \in \{1,\dots,K\}: w_i = w_k} \frac{\delta}{\delta u_k} \log\left(\sigma(-u_k^T v_c)\right)\right]$$

Using the chain rule on $\log$,

$$\begin{aligned}\frac{\delta J_{\text{neg-sample}}(v_c, o, U)}{\delta u_k} &= -\sum_{i\in\{1,\dots,K\}:w_i=w_k} \frac{-v_c\sigma(-u_k^T v_c)(1-\sigma(-u_k^T v_c))}{\sigma(-u_k^T v_c)} \\ &= \sum_{i\in\{1,\dots,K\}:w_i=w_k} v_c(1-\sigma(-u_k^T v_c)) \\ &= \boxed{\sum_{i\in\{1,\dots,K\}:w_i=w_k} -v_c(\sigma(-u_k^T v_c)-1)}\end{aligned}$$

(h) (3 points) Suppose the center word is $c = w_t$ and the context window is $[w_{t-m}, \dots, w_{t-1}, w_t, w_{t+1}, \dots, w_{t+m}]$, where m is the context window size. Recall that for the skip-gram version of `word2vec`, the total loss for the context window is:

$$J_{\text{skip-gram}}(v_c, w_{t-m}, \dots, w_{t+m}, U) = \sum_{\substack{-m\le j\le m \\ j\ne 0}} J(v_c, w_{t+j}, U)$$

Here, $J(v_c, w_{t+j}, U)$ represents an arbitrary loss term for the center word $c = w_t$ and outside word w_{t+j}. $J(v_c, w_{t+j}, U)$ could be $J_{\text{naive-softmax}}(v_c, w_{t+j}, U)$ or $J_{\text{neg-sample}}(v_c, w_{t+j}, U)$, depending on your implementation.

Write down three partial derivatives:

(i) $\delta J_{\text{skip-gram}}(v_c, w_{t-m}, \dots, w_{t+m}, U)/\delta U$

(ii) $\delta J_{\text{skip-gram}}(v_c, w_{t-m}, \dots, w_{t+m}, U)/\delta v_c$

(iii) $\delta J_{\text{skip-gram}}(v_c, w_{t-m}, \dots, w_{t+m}, U)/\delta v_w \text{ when } w \neq c$

Write your answers in terms of $\delta J(v_c, w_{t+j}, U)/\delta U$ and $\delta J(v_c, w_{t+j}, U)/\delta v_c$. This is very simple – each solution should be one line.

Once you're done: Given that you computed the derivatives of $J(v_c, w_{t+j}, U)$ with respect to all the model parameters U and V in parts (a) to (c), you have now computed the derivatives of the full loss function $J_{\text{skip-gram}}$ with respect to all parameters. You're ready to implement `word2vec`!

💡 **Answer:**

$$\frac{\delta J_{\text{skip-gram}}(v_c, w_{t-m}, ..., w_{t+m}, U)}{\delta U} = \sum_{\substack{-m \leq j \leq m \\ j \neq 0}} \frac{\delta J(v_c, w_{t+j}, U)}{\delta U}$$

$$\frac{\delta J_{\text{skip-gram}}(v_c, w_{t-m}, ..., w_{t+m}, U)}{\delta v_c} = \sum_{\substack{-m \leq j \leq m \\ j \neq 0}} \frac{\delta J(v_c, w_{t+j}, U)}{\delta v_c}$$

$$\frac{\delta J_{\text{skip-gram}}(v_c, w_{t-m}, ..., w_{t+m}, U)}{\delta v_w} = 0$$