Lecture01_Word2Vec

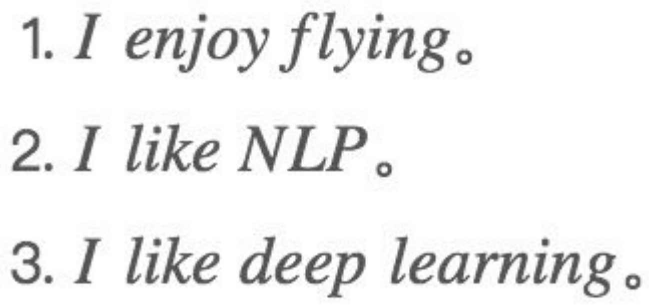
AIA 705 Felix

- 先看一下其它的Word Encoding, 以Co-occurence matrix為例
- word2vec 是什麼？ 要解決的問題又是什麼?
- word2vec的兩種機率模型 (skip-gram & continuous bag-of-words)
- word2vec的兩種訓練方法 (negative sampling & hierarchical softmax)
- Sampel Code

Co-occurrence matrix 在NLP中的限制

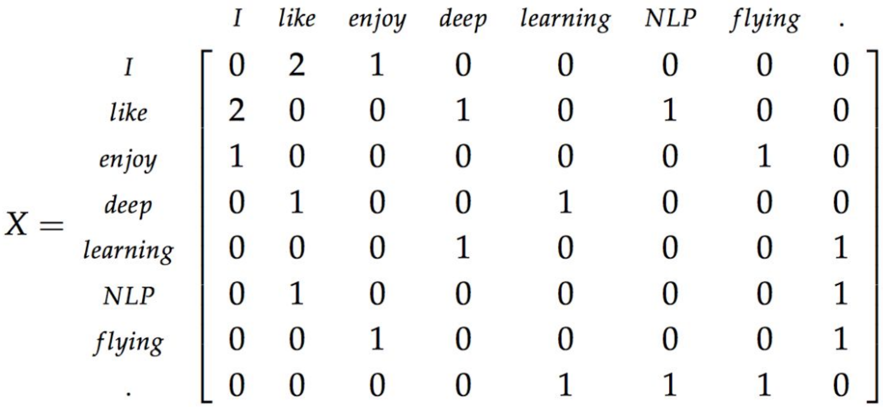
1、向量維度過高

2、資料過度稀疏，記憶體佔用大，難以實作



1. *I enjoy flying*。
2. *I like NLP*。
3. *I like deep learning*。

Sample code



$$X = \begin{array}{c|cccccccc} & I & like & enjoy & deep & learning & NLP & flying & . \\ \hline I & 0 & 2 & 1 & 0 & 0 & 0 & 0 & 0 \\ like & 2 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ enjoy & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ deep & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ learning & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\ NLP & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \\ flying & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \\ . & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 \end{array}$$

word2vec 是什麼？ 要解決什麼問題

■ word2vec 是什麼？

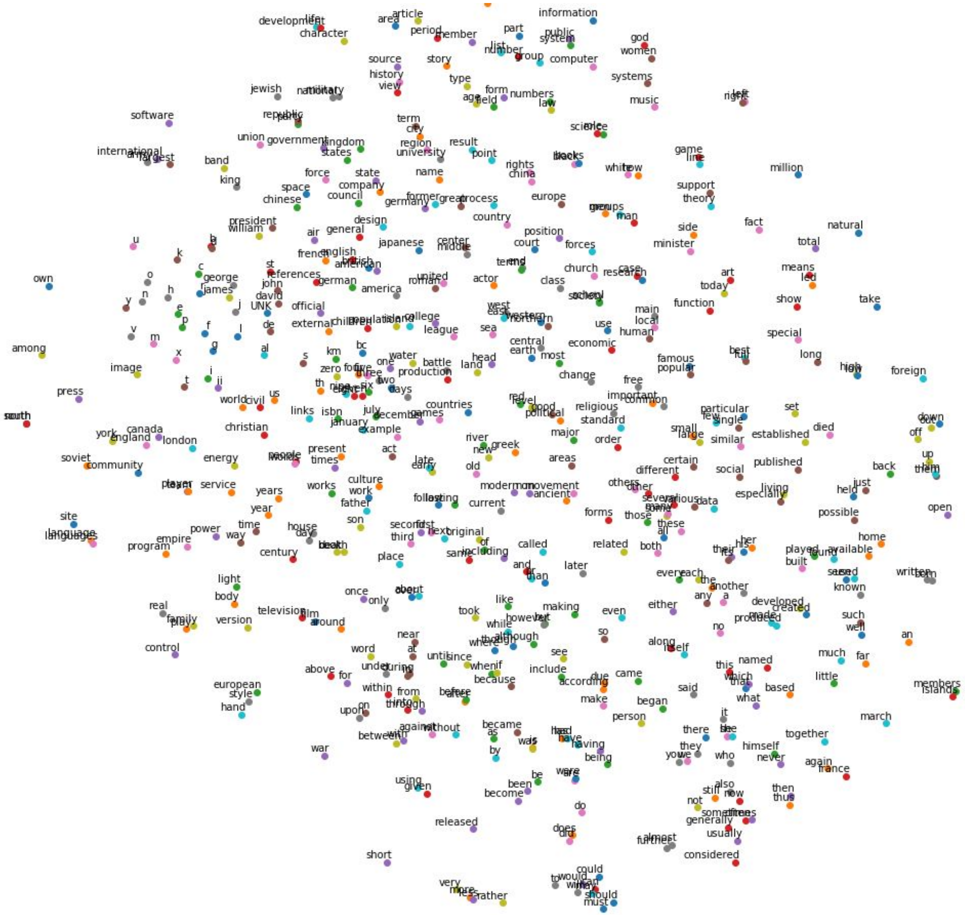
一種NLP工具, 包括2種主要訓練模型和2種訓練方法

(1) skip-gram & continuous bag-of-words

(2) negative sampling & hierarchical softmax

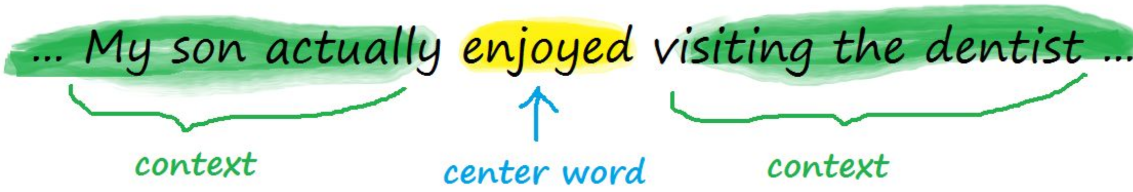
■ word2vec 解決什麼問題

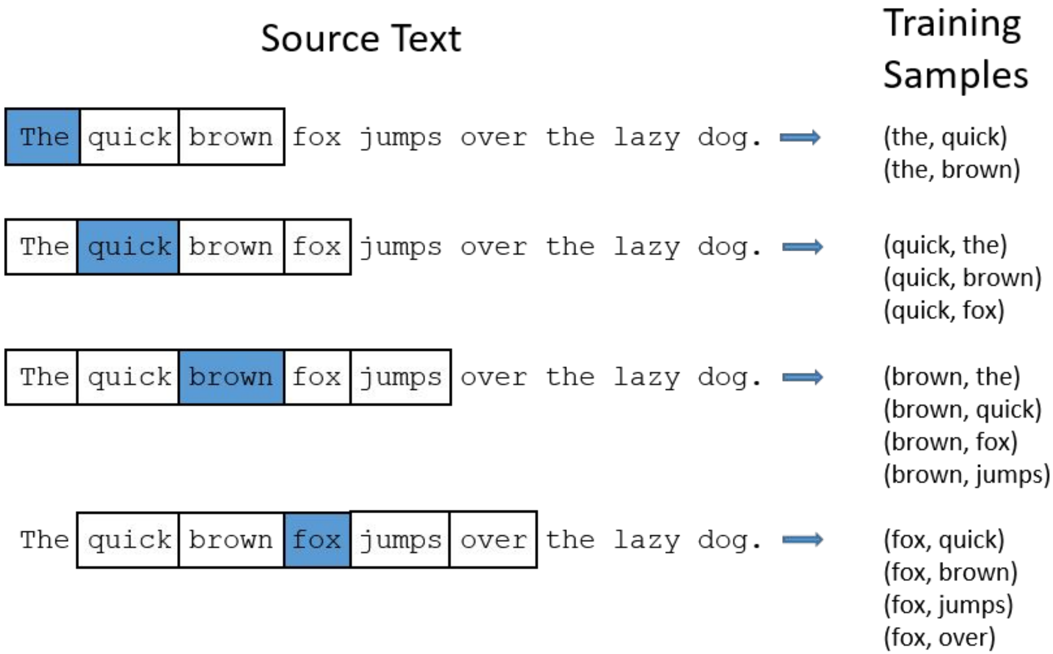
有效建立具備詞彙相關性的詞向量（Word Embedding)



名詞解釋

- Center word (中心詞)
- Context word (背景詞)
- context window size





Source Text

Training Samples

The quick brown fox jumps over the lazy dog. → (the, quick) (the, brown)

The quick brown fox jumps over the lazy dog. → (quick, the) (quick, brown) (quick, fox)

The quick brown fox jumps over the lazy dog. → (brown, the) (brown, quick) (brown, fox) (brown, jumps)

The quick brown fox jumps over the lazy dog. → (fox, quick) (fox, brown) (fox, jumps) (fox, over)

■ Skip-gram 目標:

假設詞典索引集V的大小為||V|, 且={0,1,...,||−1}V={0,1,...,|V|−1}。給定一個長度為TT的文本序列中, 時間步tt的詞為w(t)w(t)。當時間窗口大小為mm時, **skip-gram要最大化給定任一中心詞生成所有背景詞的概率**

$$\prod_{t=1}^{T} \prod_{-m \leq j \leq m,\, j \neq 0} \mathbb{P}(w^{(t+j)} \mid w^{(t)}).$$

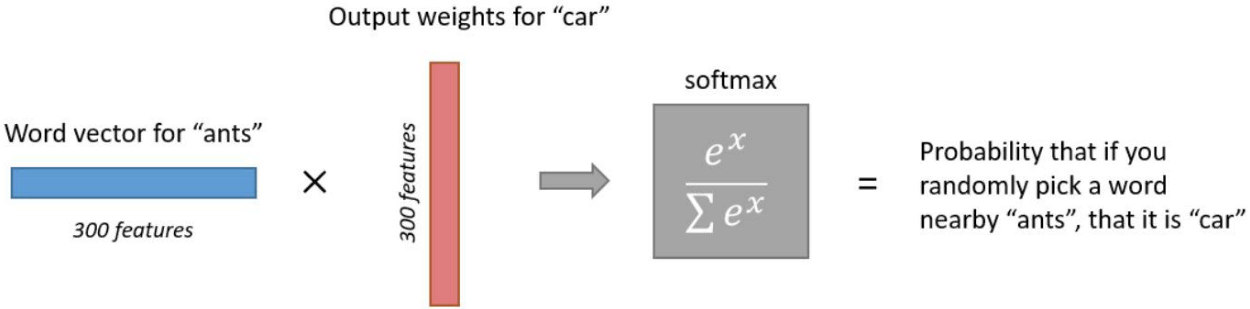
=> 最小化右側損失函數 (Cost function)

(連乘取log變連加)

$$-\frac{1}{T} \sum_{t=1}^{T} \sum_{-m \leq j \leq m,\, j \neq 0} \log \mathbb{P}(w^{(t+j)} \mid w^{(t)}).$$

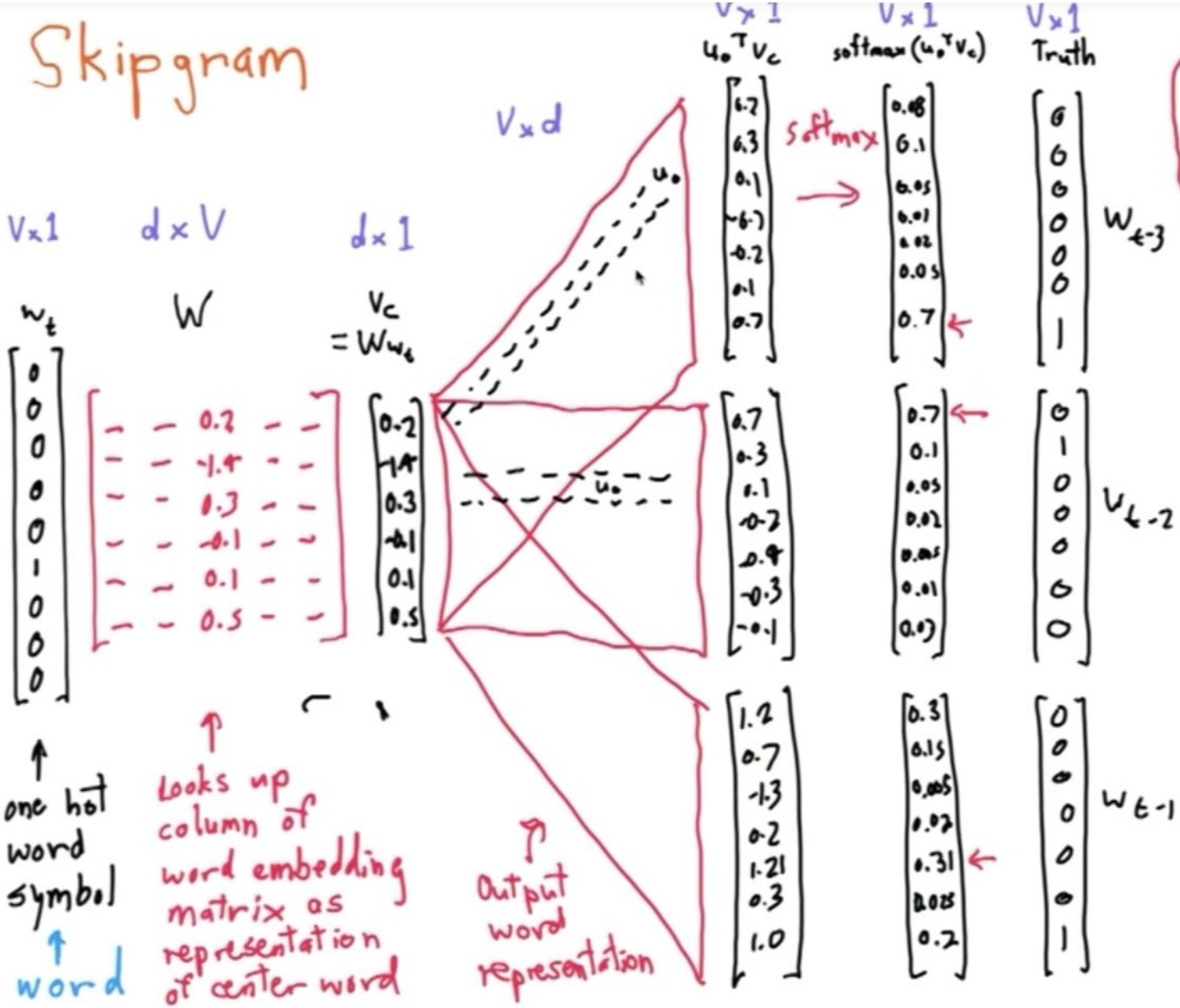
Eg: 當我們取得ants,

相鄰詞為car的機率



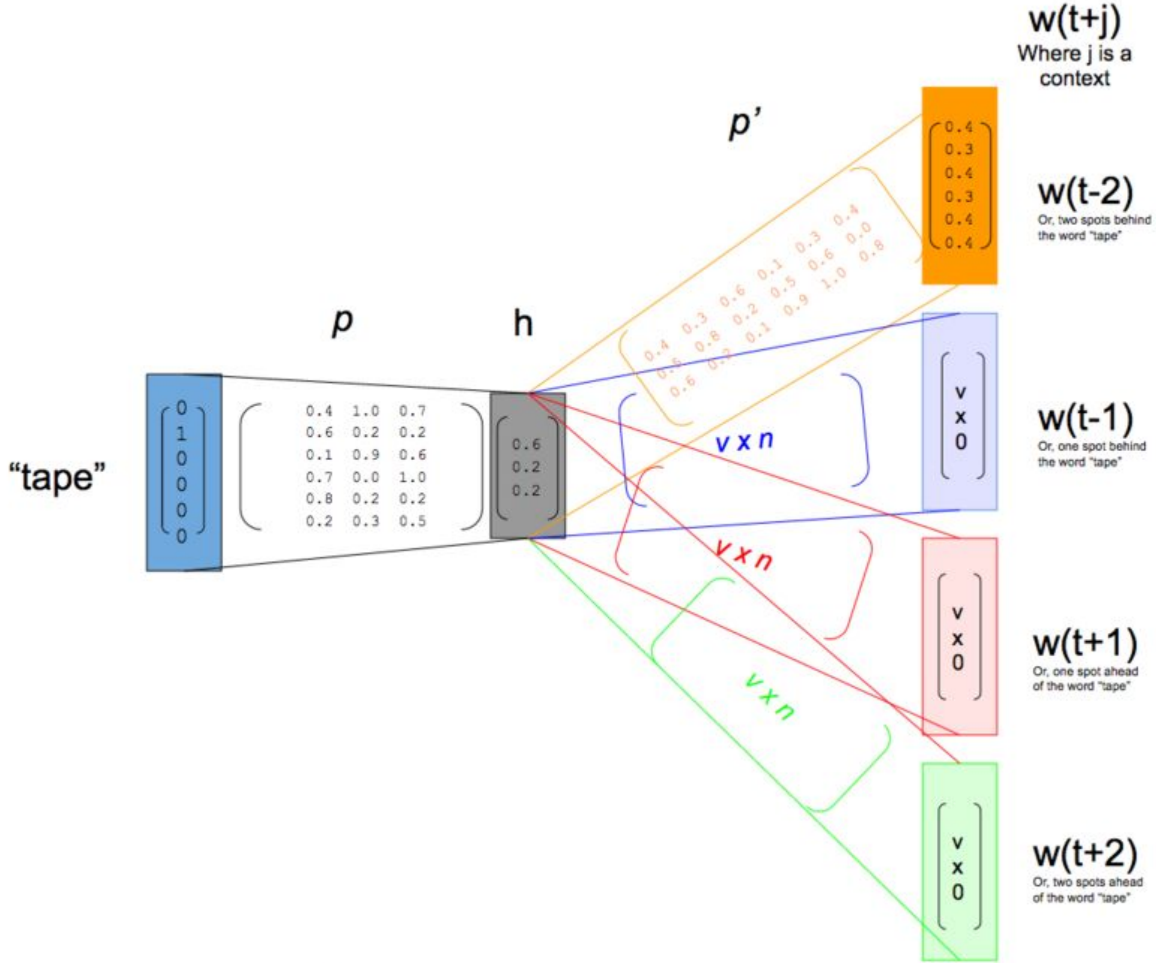
Probability that if you randomly pick a word nearby “ants”, that it is “car”

Skipgram

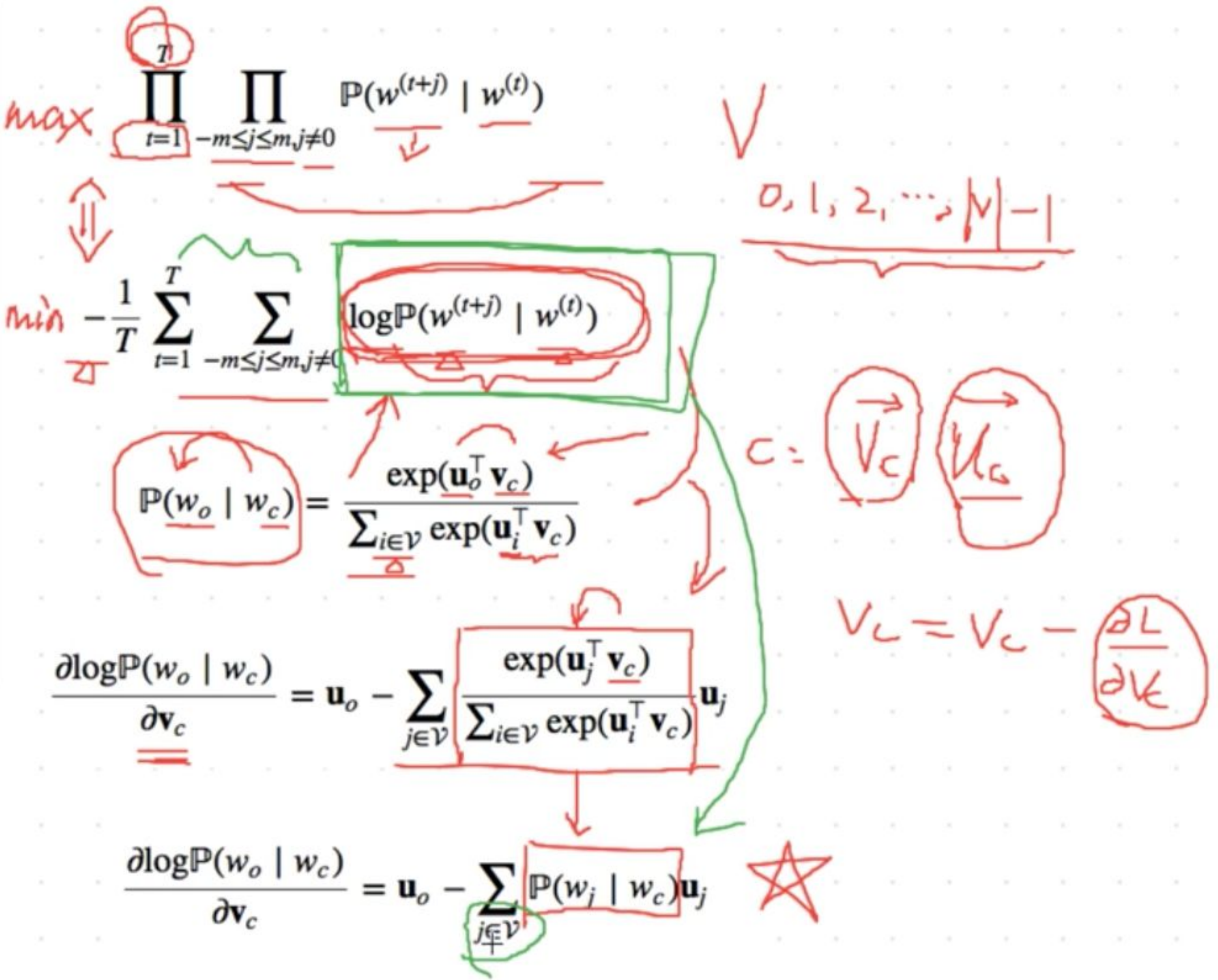


Softmax

$$p_i = \frac{e^{x_i}}{\sum_j e^{x_i}}$$



Sending w(t) = "tape" through the net once for context vector w(t-2) with randomized *p* and *p'* weight matrices



max
$$\prod_{t=1}^{T}\prod_{-m\leq j\leq m,\, j\neq 0}\mathbb{P}(w^{(t+j)} \mid w^{(t)})$$

V

0, 1, 2, ⋯, |V|-1

min
$$-\frac{1}{T}\sum_{t=1}^{T}\sum_{-m\leq j\leq m,\, j\neq 0}\log\mathbb{P}(w^{(t+j)} \mid w^{(t)})$$

$$\mathbb{P}(w_o \mid w_c) = \frac{\exp(\mathbf{u}_o^\top \mathbf{v}_c)}{\sum_{i\in\mathcal{V}}\exp(\mathbf{u}_i^\top \mathbf{v}_c)}$$

c: $\vec{V_c}$ $\vec{U_c}$

$V_c = V_c - \frac{\partial L}{\partial V_c}$

梯度(Gradient) =>

$$\frac{\partial\log\mathbb{P}(w_o \mid w_c)}{\partial\mathbf{v}_c} = \mathbf{u}_o - \sum_{j\in\mathcal{V}}\frac{\exp(\mathbf{u}_j^\top \mathbf{v}_c)}{\sum_{i\in\mathcal{V}}\exp(\mathbf{u}_i^\top \mathbf{v}_c)}\mathbf{u}_j$$

$$\frac{\partial\log\mathbb{P}(w_o \mid w_c)}{\partial\mathbf{v}_c} = \mathbf{u}_o - \sum_{j\in\mathcal{V}}\mathbb{P}(w_j \mid w_c)\mathbf{u}_j$$

■ CBOW 目標:

假設詞典索引集V的大小為|||V|，且={0,1,…,||−1}V={0,1,…,|V|−1}。給定一個長度為TT的文本序列中，時間步tt的詞為w(t)w(t)。當時間窗口大小為mm時，**CBOW模型需要最大化由背景詞生成任一中心詞的概率.**

$$\prod_{t=1}^{T} \mathbb{P}(w^{(t)} \mid w^{(t-m)}, \ldots, w^{(t-1)}, w^{(t+1)}, \ldots, w^{(t+m)}).$$

=> 最小化右側損失函數 (Cost function)

(連乘取log變連加)

$$-\sum_{t=1}^{T} \log \mathbb{P}(w^{(t)} \mid w^{(t-m)}, \ldots, w^{(t-1)}, w^{(t+1)}, \ldots, w^{(t+m)}).$$

Eg: 當我們取得ants,

相鄰詞為car的機率

■ CBOW 目標：

假設詞典索引集V的大小為|||V|，且={0,1,…,||−1}V={0,1,…,|V|−1}。

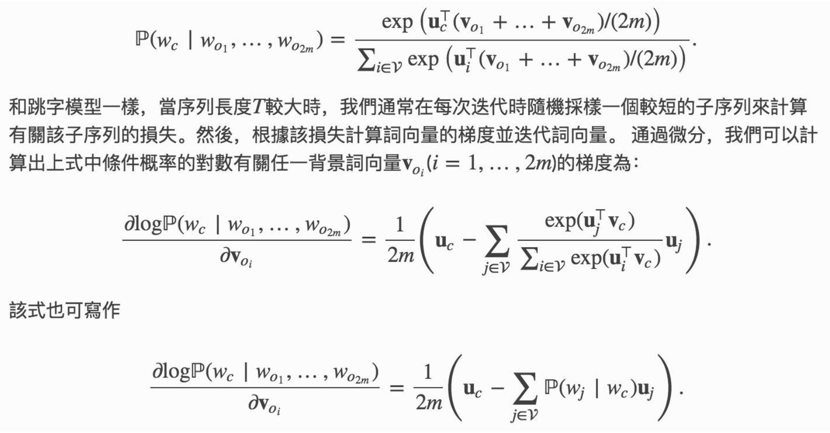
給定一個長度為TT的文本序列中，時間步tt的詞為w(t)w(t)。當時間窗口大小為mm時，**CBOW模型需要最大化由背景詞生成任一中心詞的概率.**

$$\prod_{t=1}^{T} \mathbb{P}(w^{(t)} \mid w^{(t-m)}, \ldots, w^{(t-1)}, w^{(t+1)}, \ldots, w^{(t+m)})$$

=> 最小化右側損失函數 (Cost function)

(連乘取log變連加)

$$-\sum_{t=1}^{T} \log \mathbb{P}(w^{(t)} \mid w^{(t-m)}, \ldots, w^{(t-1)}, w^{(t+1)}, \ldots, w^{(t+m)}).$$



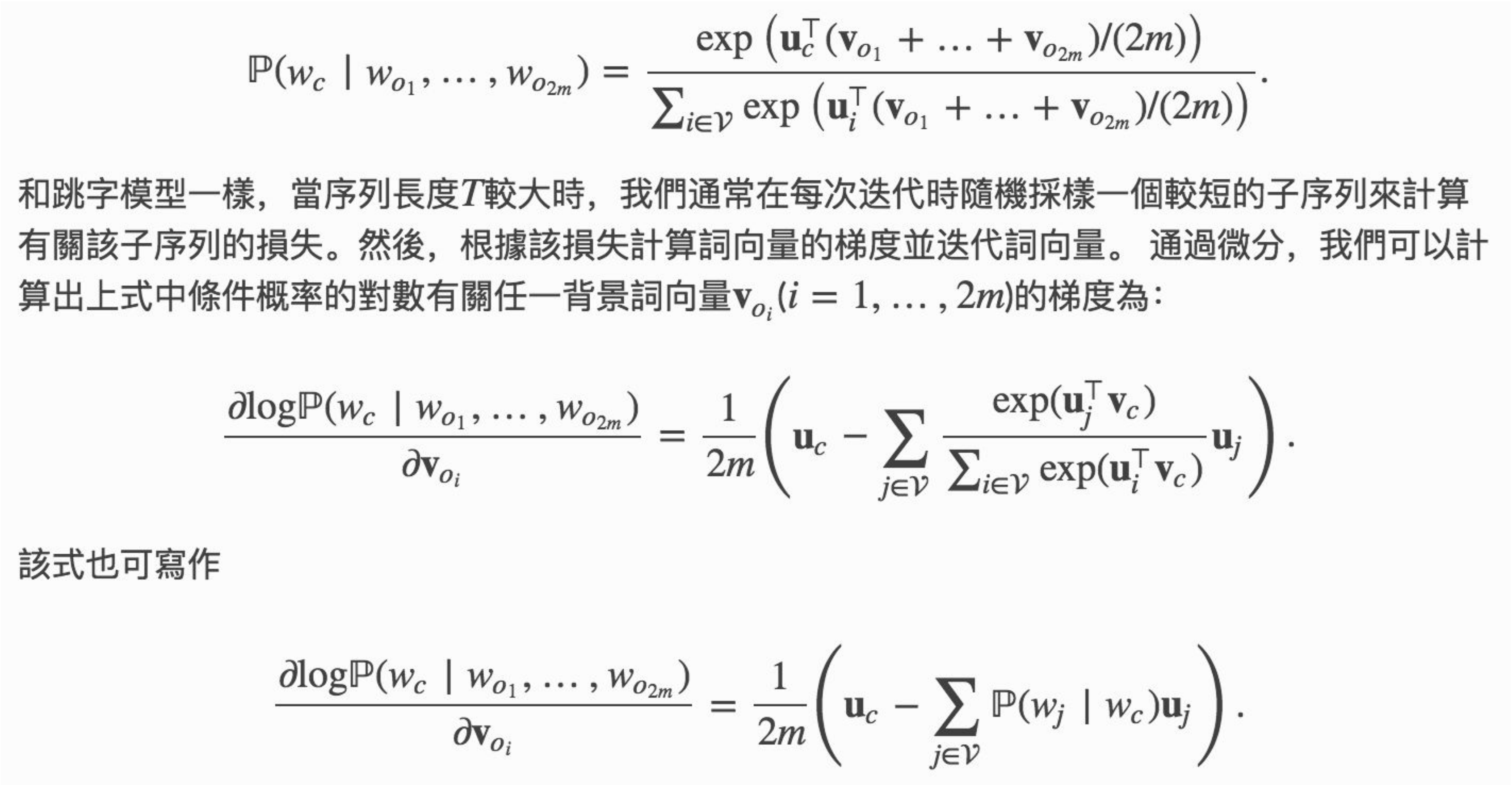
$$\mathbb{P}(w_c \mid w_{o_1}, \ldots, w_{o_{2m}}) = \frac{\exp\left(\mathbf{u}_c^\top (\mathbf{v}_{o_1} + \ldots + \mathbf{v}_{o_{2m}})/(2m)\right)}{\sum_{i\in\mathcal{V}} \exp\left(\mathbf{u}_i^\top (\mathbf{v}_{o_1} + \ldots + \mathbf{v}_{o_{2m}})/(2m)\right)}.$$

和跳字模型一樣，當序列長度T較大時，我們通常在每次迭代時隨機採樣一個較短的子序列來計算有關該子序列的損失。然後，根據該損失計算詞向量的梯度並迭代詞向量。 通過微分，我們可以計算出上式中條件概率的對數有關任一背景詞向量$\mathbf{v}_{o_i}(i = 1, \ldots, 2m)$的梯度為：

$$\frac{\partial \log \mathbb{P}(w_c \mid w_{o_1}, \ldots, w_{o_{2m}})}{\partial \mathbf{v}_{o_i}} = \frac{1}{2m}\left(\mathbf{u}_c - \sum_{j\in\mathcal{V}} \frac{\exp(\mathbf{u}_j^\top \mathbf{v}_c)}{\sum_{i\in\mathcal{V}} \exp(\mathbf{u}_i^\top \mathbf{v}_c)} \mathbf{u}_j\right).$$

該式也可寫作

$$\frac{\partial \log \mathbb{P}(w_c \mid w_{o_1}, \ldots, w_{o_{2m}})}{\partial \mathbf{v}_{o_i}} = \frac{1}{2m}\left(\mathbf{u}_c - \sum_{j\in\mathcal{V}} \mathbb{P}(w_j \mid w_c)\mathbf{u}_j\right).$$



$$\mathbb{P}(w_c \mid w_{o_1}, \ldots, w_{o_{2m}}) = \frac{\exp\left(\mathbf{u}_c^\top (\mathbf{v}_{o_1} + \ldots + \mathbf{v}_{o_{2m}})/(2m)\right)}{\sum_{i \in \mathcal{V}} \exp\left(\mathbf{u}_i^\top (\mathbf{v}_{o_1} + \ldots + \mathbf{v}_{o_{2m}})/(2m)\right)}.$$

和跳字模型一樣，當序列長度T較大時，我們通常在每次迭代時隨機採樣一個較短的子序列來計算有關該子序列的損失。然後，根據該損失計算詞向量的梯度並迭代詞向量。 通過微分，我們可以計算出上式中條件概率的對數有關任一背景詞向量$\mathbf{v}_{o_i}(i = 1, \ldots, 2m)$的梯度為：

$$\frac{\partial \log \mathbb{P}(w_c \mid w_{o_1}, \ldots, w_{o_{2m}})}{\partial \mathbf{v}_{o_i}} = \frac{1}{2m}\left(\mathbf{u}_c - \sum_{j \in \mathcal{V}} \frac{\exp(\mathbf{u}_j^\top \mathbf{v}_c)}{\sum_{i \in \mathcal{V}} \exp(\mathbf{u}_i^\top \mathbf{v}_c)} \mathbf{u}_j\right).$$

該式也可寫作

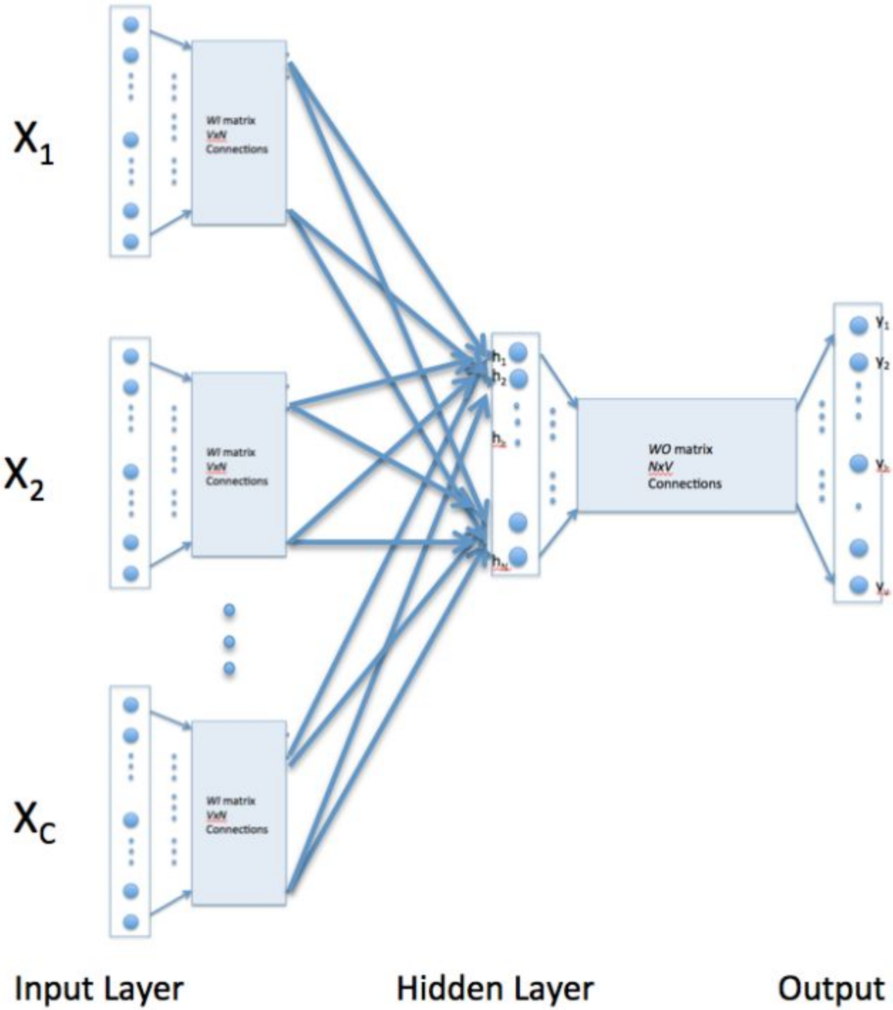
$$\frac{\partial \log \mathbb{P}(w_c \mid w_{o_1}, \ldots, w_{o_{2m}})}{\partial \mathbf{v}_{o_i}} = \frac{1}{2m}\left(\mathbf{u}_c - \sum_{j \in \mathcal{V}} \mathbb{P}(w_j \mid w_c) \mathbf{u}_j\right).$$

- 和skip-gram 一樣，**當序列長度T較大時，我們通常在每次迭代時隨機採樣一個較短的子序列來計算有關該子序列的損失**。然後，根據該損失計算詞向量的梯度並迭代詞向量。通過微分，我們可以計算出上式中條件概率的對數有關任一背景詞向量**v**oivoi(i=1,…,2mi=1,…,2m)的梯度為

$$\frac{\partial \log \mathbb{P}(w_c \mid w_{o_1}, \ldots, w_{o_{2m}})}{\partial \mathbf{v}_{o_i}} = \frac{1}{2m}\left(\mathbf{u}_c - \sum_{j \in \mathcal{V}} \frac{\exp(\mathbf{u}_j^\top \mathbf{v}_c)}{\sum_{i \in \mathcal{V}} \exp(\mathbf{u}_i^\top \mathbf{v}_c)} \mathbf{u}_j\right).$$

該式也可寫作

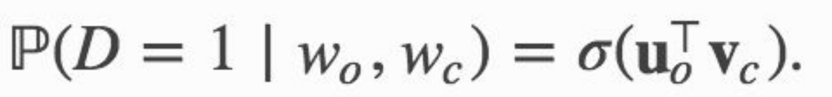
$$\frac{\partial \log \mathbb{P}(w_c \mid w_{o_1}, \ldots, w_{o_{2m}})}{\partial \mathbf{v}_{o_i}} = \frac{1}{2m}\left(\mathbf{u}_c - \sum_{j \in \mathcal{V}} \mathbb{P}(w_j \mid w_c)\mathbf{u}_j\right).$$



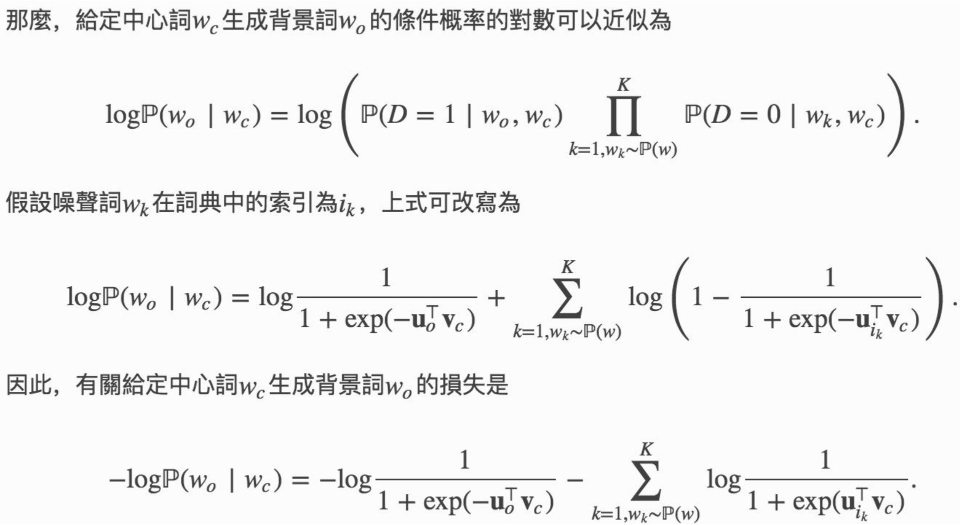
■ 訓練方法:nagative sampling

假設中心詞Wcwc生成背景詞Wowo由以下相互獨立事件聯合組成來近似:

- 中心詞Wcwc和背景詞Wowo同時出現時間窗口。
- 中心詞Wcwc和第1個噪聲詞W1w1不同時出現在該時間窗口（噪聲詞W1w1按噪聲詞分佈ℙ(W)P(w)隨機生成，且假設一定和Wc不同時出現在該時間窗口）。
- …
- 中心詞Wcwc和第KK個噪聲詞WKwK不同時出現在該時間窗口（噪聲詞WKwK按噪聲詞分佈ℙ(W)P(w)隨機生成，且假設一定和Wcwc不同時出現在該時間窗口）。



$$\mathbb{P}(D = 1 \mid w_o, w_c) = \sigma(\mathbf{u}_o^\top \mathbf{v}_c).$$



那麼，給定中心詞w_c生成背景詞w_o的條件概率的對數可以近似為

$$\log \mathbb{P}(w_o \mid w_c) = \log\left(\mathbb{P}(D = 1 \mid w_o, w_c) \prod_{k=1, w_k \sim \mathbb{P}(w)}^{K} \mathbb{P}(D = 0 \mid w_k, w_c)\right).$$

假設噪聲詞w_k在詞典中的索引為i_k，上式可改寫為

$$\log \mathbb{P}(w_o \mid w_c) = \log \frac{1}{1 + \exp(-\mathbf{u}_o^\top \mathbf{v}_c)} + \sum_{k=1, w_k \sim \mathbb{P}(w)}^{K} \log\left(1 - \frac{1}{1 + \exp(-\mathbf{u}_{i_k}^\top \mathbf{v}_c)}\right).$$

因此，有關給定中心詞w_c生成背景詞w_o的損失是

$$-\log \mathbb{P}(w_o \mid w_c) = -\log \frac{1}{1 + \exp(-\mathbf{u}_o^\top \mathbf{v}_c)} - \sum_{k=1, w_k \sim \mathbb{P}(w)}^{K} \log \frac{1}{1 + \exp(\mathbf{u}_{i_k}^\top \mathbf{v}_c)}.$$