1 Written: Understanding word2vec (30 points)

(a) (3 points) Prove that the naive-softmax loss (Equation 2) is the same as the cross-entropy loss between $\boldsymbol{y}$ and $\hat{\boldsymbol{y}}$, i.e. (note that $\boldsymbol{y}, \hat{\boldsymbol{y}}$ are vectors and $\hat{\boldsymbol{y}}_o$ is a scalar):

$$-\sum_{w \in \text{Vocab}} \boldsymbol{y}_w \log(\hat{\boldsymbol{y}}_w) = -\log(\hat{\boldsymbol{y}}_o). \tag{3}$$

Your answer should be one line. You may describe your answer in words.

y is a one-hot vector, and has only a 1 for y_o, rest are zeros.

$$-y \log(\hat{y}) = -\sum_w y_w \log(\hat{y}_w) = -\log(\hat{y}_o) \qquad y_w, \hat{y}_w \in R. \quad y, \hat{y} \in R^m$$

$m = |w|$ i.e. size of Vocab

(b) (5 points) Compute the partial derivative of $\boldsymbol{J}_{\text{naive-softmax}}(\boldsymbol{v}_c, o, \boldsymbol{U})$ with respect to $\boldsymbol{v}_c$. Please write your answer in terms of $\boldsymbol{y}$, $\hat{\boldsymbol{y}}$, and $\boldsymbol{U}$. Additionally, answer the following two questions with one sentence each: (1) When is the gradient zero? (2) Why does subtracting this gradient, in the general case when it is nonzero, make $\boldsymbol{v}_c$ a more desirable vector (namely, a vector closer to outside word vectors in its window)?

$v_i, u_i \in R^d$, $U \in R^{d \times m}$, where $d = \#$dimensions, $m = |w|$ size of vocab.

$$\left(\log \hat{y}_o\right)'_{v_c} = \frac{1}{\hat{y}_o} \left(\frac{\exp u_o^T v_c}{\sum_w \exp u_w^T v_c} \right)'_{v_c}$$

$$= \frac{1}{\hat{y}_o} \left(\frac{\exp u_o^T v_c}{\sum_w \exp u_w^T v_c} \cdot u_o - \frac{\exp u_o^T v_c}{\left(\sum_w \exp u_w^T v_c\right)^2} \sum_w \exp u_w^T v_c \cdot u_w \right)$$

$$= \frac{1}{\hat{y}_o} \left(\hat{y}_o u_o - \hat{y}_o \cdot U \cdot \hat{y} \right)$$

$$= u_o - U \hat{y} \quad (\in R^d)$$

$$\nabla J_{v_c} = U \hat{y} - u_o$$

(1) when $U \hat{y} = u_o$, gradient is zero

(2) Loss J_{v_c} increases the fastest in the direction of ∇J_{v_c} w.r.t. v_c.

i.e. Substracting ∇J_{v_c} reduce loss the fastest w.r.t. v_c.

(c) (5 points) Compute the partial derivatives of $\boldsymbol{J}_{\text{naive-softmax}}(\boldsymbol{v}_c, o, \boldsymbol{U})$ with respect to each of the 'outside' word vectors, $\boldsymbol{u}_w$'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 $\boldsymbol{y}$, $\hat{\boldsymbol{y}}$, and $\boldsymbol{v}_c$. In this subpart, you may use specific elements within these terms as well (such as $\boldsymbol{y}_1$, $\boldsymbol{y}_2$, ...). Note that $\boldsymbol{u}_w$ is a vector while $\boldsymbol{y}_1, \boldsymbol{y}_2, \ldots$ are scalars.

$$(\log \hat{y}_o)'_{u_o} = \frac{1}{\hat{y}_o}\left(\frac{\exp u_o^T v_c}{\sum_w \exp u_w^T v_c}\right)'_{u_o} \qquad w=o$$

$$= \frac{1}{\hat{y}_o}\left(\frac{\exp u_o^T v_c}{\sum_w \exp u_w^T v_c} \cdot v_c - \frac{\exp u_o^T v_c}{\left(\sum_w \exp u_w^T v_c\right)^2} \exp u_o^T v_c \cdot v_c\right)$$

$$= \frac{1}{\hat{y}_o}(\hat{y}_o v_c - \hat{y}_o^2 v_c)$$

$$= v_c - \hat{y}_o v_c$$

$$= (1 - \hat{y}_o) v_c.$$

$$(\log \hat{y}_o)'_{u_w} = -\frac{1}{\hat{y}_o} \frac{\exp u_o^T v_c}{\left(\sum_w \exp u_w^T v_c\right)^2} \exp u_w^T v_c \cdot v_c \qquad w \neq o$$

$$= -\frac{1}{\hat{y}_o} \cdot \hat{y}_o \cdot \hat{y}_w v_c$$

$$= -\hat{y}_w \cdot v_c$$

$$\nabla J_{u_w} = \begin{cases} (\hat{y}_o - 1) v_c & w = o \\ \hat{y}_w v_c & w \neq o. \end{cases}$$

When $\hat{y}$ approaches y, $\hat{y}_o \to 1$, $\hat{y}_w \to 0$, gradient $\to 0$

(d) (1 point) Write down the partial derivative of $\boldsymbol{J}_{\text{naive-softmax}}(\boldsymbol{v}_c, o, \boldsymbol{U})$ with respect to $\boldsymbol{U}$. Please break down your answer in terms of $\frac{\partial \boldsymbol{J}(\boldsymbol{v}_c,o,\boldsymbol{U})}{\partial \boldsymbol{u}_1}, \frac{\partial \boldsymbol{J}(\boldsymbol{v}_c,o,\boldsymbol{U})}{\partial \boldsymbol{u}_2}, \cdots, \frac{\partial \boldsymbol{J}(\boldsymbol{v}_c,o,\boldsymbol{U})}{\partial \boldsymbol{u}_{|\text{Vocab}|}}$. The solution should be one or two lines long.

Let $m = |\text{vocab}|$, $d = \#\text{dimension}$ $\quad U \in R^{d \times m}$

$$\nabla J_u = \left(\frac{\partial J}{\partial u_1} \quad \frac{\partial J}{\partial u_2} \cdots \frac{\partial J}{\partial u_m} \right) \in R^{d \times m}$$

$$\frac{\partial J}{\partial u_w} \in R^{d \times 1}$$

(e) (2 points) The ReLU (Rectified Linear Unit) activation function is given by Equation 4:

$$f(x) = \max(0, x) \tag{4}$$

Please compute the derivative of $f(x)$ with respect to x, where x is a scalar. You may ignore the case that the derivative is not defined at 0.[5]

$$\frac{df}{dx} = \begin{cases} 0 & x \le 0 \\ 1 & x > 0 \end{cases}$$

(f) (3 points) The sigmoid function is given by Equation 5:

$$\sigma(x) = \frac{1}{1 + e^{-x}} = \frac{e^x}{e^x + 1} \tag{5}$$

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 $\sigma(x)$.

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

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

(g) (6 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 $w_1, w_2, \dots, w_K$, and their outside vectors as $\boldsymbol{u}_{w_1}, \boldsymbol{u}_{w_2}, \dots, \boldsymbol{u}_{w_K}$. [6] For this question, assume that the K negative samples are distinct. In other words, $i \neq j$ implies $w_i \neq w_j$ for $i, j \in \{1, \dots, K\}$. Note that $o \notin \{w_1, \dots, w_K\}$. For a center word c and an outside word o, the negative sampling loss function is given by:

$$\boldsymbol{J}_{\text{neg-sample}}(\boldsymbol{v}_c, o, \boldsymbol{U}) = -\log(\sigma(\boldsymbol{u}_o^\top \boldsymbol{v}_c)) - \sum_{s=1}^{K} \log(\sigma(-\boldsymbol{u}_{w_s}^\top \boldsymbol{v}_c)) \tag{6}$$

for a sample $w_1, \dots w_K$, where $\sigma(\cdot)$ is the sigmoid function.[7]

(i) Please repeat parts (b) and (c), computing the partial derivatives of $\boldsymbol{J}_{\text{neg-sample}}$ with respect to $\boldsymbol{v}_c$, with respect to $\boldsymbol{u}_o$, and with respect to the s^{th} negative sample $\boldsymbol{u}_{w_s}$. Please write your answers in terms of the vectors $\boldsymbol{v}_c$, $\boldsymbol{u}_o$, and $\boldsymbol{u}_{w_s}$, where $s \in [1, K]$. **Note:** you should be able to use your solution to part (f) to help compute the necessary gradients here.

$$\nabla J_{v_c} = -\frac{1}{\sigma(u_o^T v_c)}(1-\sigma(u_o^T v_c))\,\sigma(u_o^T v_c)\,u_o - \sum_{s=1}^{K}\frac{(1-\sigma(-u_{w_s}^T v_c))\cdot\sigma(-u_{w_s}^T v_c)}{\sigma(-u_{w_s}^T v_c)}(-u_{w_s})$$

$$= -(1-\sigma(u_o^T v_c))\,u_o - \sum_{s=1}^{k}(1-\sigma(-u_{w_s}^T v_c))(-u_{w_s})$$

$$\nabla J_{u_o} = -(1-\sigma(u_o^T v_c))\,v_c$$

$$\nabla J_{u_{w_s}} = -(1-\sigma(-u_{w_s}^T v_c))(-v_c) = (1-\sigma(-u_{w_s}^T v_c))\,v_c.$$

(ii) In lecture, we learned that an efficient implementation of backpropagation leverages the re-use of previously-computed partial derivatives. Which quantity could you reuse between the three partial derivatives to minimize duplicate computation? Write your answer in terms of $\boldsymbol{U}_{o,\{w_1,\dots,w_K\}} = [\boldsymbol{u}_o, -\boldsymbol{u}_{w_1}, \dots, -\boldsymbol{u}_{w_K}]$, a matrix with the outside vectors stacked as columns, and $\mathbf{1}$, a $(K+1) \times 1$ vector of 1's. [8]

$$C_{k+1} - \sigma(U_{o,\{w_1\cdots w_K\}}^T v_c)$$

$$= \begin{bmatrix} 1-\sigma(u_o^T v_c) \\ 1-\sigma(-u_{w_1}^T v_c) \\ \vdots \\ 1-\sigma(-u_{w_K}^T v_c) \end{bmatrix}$$

$u_o \in R^d,\ v_c \in R^d$

$U_{o,\{w_1,\cdots w_K\}} \in R^{d\times(k+1)}$

$$C_{k+1} = \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} \in R^{k+1}$$

(iii) Describe with one sentence why this loss function is much more efficient to compute than the naive-softmax loss.

The naive-softmax loss requires to go through the entire vocabulary, whereas this negative sampling loss sums k randomly sampled vectors.

(h) (2 points) 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 $w_1, w_2, \dots, w_K$ and their outside vectors as $\boldsymbol{u}_{w_1}, \dots, \boldsymbol{u}_{w_K}$. 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 $o \notin \{w_1, \dots, w_K\}$. For a center word c and an outside word o, the negative sampling loss function is given by:

$$\boldsymbol{J}_{\text{neg-sample}}(\boldsymbol{v}_c, o, \boldsymbol{U}) = -\log(\sigma(\boldsymbol{u}_o^\top \boldsymbol{v}_c)) - \sum_{s=1}^{K} \log(\sigma(-\boldsymbol{u}_{w_s}^\top \boldsymbol{v}_c)) \tag{7}$$

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

Compute the partial derivative of $\boldsymbol{J}_{\text{neg-sample}}$ with respect to a negative sample $\boldsymbol{u}_{w_s}$. Please write your answers in terms of the vectors $\boldsymbol{v}_c$ and $\boldsymbol{u}_{w_s}$, where $s \in [1, K]$. Hint: break up the sum in the loss function into two sums: a sum over all sampled words equal to w_s and a sum over all sampled words not equal to w_s. Notation-wise, you may write 'equal' and 'not equal' conditions below the summation symbols, such as in Equation 8.

$$\nabla J_{u_{w_s}} = \sum_{\substack{i=1 \\ w_i = w_s}}^{k} (1 - \sigma(-u_{w_i}^\top v_c))\, v_c \,.$$

(i) (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:

$$\boldsymbol{J}_{\text{skip-gram}}(\boldsymbol{v}_c, w_{t-m}, \dots w_{t+m}, \boldsymbol{U}) = \sum_{\substack{-m \le j \le m \\ j \neq 0}} \boldsymbol{J}(\boldsymbol{v}_c, w_{t+j}, \boldsymbol{U}) \tag{8}$$

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

Write down three partial derivatives:

(i) $\frac{\partial \boldsymbol{J}_{\text{skip-gram}}(\boldsymbol{v}_c, w_{t-m}, \dots w_{t+m}, \boldsymbol{U})}{\partial \boldsymbol{U}}$

(ii) $\frac{\partial \boldsymbol{J}_{\text{skip-gram}}(\boldsymbol{v}_c, w_{t-m}, \dots w_{t+m}, \boldsymbol{U})}{\partial \boldsymbol{v}_c}$

(iii) $\frac{\partial \boldsymbol{J}_{\text{skip-gram}}(\boldsymbol{v}_c, w_{t-m}, \dots w_{t+m}, \boldsymbol{U})}{\partial \boldsymbol{v}_w}$ when $w \neq c$

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

(i) $$\frac{\partial J_{\text{skip-gram}}(v_c, w_{t-m}, \cdots w_{t+m}, U)}{\partial U} = \sum_{\substack{-m \le j \le m \\ j \neq 0}} \frac{\partial J(v_c, w_{t+j}, U)}{\partial U}$$

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

(iii) $$\frac{\partial J_{\text{skip-gram}}(v_c, w_{t-m}, \cdots w_{t+m}, U)}{\partial v_w} = 0 \quad \text{when } w \neq c$$