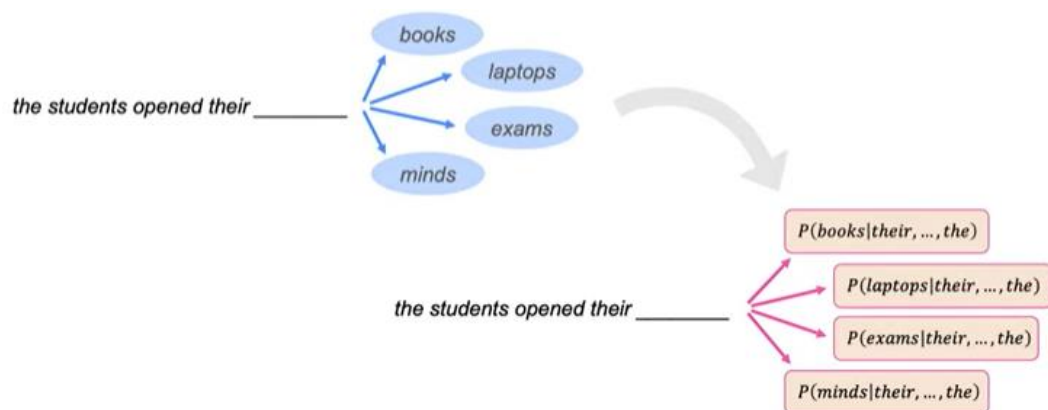


CS224N Lecture6. Language Models and Recurrent Neural Networks

* Language Model 이란?

- 단어의 시퀀스에 대해서 얼마나 자연스러운 문장인지를 확률을 통해 예측
- 주어진 단어의 시퀀스에 대해서 다음에 나타날 단어가 어떤 것인지 예측하는 작업을 Language Modeling이라고 한다.



$$w_1, w_2, \dots, w_{t-n+1}, \dots, w_{t-1}, w_t, \dots, w_{T-1}, w_T$$

$$P(w_1, \dots, w_T) = P(w_1) \times P(w_2|w_1) \times \dots \times P(w_T|w_{T-1}, \dots, w_1)$$

$$= \prod_{t=1}^T P(w_t|w_{t-1}, \dots, w_1)$$

* n-gram Language Models

- Neural Network 이전에 사용되었던 Language Model
- 예측에 사용할 앞 단어들의 개수를 정하여 모델링하는 방법

(uni-grams, bi-grams, tri-grams, 4-grams)

(앞 단어 n개를 가지고 다음에 올 단어를 예측한다.)

- > uni-grams: "the", "students", "opened", "their"
- > bi-grams: "the students", "students opened", "opened their"
- > tri-grams: "the students opened", "students opened their"
- > 4-grams: "the students opened their"

$$w_1, w_2, \dots, w_{t-n+1}, \dots, w_{t-1}, w_t, \dots, w_{T-1}, w_T$$

$$P(w_t | w_{t-1}, \dots, w_1) \approx P(w_t | w_{t-1}, \dots, w_{t-n+1}) \quad (\text{assumption})$$

$$\begin{aligned} \text{prob of a } n\text{-gram} & \rightarrow P(w_t, w_{t-1}, \dots, w_{t-n+1}) \\ \text{prob of a } (n-1)\text{-gram} & \rightarrow P(w_{t-1}, \dots, w_{t-n+1}) \end{aligned}$$

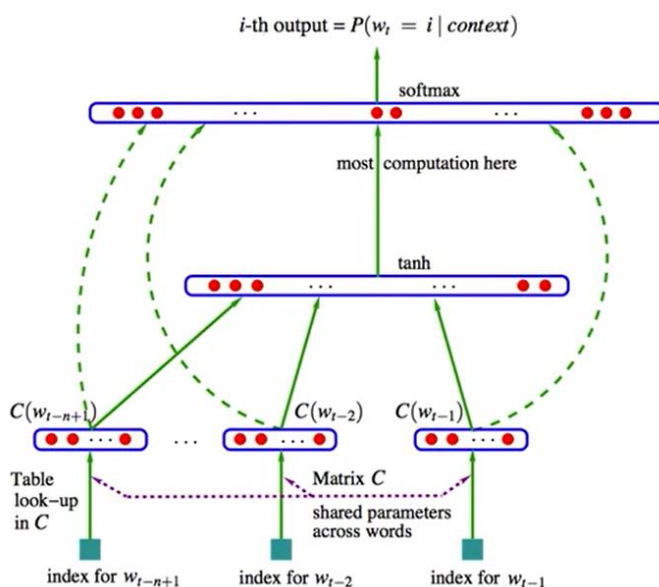
$$= \frac{P(w_t, w_{t-1}, \dots, w_{t-n+1})}{P(w_{t-1}, \dots, w_{t-n+1})} \quad (\text{definition of conditional prob})$$

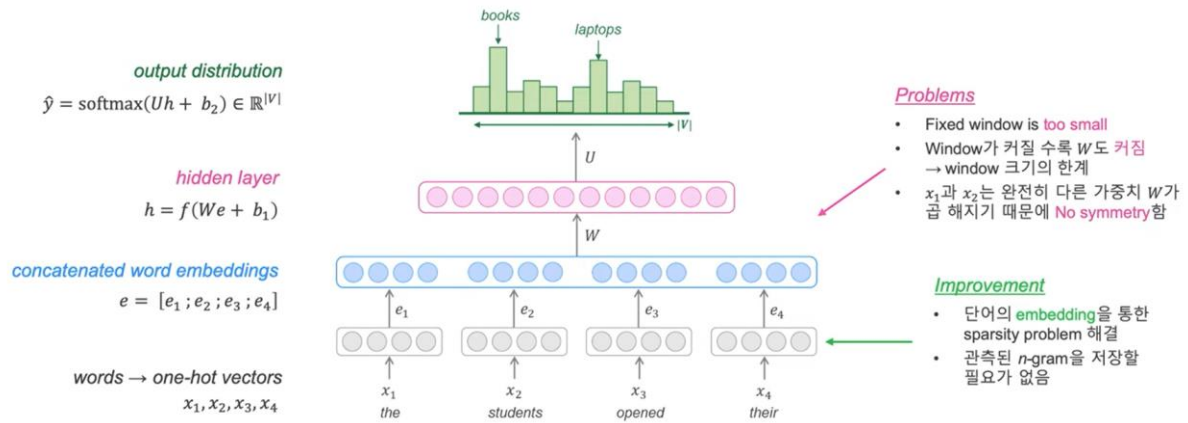
* n-gram Language Models의 문제점

- Sparsity problems: n 이 커질수록 성능이 떨어지며, 일반적으로 n 값으로 5가 되지 않는 수를 설정함.
- Storage problems: n 이 커지거나 corpus가 증가하면 모델의 크기가 증가함.

* Window-based Neural Network Language Model(NNLM)

- 'curse of dimensionality'를 해결하기 위해 제안된 신경망 기반 Language Model
- Language Model이면서 동시에 단어의 'distributed representation' 학습





* Recurrent Neural Networks(RNN)

- Take sequential input of any length
- Apply the same weights on each step
- Can optionally produce output on each step

A RNN Language Model

output distribution

$$\hat{y}^{(t)} = \text{softmax}(Uh^{(t)} + b_2) \in \mathbb{R}^{|V|}$$

hidden states

$$h^{(t)} = \sigma(W_h h^{(t-1)} + W_e e^{(t)} + b_1)$$

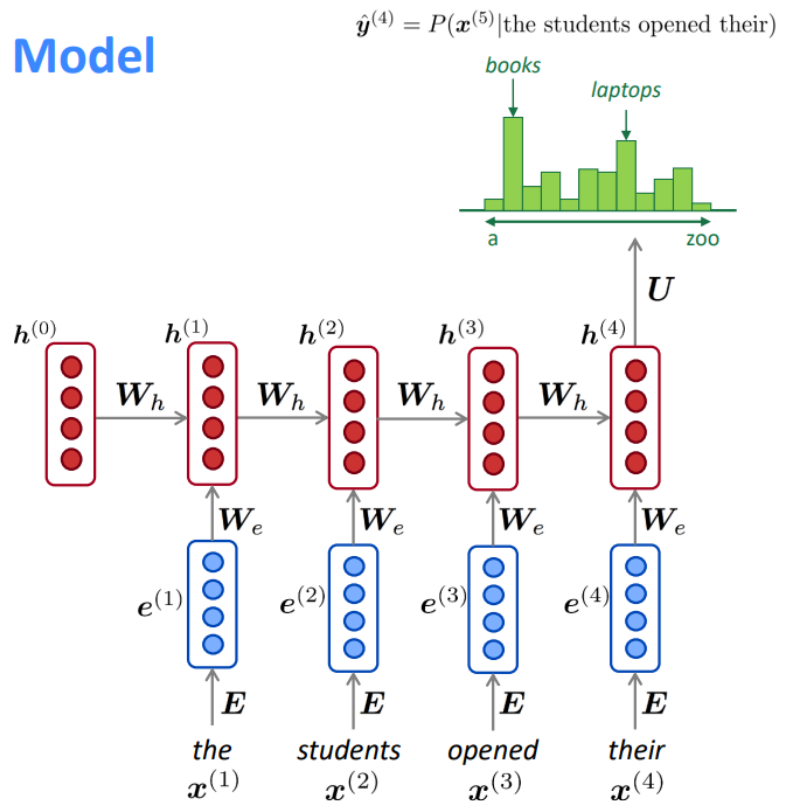
$h^{(0)}$ is the initial hidden state

word embeddings

$$e^{(t)} = E x^{(t)}$$

words / one-hot vectors

$$x^{(t)} \in \mathbb{R}^{|V|}$$



Note: this input sequence could be much longer, but this slide doesn't have space!