CS 224n Assignment 2: word2vec

Lukashevich Ilya

November 28, 2020

1 Written part: Understanding word2vec

Let's write some formulas, which were introduced in the recap section of *word2vec* algorithm, that will be used in solutions. The goal of the skip-gram *word2vec* algorithm is to accurately learn the probability distribution $P(O \mid C)$. The formula for the probability that the word o falls within the contextual window of c is:

$$P(O = o \mid C = c) = \frac{\exp\left(\boldsymbol{u}_o^T \boldsymbol{v}_c\right)}{\sum_{w \in Vocab} \exp\left(\boldsymbol{u}_w^T \boldsymbol{v}_c\right)} \tag{1}$$

For a single pair of words c and o the loss in given by:

$$\boldsymbol{J}_{\text{naive-softmax}}(\boldsymbol{v}_c,\ o,\ \boldsymbol{U}) = -\log P(O = o \mid C = c) \tag{2}$$

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

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

Solution. We know that $\boldsymbol{y}$ shows the true empirical distribution and it is a one-hot vector with 1 for the true outside word, and 0 everywhere else. So, the sum contains only one non-zero item:

$$-\sum_{w \in Vocab} y_w \log(\hat{y}_w) = -(0 + \ldots 0 + 1 \cdot \log(\hat{y}_o) + 0 + \ldots + 0) = -\log(\hat{y}_o)$$

(b) 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 $\boldsymbol{U}$.

Solution.

$$\begin{aligned}
\frac{\partial \boldsymbol{J}_{\text{naive-softmax}}}{\partial \boldsymbol{v}_c} &= -\frac{\partial}{\partial \boldsymbol{v}_c} \log \frac{\exp\left(\boldsymbol{u}_o^T \boldsymbol{v}_c\right)}{\sum_{w \in \text{Vocab}} \exp\left(\boldsymbol{u}_w^T \boldsymbol{v}_c\right)} = -\frac{\partial}{\partial \boldsymbol{v}_c} \log \frac{\exp\left(\boldsymbol{u}_o^T \boldsymbol{v}_c\right)}{\sum_{w \in \text{Vocab}} \exp\left(\boldsymbol{u}_w^T \boldsymbol{v}_c\right)} \\
&= -\frac{\partial}{\partial \boldsymbol{v}_c} \log \exp\left(\boldsymbol{u}_o^T \boldsymbol{v}_c\right) + \frac{\partial}{\partial \boldsymbol{v}_c} \log \sum_{w \in \text{Vocab}} \exp\left(\boldsymbol{u}_w^T \boldsymbol{v}_c\right) \\
&= -\boldsymbol{u}_o + \frac{1}{\sum_{w \in \text{Vocab}} \exp\left(\boldsymbol{u}_w^T \boldsymbol{v}_c\right)} \frac{\partial}{\partial \boldsymbol{v}_c} \sum_{x \in \text{Vocab}} \exp\left(\boldsymbol{u}_x^T \boldsymbol{v}_c\right) \\
&= -\boldsymbol{u}_o + \frac{1}{\sum_{w \in \text{Vocab}} \exp\left(\boldsymbol{u}_w^T \boldsymbol{v}_c\right)} \sum_{x \in \text{Vocab}} \exp\left(\boldsymbol{u}_x^T \boldsymbol{v}_c\right) \frac{\partial}{\partial \boldsymbol{v}_c} \boldsymbol{u}_x^T \boldsymbol{v}_c \\
&= -\boldsymbol{u}_o + \frac{1}{\sum_{w \in \text{Vocab}} \exp\left(\boldsymbol{u}_w^T \boldsymbol{v}_c\right)} \sum_{x \in \text{Vocab}} \exp\left(\boldsymbol{u}_x^T \boldsymbol{v}_c\right) \boldsymbol{u}_x \\
&= -\boldsymbol{u}_o + \sum_{x \in \text{Vocab}} \frac{\exp\left(\boldsymbol{u}_x^T \boldsymbol{v}_c\right)}{\sum_{w \in \text{Vocab}} \exp\left(\boldsymbol{u}_w^T \boldsymbol{v}_c\right)} \boldsymbol{u}_x \\
&= -\boldsymbol{u}_o + \sum_{x \in \text{Vocab}} P(O = x \mid C = c) \boldsymbol{u}_x = -\boldsymbol{U}\boldsymbol{y} + \boldsymbol{U}\hat{\boldsymbol{y}} = \boldsymbol{U}(\hat{\boldsymbol{y}} - \boldsymbol{y})
\end{aligned}$$

(c) 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' vector, and $w \neq o$, for all other words. Please write down your answer in terms of $\boldsymbol{y}$, $\hat{\boldsymbol{y}}$, and $\boldsymbol{v}_c$.

Solution.

$$\begin{aligned}\frac{\partial \boldsymbol{J}_{\text{naive-softmax}}}{\partial \boldsymbol{u}_w} &= -\frac{\partial}{\partial \boldsymbol{u}_w} \log \frac{\exp\left(\boldsymbol{u}_o^T \boldsymbol{v}_c\right)}{\sum_{x \in \text{Vocab}} \exp\left(\boldsymbol{u}_x^T \boldsymbol{v}_c\right)} \\ &= -\frac{\partial}{\partial \boldsymbol{u}_w} \log \exp\left(\boldsymbol{u}_o^T \boldsymbol{v}_c\right) + \frac{\partial}{\partial \boldsymbol{u}_w} \log \sum_{x \in \text{Vocab}} \exp\left(\boldsymbol{u}_x^T \boldsymbol{v}_c\right)\end{aligned}$$

Now let's consider two different cases:

1. $w = o$:

$$\begin{aligned}\frac{\partial \boldsymbol{J}_{\text{naive-softmax}}}{\partial \boldsymbol{u}_o} &= -\boldsymbol{v}_c + \frac{1}{\sum_{x \in \text{Vocab}} \exp\left(\boldsymbol{u}_x^T\right)} \sum_{z \in \text{Vocab}} \frac{\partial}{\partial \boldsymbol{u}_o} \exp\left(\boldsymbol{u}_z^T \boldsymbol{v}_c\right) \\ &= -\boldsymbol{v}_c + \frac{1}{\sum_{x \in \text{Vocab}} \exp\left(\boldsymbol{u}_x^T\right)} \frac{\partial}{\partial \boldsymbol{u}_o} \exp\left(\boldsymbol{u}_o^T \boldsymbol{v}_c\right) \\ &= -\boldsymbol{v}_c + \frac{\exp\left(\boldsymbol{u}_o^T \boldsymbol{v}_c\right)}{\sum_{x \in \text{Vocab}} \exp\left(\boldsymbol{u}_x^T\right)} \boldsymbol{v}_c \\ &= -\boldsymbol{v}_c + P(O = o \mid C = c)\boldsymbol{v}_c \\ &= (P(O = w = o \mid C = c) - 1)\boldsymbol{v}_c\end{aligned}$$

2. $w \neq o$:

$$\begin{aligned}\frac{\partial \boldsymbol{J}_{\text{naive-softmax}}}{\partial \boldsymbol{u}_w} &= \frac{\partial}{\partial \boldsymbol{u}_w} \log \sum_{x \in \text{Vocab}} \exp\left(\boldsymbol{u}_x^T \boldsymbol{v}_c\right) \\ &= \frac{\exp\left(\boldsymbol{u}_w^T \boldsymbol{v}_c\right)}{\sum_{x \in \text{Vocab}} \exp\left(\boldsymbol{u}_x^T\right)} \boldsymbol{v}_c \\ &= P(O = w \mid C = c)\boldsymbol{v}_c \\ &= (P(O = w \mid C = c) - 0)\boldsymbol{v}_c\end{aligned}$$

To conclude, we have

$$\frac{\partial \boldsymbol{J}_{\text{naive-softmax}}}{\partial \boldsymbol{u}_w} = (\hat{y}_w - y_w)\boldsymbol{v}_c.$$

(d) The sigmoid function is given by Equation (4):

$$\sigma(x) = \frac{1}{1 + e^{-x}} = \frac{e^x}{e^x + 1} \tag{4}$$

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 $\sigma(x)$.

Solution.

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

(e) 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 $w_1, w_2, \ldots, w_K$ and their outside vectors as $\boldsymbol{u}_1, \ldots, \boldsymbol{u}_K$. Note that $o \notin \{w_1, \ldots, w_K\}$. For a center word c and an outside word o, the negative sampling loss function is given by:

$$\boldsymbol{J}_{\text{neg-sample}}(\boldsymbol{v}_c,\ o,\ \boldsymbol{U}) = -\log\left(\sigma(\boldsymbol{u}_o^T\boldsymbol{v}_c)\right) - \sum_{k=1}^{K}\log\left(\sigma(\boldsymbol{u}_k^T\boldsymbol{v}_c)\right) \tag{5}$$

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

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 a negative sample $\boldsymbol{u}_k$. Please write your answers in terms of the vectors $\boldsymbol{u}_o$, $\boldsymbol{v}_c$, and $\boldsymbol{u}_k$, 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 (d) to help compute the necessary gradients here.

Solution.

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

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

$$
\begin{aligned}
\frac{\partial \boldsymbol{J}_{\text{neg-sample}}}{\partial \boldsymbol{u}_k} &= -\frac{\partial}{\partial \boldsymbol{u}_k} \log\left(\sigma\left(\boldsymbol{u}_o^T \boldsymbol{v}_c\right)\right) - \frac{\partial}{\partial \boldsymbol{u}_k} \sum_{x=1}^{K} \log\left(\sigma\left(-\boldsymbol{u}_x^T \boldsymbol{v}_c\right)\right) \\
&= -\frac{\partial}{\partial \boldsymbol{u}_k} \sum_{x=1}^{K} \log\left(\sigma\left(-\boldsymbol{u}_x^T \boldsymbol{v}_c\right)\right) \\
&= -\frac{\partial}{\partial \boldsymbol{u}_k} \log\left(\sigma\left(-\boldsymbol{u}_k^T \boldsymbol{v}_c\right)\right) \\
&= -\frac{1}{\sigma\left(-\boldsymbol{u}_k^T \boldsymbol{v}_c\right)} \frac{\partial}{\partial \boldsymbol{u}_k} \sigma\left(-\boldsymbol{u}_k^T \boldsymbol{v}_c\right) \\
&= -\frac{1}{\sigma\left(-\boldsymbol{u}_k^T \boldsymbol{v}_c\right)} \sigma\left(-\boldsymbol{u}_k^T \boldsymbol{v}_c\right)\left(1 - \sigma\left(-\boldsymbol{u}_k^T \boldsymbol{v}_c\right)\right) \frac{\partial}{\partial \boldsymbol{u}_k} \left(-\boldsymbol{u}_k^T \boldsymbol{v}_c\right) \\
&= \left(1 - \sigma\left(-\boldsymbol{u}_k^T \boldsymbol{v}_c\right)\right) \boldsymbol{v}_c
\end{aligned}
$$

This loss function is much more efficient to compute than the naive-softmax loss because at each step we do not go through all the words in the vocabulary, which is quite expensive in terms on speed costs.

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

$$
\boldsymbol{J}_{\text{skip-gram}}(\boldsymbol{v}_c, w_{t-m}, \ldots, w_{t+m}, \boldsymbol{U}) = \sum_{\substack{-m \le j \le m \\ j \neq 0}} \boldsymbol{J}(\boldsymbol{v}_c, w_{t+j}, \boldsymbol{U}) \tag{6}
$$

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:

1. $\partial \boldsymbol{J}_{\text{skip-gram}}(\boldsymbol{v}_c, w_{t-m}, \ldots, w_{t+m}, \boldsymbol{U}) / \partial \boldsymbol{U}$
2. $\partial \boldsymbol{J}_{\text{skip-gram}}(\boldsymbol{v}_c, w_{t-m}, \ldots, w_{t+m}, \boldsymbol{U}) / \partial \boldsymbol{v}_c$
3. $\partial \boldsymbol{J}_{\text{skip-gram}}(\boldsymbol{v}_c, w_{t-m}, \ldots, w_{t+m}, \boldsymbol{U}) / \partial \boldsymbol{v}_w$ when $w \neq c$

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

Solution.

1. $$\partial \boldsymbol{J}_{\text{skip-gram}}(\boldsymbol{v}_c, w_{t-m}, \ldots, w_{t+m}, \boldsymbol{U})/\partial \boldsymbol{U} = \sum_{\substack{-m \le j \le m \\ j \neq 0}} \partial \boldsymbol{J}(\boldsymbol{v}_c, w_{t+j}, \boldsymbol{U})/\partial \boldsymbol{U}$$

2. $$\partial \boldsymbol{J}_{\text{skip-gram}}(\boldsymbol{v}_c, w_{t-m}, \ldots, w_{t+m}, \boldsymbol{U})/\partial \boldsymbol{v}_c = \sum_{\substack{-m \le j \le m \\ j \neq 0}} \partial \boldsymbol{J}(\boldsymbol{v}_c, w_{t+j}, \boldsymbol{U})/\partial \boldsymbol{v}_c$$

3. $$\partial \boldsymbol{J}_{\text{skip-gram}}(\boldsymbol{v}_c, w_{t-m}, \ldots, w_{t+m}, \boldsymbol{U})/\partial \boldsymbol{v}_w = 0 \text{ when } w \neq c$$