

CH. 1 thesaurus & 통계 기반 방법

Thm 1.1 단어 베트워크를 통해 단어 사이의 유사도를 구할 수 있다. (WordNet)

Rmk 1.2 사전에는 사람이 손으로 단어를 연결 것이다.

Def 1.3 Corpus (корпус) : 대량의 텍스트 데이터

ex) text = "you say goodbye and i say hello."

Preprocess
⇒ id-to-word ...
 word-to-id ...
Corpus = [0, 1, 2, 3, 4, 1, 5, 6]

Def 1.4 distributional representation (단어의 분산 표현)

dense vector
word ⇒ [0.21, -0.45, 0.83]

Thm 1.5 distributional hypothesis

'단어의 의미는 주변 단어에 의해 형성된다.'

Def 1.6 co-occurrence_matrix

"you say goodbye and i say hello."

ex)

	you	say	goodbye	and	i	hello	.
you	0	1	0	0	0	0	0
say	1	0	1	0	1	1	0

Def. 1.7 cosine similarity

Let $x, y \in \mathbb{R}^n$, then similarity : $\mathbb{R}^n \times \mathbb{R}^n \rightarrow \mathbb{R}$ defined by $\text{similarity}(x, y) = \frac{x \cdot y}{\|x\| \|y\|}$

Def. 1.8 Pointwise Mutual Information (PMI)

Motivation) The car >> car drive (빈도가 다른 어휘가...)

$\text{PMI} : E \times E \rightarrow \mathbb{R}$

$$\text{s.t. } \text{PMI}(x, y) = \log_2 \frac{P(x, y)}{P(x)P(y)} = \log_2 \frac{\frac{C(x, y)}{N}}{\frac{C(x)}{N} \cdot \frac{C(y)}{N}} = \log_2 \frac{N \cdot C(x, y)}{C(x) \cdot C(y)}$$

Note 1.9

Let $N = 10,000$ and $C(\text{"the"}) = 1000$, $C(\text{"car"}) = 20$, $C(\text{"drive"}) = 10$

$$\text{then, } \begin{array}{c} \text{PMI}(\text{"the", "car"}) \\ \text{ " } \\ 2.32 \end{array} < \begin{array}{c} \text{PMI}(\text{"car", "drive"}) \\ \text{ " } \\ 1.917 \end{array}$$

Note 1.10

If $P(x, y) \rightarrow 0$, $\text{PMI}(x, y) \rightarrow -\infty$

$$\text{Thus } \boxed{\text{PPMI}(x, y) = \max(0, \text{PMI}(x, y))}$$

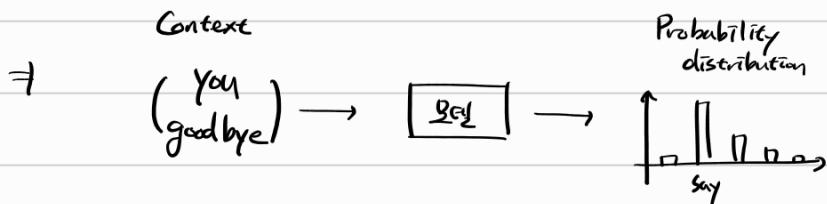
Rmk 1.11 SVD for PPMI matrix!

In Particular, we would use truncated SVD for large N

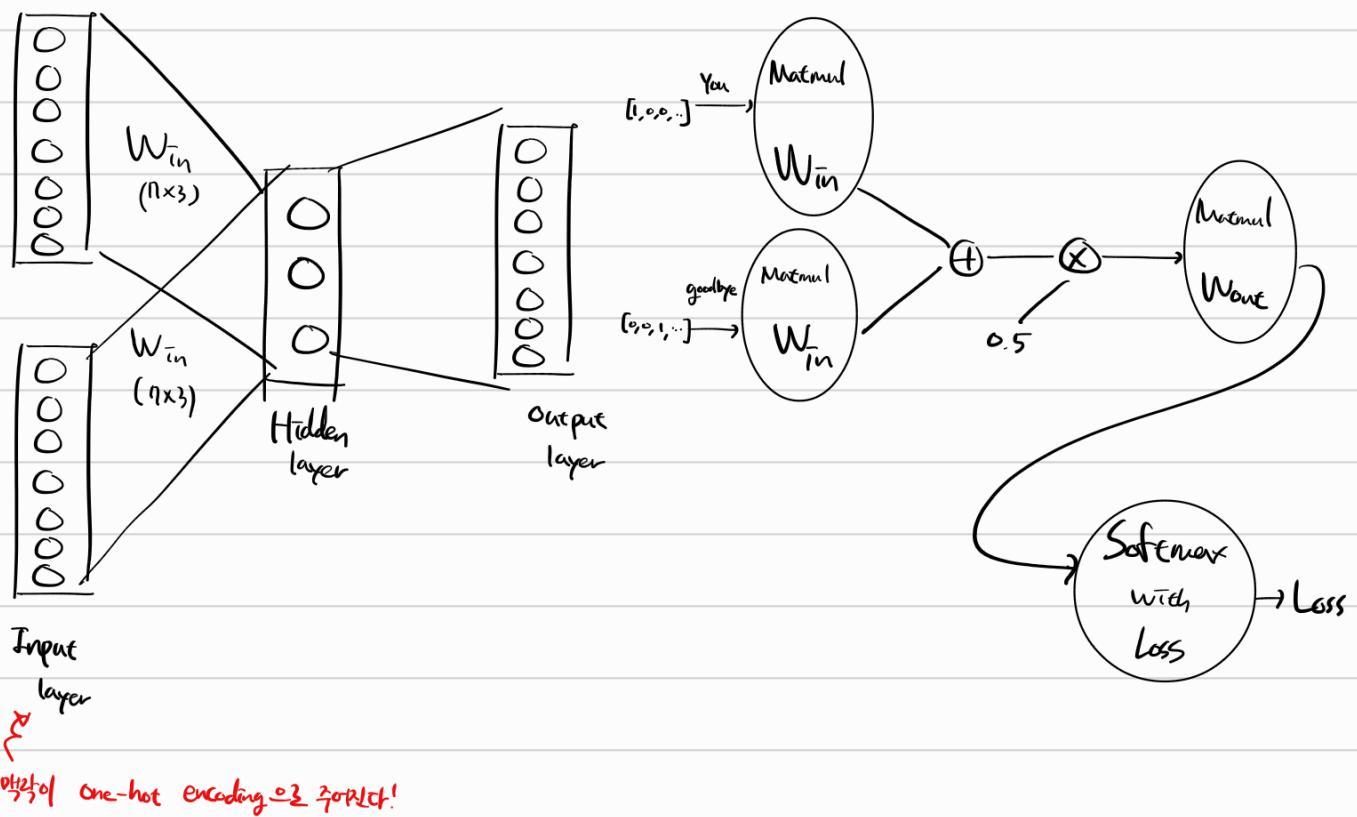
CH.2 추론 기반 기법 (Word2vec)

Intro.

You [?] good bye and I say hello.



def 2.1 CBOW (Continuous bag-of-word)



Note 2.2 ↳ 여기서 W_{in} 이 바로 단어의 분산 표현!

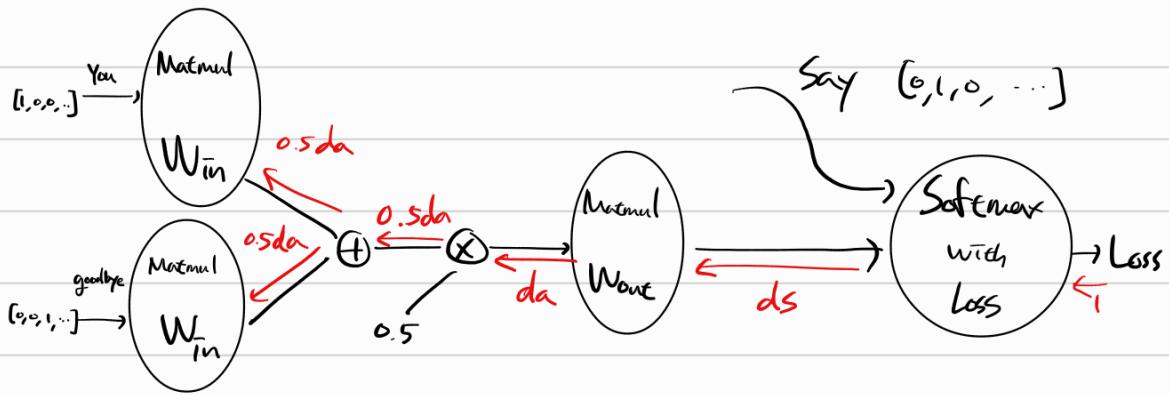
Rmk 2.3 CBOW의 size

Let N be a corpus size & w window size
 L be a length of text

Then Contexts target

$$(L-w, w, N), (L-w, N)$$

Thm 2.4 back propagation of CBOW



def 2.5 negative log likelihood

$$w_1 w_2 \dots w_{t-1} \boxed{w_t} a_{t+1} \dots w_T$$

$$\text{Loss} = -\sum_k t_k \log y_k$$

$t_k \in X_{w_t}$ 로 볼수 있다. 고지,

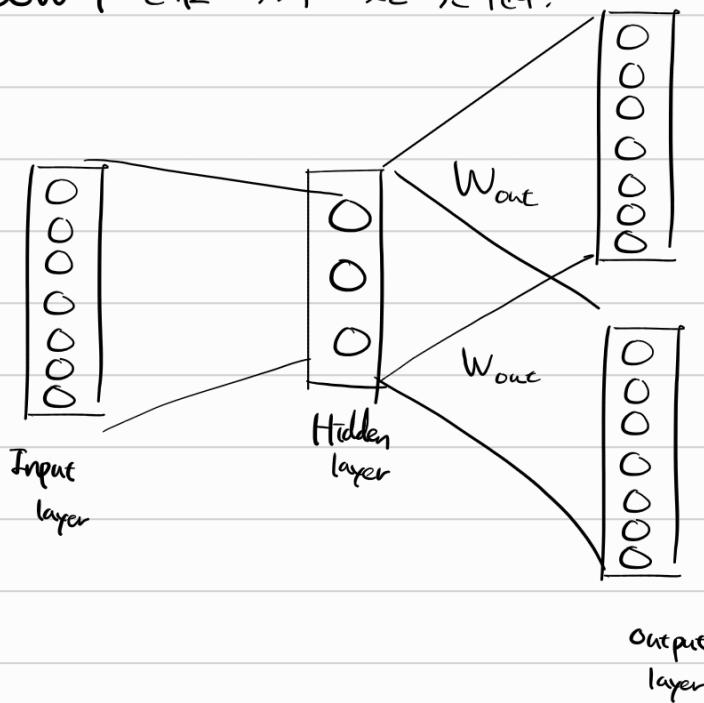
$$= -\log P(w_t | w_{t-1}, w_{t+1})$$

In Particular, Using entire corpus

$$L = -\frac{1}{T} \sum_{t=1}^T \log P(w_t | w_{t-1}, w_{t+1})$$

def 2.6 skip-gram

* CBOW 와 반대로 맥락과 타겟을 역전시킨다!



[?] Say [?] and I say hello.

Rmk. 2.7 skip-gram's Loss

$$\begin{aligned}
 L &= -\log P(w_{t+1}, w_{t+2} | w_t) \\
 &= -\log P(w_{t+1} | w_t) \cdot P(w_{t+2} | w_t) \\
 &= -[\log P(w_{t+1} | w_t) + \log P(w_{t+2} | w_t)]
 \end{aligned}$$

w_{t+1}
 w_{t+2}은
 조건부
 둘다이다 가정

$$\Rightarrow L = -\frac{1}{T} \sum_{t=1}^T \log P(w_{t+1} | w_t) + \log P(w_{t+2} | w_t)$$

Note 2.8 Comparison

단어 분산 표현의 정밀도 관점, 특히 저빈도 단어나 유추 문제에서 good.

Performance skip-gram > CBow

train speed skip-gram < CBow

Thm 2.9 통계 기반 vs 추론 기반, 단어의 분산 표현 특징의 차이



주로 단어의 유사성

유사성 + 복잡한 단어 사이의 패턴 유사

ex) King-man + woman = Queen!

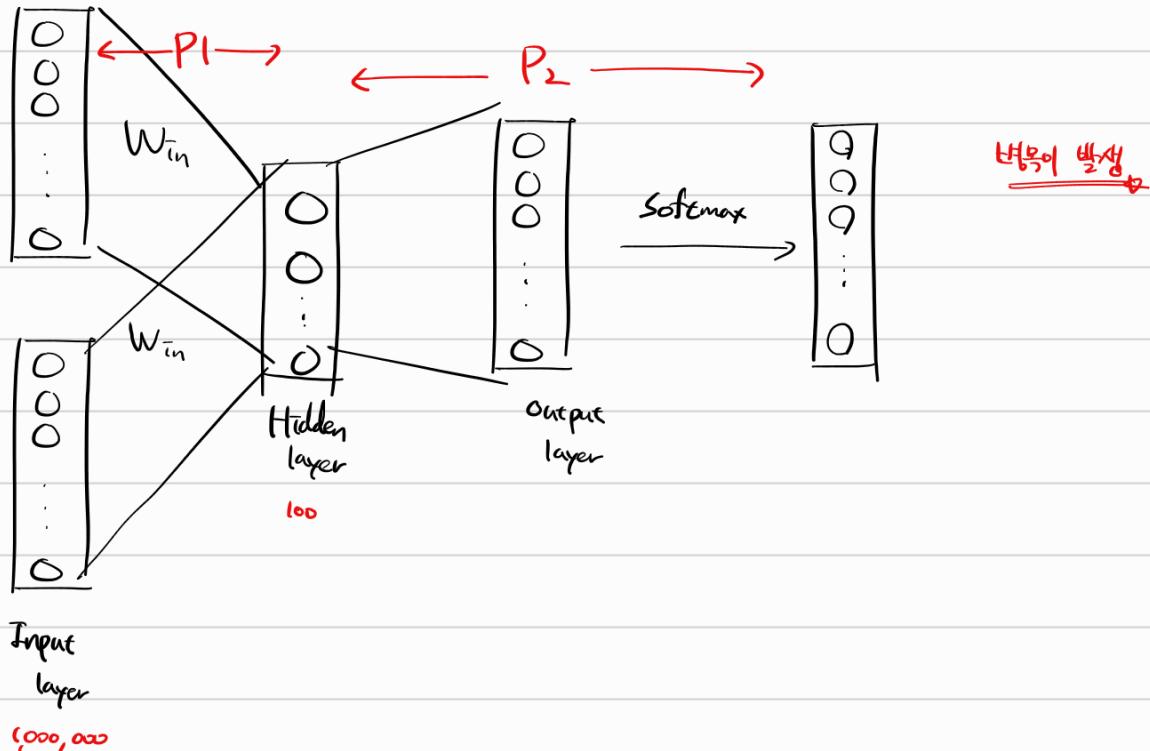
☞ 추가로 데이터셋에 약간의 수정이 일어났을 때 추론 기반 기법이 나을 것!

☞ 통계 기반은 차원부터 해석 가능...

현재, 통계 기반 정보를 Loss에 도입해 미니 배치 학습을 하는 Glove라는 것도 있음!

CH.3 Word2Vec

Motivation $\text{vocab} \quad \text{len}(\text{corpus}) = 1,000,000$ 이 데면 ??



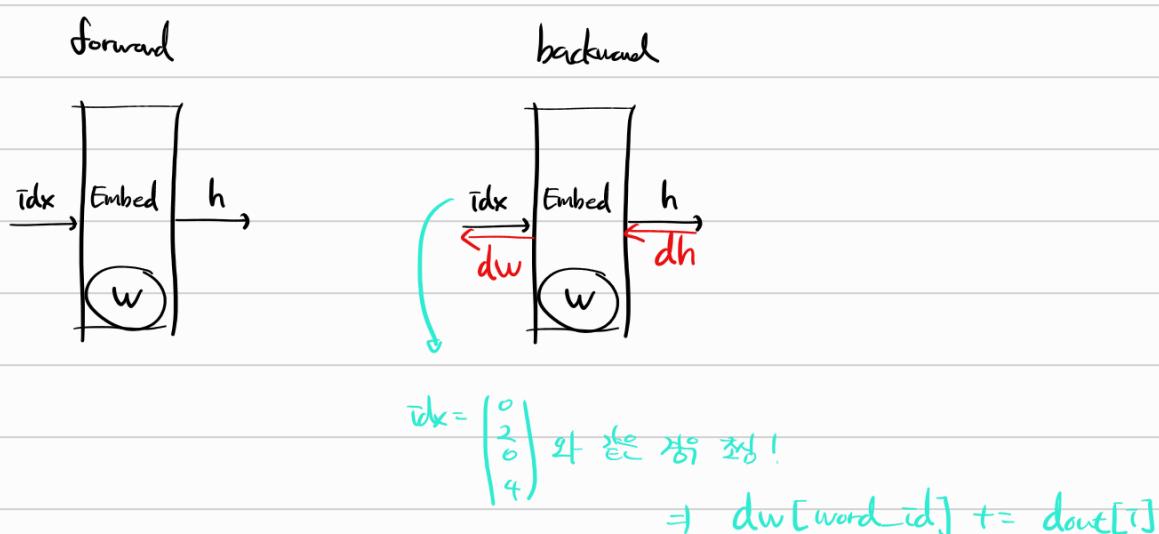
S1 : Embedding

S2 : Negative sampling.

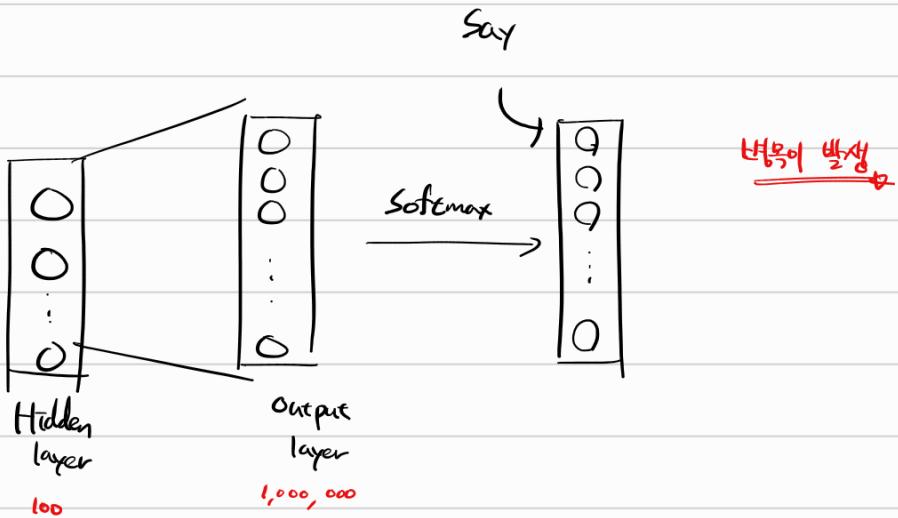
Note 3.1

사실 P1에서 하는 일은 그저 W에서 특정 행을 추출하는 것 뿐이다.

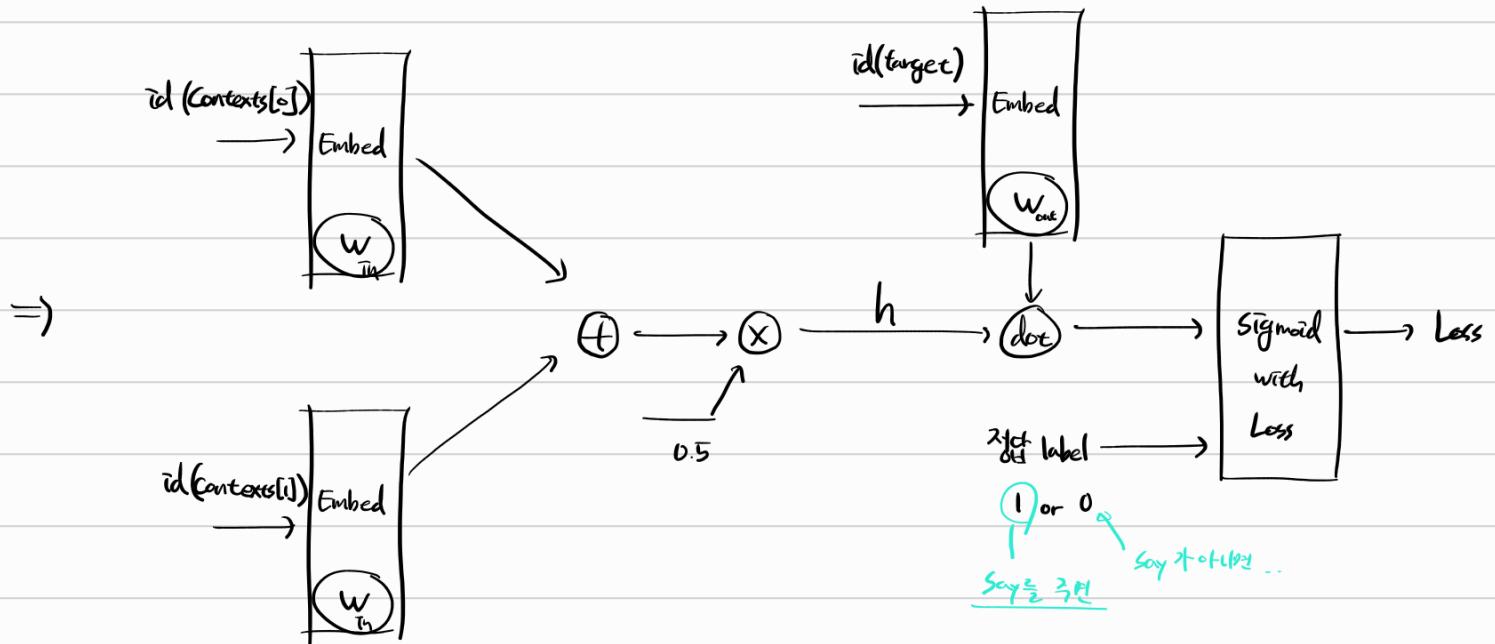
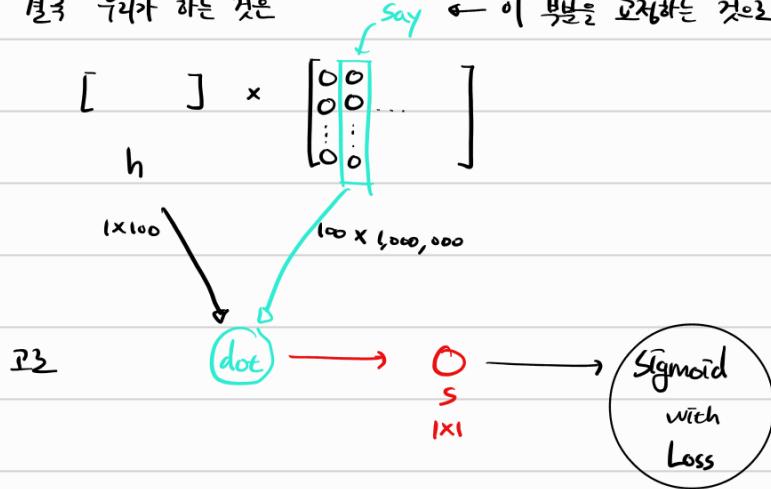
Def 3.2 Embedding



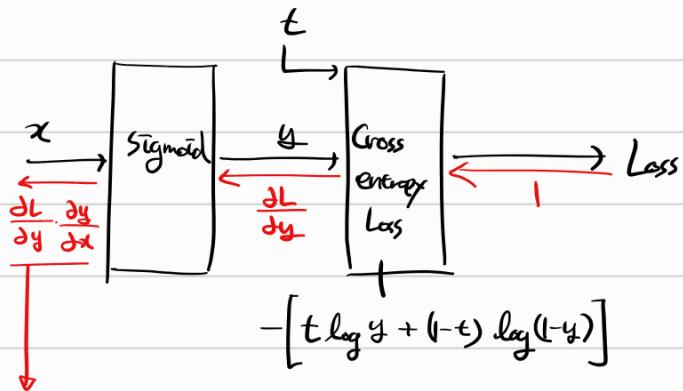
Thm 3.3 다중불규어에서 이진불규로!



결국 우리가 하는 것은 *say* ← 이 부분을 고정하는 것으로 단순화 할 수 있다.



Def 3.4 Sigmoid with Loss (binary cross entropy loss)



$$\frac{\partial L}{\partial y} = -\left(\frac{t}{y} - \frac{1-t}{1-y}\right) = \frac{y-t}{y(1-y)}$$

$$\frac{\partial y}{\partial x} = y(1-y)$$

$$\Rightarrow \frac{\partial L}{\partial y} \cdot \frac{\partial y}{\partial x} = y-t$$

◻

Thm 3.5 Negative sampling.

정답이 아닌 경우에 대해서 확률을 낮춰야 한다!

$$\Rightarrow L' = \sum_{i \neq k} (1-t) \log(1-y_{i \neq k}) \quad \text{where } t=0, \text{ for some } k$$

In Particular, we would use id_k s.t frequently used!

Plus we use probability distribution s.t $P(w_j) = \frac{P(w_j)^{0.05}}{\sum_j P(w_j)^{0.05}}$

↳ 확률이 낮은 값은 제거하기 위함