Subject: written assignments Date

a)

which is y = $[0, 0, \ldots, \underset{i}{1}, \ldots, 0, 0]$ ← true label

y_0

which is $\hat{y}$ = $[P_0, P_1, \ldots, P_0, \ldots, 0, 0]$ ← predicted label

$-\sum_{w \in vocab} y_w \log(\hat{y}_w)$ => because all elements of y is 0 except element at position 0 so we have just = $1 \times P_0$

$P_0 = \hat{y}_0$ => $-\sum_{w \in vocab} y_w \log(\hat{y}_w) = P_0 = \hat{y}_0$

b)

i) $\frac{\partial J_{naive\text{-}softmax}(v_c, o, U)}{\partial v_c} = \frac{\partial}{\partial v_c}\left(-\log\left(\frac{\exp(u_o^T v_c)}{\sum_{w \in vocab} \exp(u_w^T v_c)}\right)\right)$

split log, division to minus → $\frac{\partial}{\partial v_c}\left(-\left(\log(\exp(u_o^T v_c)) - \log\left(\sum_{w \in vocab} \exp(u_w^T v_c)\right)\right)\right)$

$\log e^u = u$ → $\frac{\partial}{\partial v_c}\left(-\left(\exp(u_o^T v_c) - \log\left(\sum_{w \in vocab} \exp(u_w^T v_c)\right)\right)\right)$

apply derivation → $\frac{\sum \exp(u_w^T v_c)\, u_w}{\sum_{w \in vocab} \exp(u_w^T v_c)} - u_o$ $\xrightarrow{\text{softmax to } \hat{y}}$ $\sum_{w \in vocab} \hat{y} u_w - u_o$

→ $U\hat{y} - u_o$ $\xrightarrow{u_o = Uy}$ $U\hat{y} - Uy \rightarrow U(\hat{y} - y)$

because y just 0's

MICRO index is 1

Subject : Date

i) it's zero when $\begin{cases} U = 0 \ (1) \\ \text{or} \\ y = \hat{y} \ (2) \end{cases}$

(1) $U = 0$ → it's not possible because vectors should vary from each other and they can't be equal to zero.

(2) $y = \hat{y}$ → when $\hat{y}$ which we predicted is completely equal to the true y. it means that all of other words probability is equal to zero and only our target U_o has probability 1. which obviously in this case we don't need to change our vectors because we're in the place that we want to be. but in real cases is not possible this happens.

ii)

because by subtracting this gradient from our v_c we moves our v_c closer to local or global minimum. how this happens relies on the concept of loss function which we want to minimize. and gradient help us in this way. in fact gradient shows how should probability of our prediction change. by subtracting $\hat{y}$, y we want to be closer to y by changing our $\hat{y}$ in each iteration.

iii) when downstream applications only pay attention to cosine similarity, normalizing would be a good idea. but if they want to capture features beyond similarities such as significance, consistency normalizing will ignore these. so in this case it's not applicable. in fact normalizing will lose length of vectors. in some cases both length and direction carries important information

MICRO

Subject : Date

c) * when $w = 0 \rightarrow -\frac{\delta}{\delta u_w} \log \frac{\exp(u_o^T v_c)}{\sum_{w \in Vocab} \exp(u_w^T v_c)} = -\frac{\delta}{\delta u_w}(u_w^T v_c)$

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

$= -v_c + \frac{v_c \exp(u_w^T \cdot v_c)}{\sum_{w \in Vocab} \exp(u_w^T \cdot v_c)} = -v_c + v_c \hat{y}$

$\hat{y} = softmax \leftarrow$

* when $w \neq 0 \Rightarrow \underbrace{-\frac{\delta}{\delta u_w}(u_0^T v_c)}_{\text{this term is zero}} + \frac{\delta}{\delta u_w} \log \sum \exp(u_w^T v_c)$

so the only term remaing is the secand part which we calculated above

$v_c \cdot softmax(w) = v_c \hat{y}$

$\frac{\delta J_{naive\,softmax}}{\delta u_w}$ $\begin{cases} w = 0 \rightarrow -v_c + v_c \hat{y} \\ w \neq 0 \rightarrow v_c \hat{y} \end{cases}$

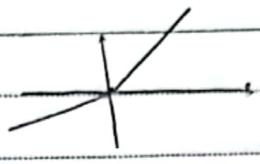
d) $\frac{\delta J_{naive\,softmax}(v_c, o, U)}{\delta U} =$

$\left[\frac{\delta J(v_c, o, U)}{\delta u_1} \quad \frac{\delta J(v_c, o, U)}{\delta u} \quad \cdots \quad \frac{\delta J(v_c, o, U)}{\delta u_{|Vocab|}} \right]$

$\left[\hat{y}_1 v_c \quad , \quad \hat{y}_2 v_c \quad \cdots \quad \hat{y}_{|Vocab|} v_c \right]$

Subject : Date

e) $f(x) = \max(ax, x)$



$$\frac{d f(x)}{dx} = \begin{cases} 1 & x > 0 \\ \alpha & x < 0 \end{cases}$$

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

$$= \sigma(x)(1 - \sigma(x))$$

g) $$J_{neg\text{-}sample} = -\log(\sigma(u_o^T v_c)) - \sum_{s=1}^{k} \log(\sigma(-u_{w_s}^T v_c))$$

i) $$\frac{\delta J_{negsampling}}{\delta v_c} = \frac{\sigma(u_o^T v_c)(1 - \sigma(u_o^T v_c))}{\sigma(u_o^T v_c)} \cdot \frac{\partial u_o^T v_c}{\partial v_c}$$

$$\sum_{k=1}^{k} \frac{\partial \log(\sigma(-u_k^T v_c))}{\partial v_c} = \sigma(u_o^T v_c - 1) u_o + \sum_{k=1}^{k} (1 - \sigma(-u_k^T v_c)) v_c$$

$$\frac{\delta J_{negsampling}}{\delta u_o} = \frac{-\delta}{\partial u_o}\left(\log \sigma(u_o^T v_c) - \sum_{k=1}^{k} \log \sigma(-u_k^T v_c)\right)$$

$$= \frac{-\delta}{\delta u_o}\left(-\frac{\log(\sigma(u_o^T v_c))}{\partial u_o} - (\sigma(u_o^T v_c) - 1) v_c\right) \quad \text{This term is 0}$$

$$\frac{\delta J_{negsampling}}{\delta u_{ws}} = \frac{\partial(-\log \sigma(-u_{ws}^T v_c))}{\delta u_{ws}} = (1 - \sigma(-u_{ws}^T v_c)) v_c$$

ii) terms which has sigmoid and inner calculations are removed

This term is repeated → $(\sigma(-u_o^T v_c) - 1)$

So we can $U_{o,\{w_1,\ldots,w_k\}}$ $v_c = \begin{bmatrix} u_o v_c \\ -u_{w_1} v_c \\ \vdots \\ u_{w_k} v_c \end{bmatrix}$ → $\sigma(U v_c) - 1 = \begin{bmatrix} \sigma(u_o v_c) - 1 \\ \sigma(u_{w_1} v_c) - 1 \\ \vdots \\ \sigma(-u_{w_k} v_c) - 1 \end{bmatrix}$

MICRO

Subject : Date

ii) in naive softmax loss we obviously iterate through whole vocabulary each time which cost a bit more then here, because we are only sampling k item in negative sampling.

h) $J_{\text{neg sampling}} = -\left(\log\left(\sigma\left(u_o^T v_c\right)\right) + \sum_{\substack{1 \le j \le k \\ w_j = w_s}} \log \sigma\left(-u_{w_j}^T v_c\right)\right)$

$$+ \sum_{\substack{1 \le j \le k \\ w_j \neq w_s}} \log\left(\sigma\left(-u_{w_j} v_c\right)\right)$$

because we want to calculate derivate from w_s so the term which doesn't have w_s will be removed. so first and third term are removed we just need to calculate derivative of second term.

$$\frac{\delta J_{\text{neg sampling}}}{\delta w_s} = -\sum_{\substack{1 \le j \le k \\ w_j = w_s}} \left(\sigma\left(-u_{w_s}^T v_c\right) - 1\right) v_c$$

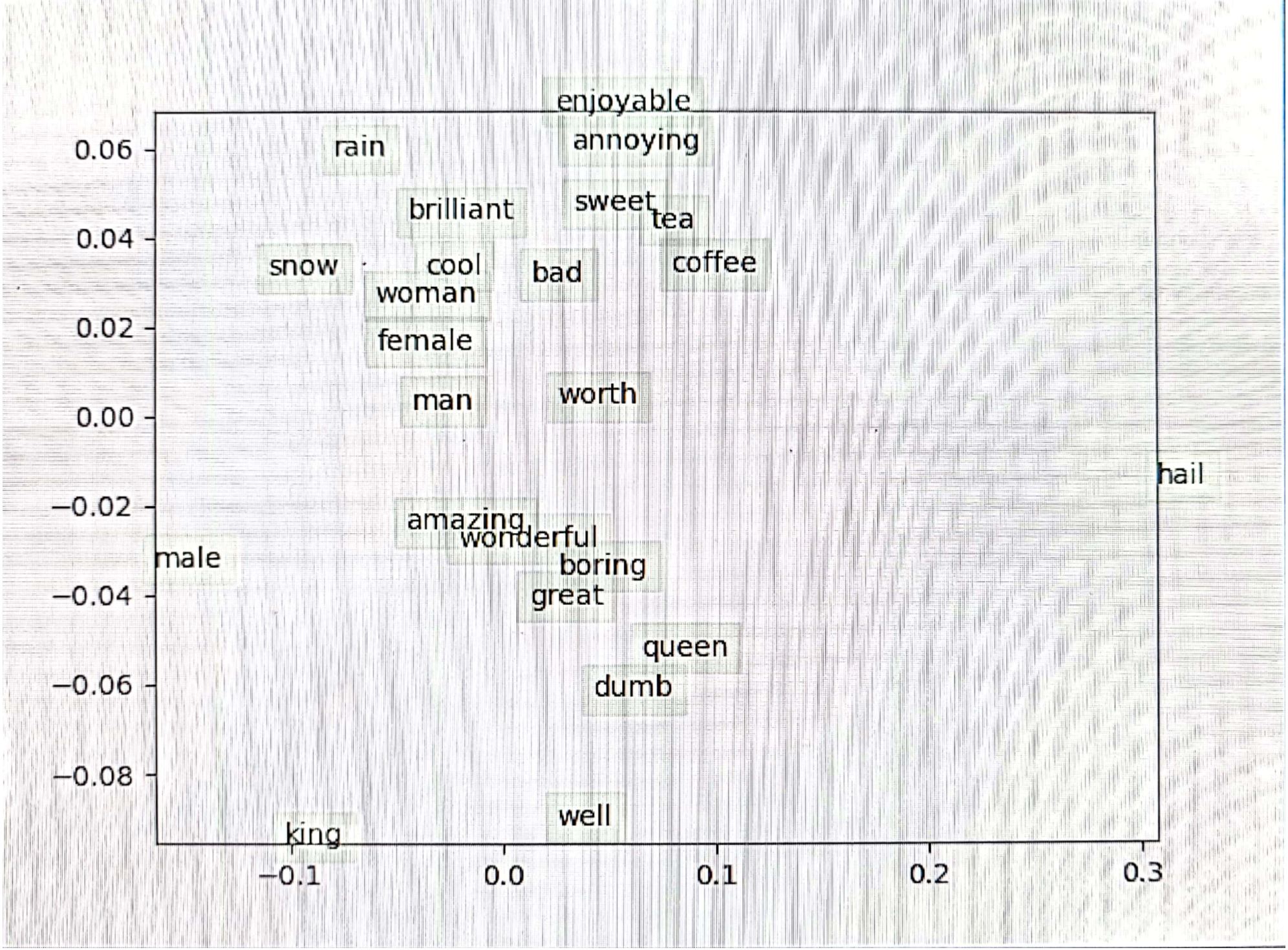
---

i

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

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

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


enjoyable
annoying
rain
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