a) we know that y is a one-hot vector where $y_w = 1$ when $w = o$ & $y_w = 0$ when $w \neq o$ So:

$$-\sum_{w \in Vocab} y_w \log(\hat{y}_w) = -[y_1 \log(\hat{y}_1) + \cdots + y_o \log(\hat{y}_o) + y_w \log(\hat{y}_w)] = -[0 + \cdots + y_o \log(\hat{y}_o) + \cdots + 0]$$

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

b) $$\frac{\partial J_{naive\text{-}softmax}}{\partial v_c} = \frac{\partial(-\log P(O|C))}{\partial v_c}$$

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

chain rule ↑

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

$$-u_o + \sum_{x=1}^{V} \frac{\exp(u_x^T v_c)}{\sum_{w=1}^{V} \exp(u_w^T v_c)} = (\hat{y} - y) U$$

c) $$\frac{\partial J_{naive\text{-}softmax}}{\partial v_w} = \frac{\partial}{\partial v_w} \frac{-\exp(u_o^T v_c)}{\sum_w \exp(u_w^T v_c)}$$

$$= + \frac{\partial}{\partial v_w} \left(-\log \exp(u_o^T v_c) + \underbrace{\log \sum_w \exp(u_w^T v_c)}_{②} \right)$$

if $w = o$: $\frac{\partial}{\partial u_o} (-u_o^T v_c) = -v_c$

②: $\frac{\exp(u_w^T v_c)}{\sum_w \exp(u_w^T v_c)} v_c = p(o|c)\, v_c$

$\Rightarrow v_c (p(o|c) - 1)$

if $w \neq o$: $\frac{\partial}{\partial u_w} (-u_o^T v_c) = 0$

②: $\frac{\exp(u_w^T v_c)}{\sum_w \exp(u_w^T v_c)} v_c = p(o|c)\, v_c$

$\Rightarrow p(o|c)\, v_c$

d) $$\frac{\partial J_{naive\text{-}softmax}}{u} = \left[\frac{\partial J(v_c, o, u)}{\partial u_1}, \frac{\partial J(v_c, o, u)}{\partial u_2}, \ldots, \frac{\partial J(v_c, o, u)}{\partial v_{vocab}} \right]$$

a matrix with each column has derivative

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

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

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

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

f)

① $$\frac{\partial J}{\partial v_c} = \frac{\partial}{\partial v_c}\left(-\log(\sigma(u_o^T v_c)) - \sum_K \log(\sigma(-u_K^T v_c))\right) =$$

$$-(1-\sigma(u_o^T v_c))u_o + \sum_K (1-\sigma(-u_K^T v_c))u_K$$

② $$\frac{\partial J}{\partial u_o} = -(1-\sigma(u_o^T v_c))v_c$$

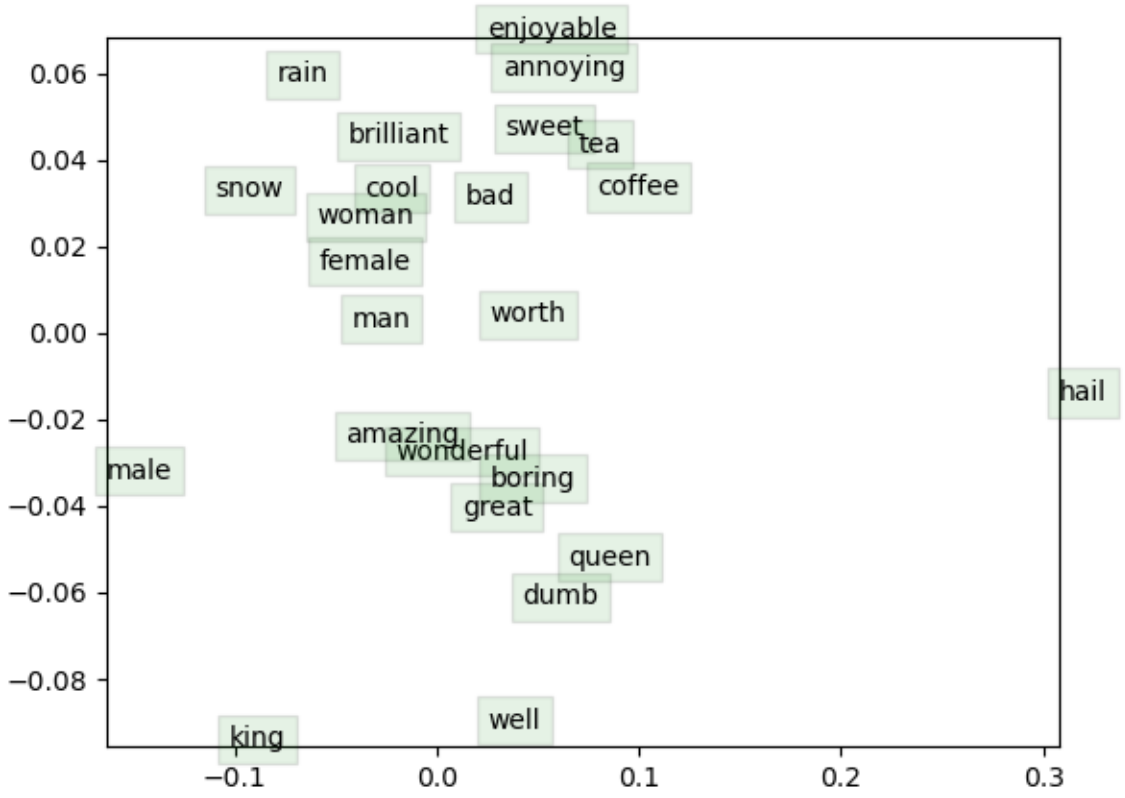
③ $$\frac{\partial J}{\partial u_K} = (1-\sigma(-u_K^T v_c))v_c$$

in naive softmax function the whole U is computed but in negative sampling loss, only k of U is computed. So it is more efficient.

h) ① $\sum_{\substack{-m<j\leq m \\ j\neq 0}} \frac{\partial J(V_c, \omega t + j, u)}{\partial u}$

② $\sum_{\substack{-m<j\leq m \\ j\neq 0}} \frac{\partial J(V_c, \omega t + j, u)}{\partial V_c}$

③ $= 0$


enjoyable
rain
annoying
brilliant
sweet
tea
snow
cool
bad
coffee
woman
female
man
worth
hail
amazing
wonderful
male
boring
great
queen
dumb
well
king
0.06
0.04
0.02
0.00
−0.02
−0.04
−0.06
−0.08
−0.1
0.0
0.1
0.2
0.3