1. Understanding word2vec

(a) Since word o is the expected word, the true distribution y is a one-hot vector with the value 1 for word o and 0 otherwise.

Therefore $-\sum_{w \in V} y_w \log(\hat{y}_w) = -y_o \log(\hat{y}_o) = -\log(\hat{y}_o)$

(b) Compute $\partial J_{naive\text{-}softmax}(v_c, o, U) / \partial v_c$, write answer in terms of y, $\hat{y}$, and U

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

$$= -\log p(o|c)$$

$$= -\log \frac{\exp(u_o^T v_c)}{\sum_w \exp(u_w^T v_c)}$$

Then, $$\frac{\partial J}{\partial v_c} = -\frac{\partial}{\partial v_c} \log \frac{\exp(u_o^T v_c)}{\sum_w \exp(u_w^T v_c)}$$

$$= -\left[\frac{\partial (u_w^T v_c)}{\partial v_c} - \frac{\partial \log \sum_w \exp(u_w^T v_c)}{\partial v_c} \right]$$

$$= -\left[u_o - \frac{1}{\sum_w \exp(u_w^T v_c)} \frac{\partial \sum_x \exp(u_x^T v_c)}{\partial v_c} \right]$$

$$= -\left[u_o - \frac{\sum_x \exp(u_x^T v_c) u_x}{\sum_w \exp(u_w^T v_c)} \right]$$

$$= -\left[u_o - \sum_x \frac{\exp(u_x^T v_c)}{\sum_w \exp(u_w^T v_c)} u_x \right]$$

$$= -\left[u_o - \sum_x p(x|c) u_x \right]$$

$$= -\left[u_o - \sum_{x=1}^{\checkmark} \hat{y}_x u_x \right] = U(\hat{y} - y)$$

(c) compute $\partial J_{naive\text{-}softmax}(v_c, o, U) / \partial u_w$, write answer interms of y, $\hat{y}$, and v_c

when $w = o$

$$\frac{\partial J}{\partial u_o} = -\frac{\partial}{\partial u_o} \log \frac{\exp(u_o^T v_c)}{\sum_w \exp(u_w^T v_c)}$$

$$= -\left[\frac{\partial (u_o^T v_c)}{\partial u_o} - \frac{\partial \log \sum_w \exp(u_w^T v_c)}{\partial u_o} \right]$$

$$= -\left(v_c - \frac{1}{\sum_w \exp(u_w^T v_c)} \frac{\partial \exp(u_o^T v_c)}{\partial u_o} \right)$$

$$= -\left(v_c - \frac{\exp(u_o^T v_c) \cdot v_c}{\sum_w \exp(u_w^T v_c)} \right)$$

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

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

when $w \neq o$

$$\frac{\partial J}{\partial u_w} = -\frac{\partial}{\partial u_w} \log \frac{\exp(u_o^T v_c)}{\sum_x \exp(u_x^T v_c)}$$

$$= -\left(0 - \frac{\partial \log \sum_x \exp(u_x^T v_c)}{\partial u_w} \right)$$

$$= \frac{1}{\sum_x \exp(u_x^T v_c)} \frac{\partial \sum_x \exp(u_x^T v_c)}{\partial u_w}$$

$$= \frac{\exp(u_w^T v_c) v_c}{\sum_x \exp(u_x^T v_c)} = \hat{y}_w \cdot v_c .$$

Combining above

$$\frac{\partial J}{\partial u_w} = \begin{cases} (\hat{y}_w - 1) v_c & w = o \\ \hat{y}_w \cdot v_c & w \neq o \end{cases}$$

$$= v_c (\hat{y} - y)^T$$

(d) Compute $\frac{\partial \sigma(x)}{\partial x}$ where $\sigma(x)$ is the sigmoid function.

$$\frac{\partial \sigma(x)}{\partial x} = \frac{\partial \left(\frac{1}{1+e^{-x}}\right)}{\partial x} = -\frac{e^{-x}(-1)}{(1+e^{-x})^2} = \frac{1+e^{-x}-1}{1+e^{-x}} = \sigma(x)(1-\sigma(x))$$

(e) compute $\partial J_{neg\text{-}sampling} / \partial v_c$, $\partial J / \partial u_o$, $\partial J / \partial u_k$

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

$$= -\frac{1}{\sigma(u_o^T v_c)} \frac{\partial \sigma(u_o^T v_c)}{\partial v_c} - \sum_k \frac{1}{\sigma(-u_k^T v_c)} \frac{\partial \sigma(-u_o^T v_c)}{\partial v_c}$$

$$= -\frac{\sigma(u_o^T v_c)(1-\sigma(u_o^T v_c))\, u_o}{\sigma(u_o^T v_c)} - \sum_k \frac{\sigma(-u_k^T v_c)(1-\sigma(u_k^T v_c))(-u_k)}{\sigma(-u_k^T v_c)}$$

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

Second $\frac{\partial J}{\partial u_o} = \frac{\partial \left[-\log \sigma(u_o^T v_c)\right]}{\partial u_o}$ note that $u_k \{k=1 \dots K\}$ do not contain u_o

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

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

Third. $\frac{\partial J}{\partial u_k} = \frac{\partial \left[-\sum_{x=1}^{K} \log(\sigma(-u_x^T v_c))\right]}{\partial u_k}$

$$= -\frac{\partial \log(\sigma(-u_k^T v_c))}{\partial u_k} = -\frac{\sigma(-u_k^T v_c)(1-\sigma(-u_k^T v_c))}{\sigma(-u_k^T v_c)} \cdot (-v_c)$$

$$= -(\sigma(-u_k^T v_c) - 1)\, v_c \quad \text{for all } k = 1 \dots K$$

This is more efficient to compute than naive-softmax loss, which need to compute $\hat{y}$, which is normalized by $\sum_{w \in V} \exp(u_w^T v_c)$, a sum across the whole vocabulary.

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

$$= \partial \sum_{\substack{-m \le j \le m \\ j \neq 0}} J(v_c, w_{t+j}, U) / \partial U$$

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

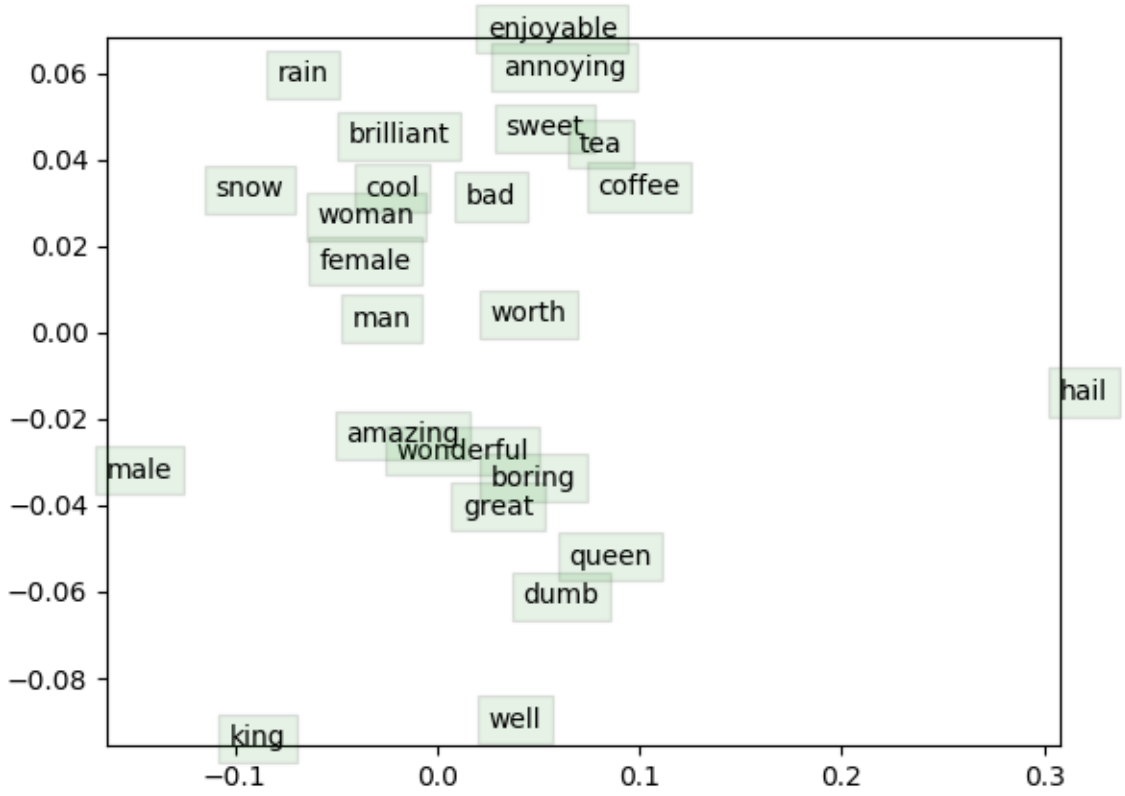
(ii) $\partial J_{\text{skip-gram}}(v_c, w_{t-m}, \ldots w_{t+m}, U) / \partial v_c$

$$= \partial \sum_{\substack{-m \le j \le m \\ j \neq 0}} J(v_c, w_{t+j}, U) / \partial v_c.$$

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

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

$= 0$



The above figure shows the output of training the word vectors. We can see analogies from the figure, i.e. the linearity between (queen: king) :: (female: male). We can also see synonyms that tend to have similar contexts cluster together, e.g. (“amazing,” “wonderful,” “great”) and (“coffee,” “tea”). The 2D visualization is not showing everything though—some word vectors may be close to each other in the higher dimensional space, but is not captured here.