


RNN 실습

언어모델 (Language model)

- 구글



what is the | 

what is the **weather**
what is the **meaning of life**
what is the **dark web**
what is the **xfl**
what is the **doomsday clock**
what is the **weather today**
what is the **keto diet**
what is the **american dream**
what is the **speed of light**
what is the **bill of rights**

Google Search I'm Feeling Lucky

언어모델 (Language model)

- 언어모델은 다음에 올 단어를 예측한다.
- 주어진 단어 시퀀스 $x(1), x(2), \dots, x(t)$ 가 주어졌을 때, 다음 단어 $x(t+1)$ 의 출현 확률을 계산한다.

$$P(x(t+1)|x(t), \dots, x(1))$$

여기서 $x(t+1)$ 은 단어집 $V = \{w_1, \dots, w_{|V|}\}$ 의 단어이다.

- 이러한 작업을 하는 시스템을 언어 모델이라 한다.

언어모델 (Language model)

- 언어모델을 텍스트가 나올 확률을 계산하는 것으로도 생각할 수 있다.
- 예를 들어 어떤 텍스트 $x(1), x(2), \dots, x(T)$ 가 있을 때, 이 텍스트가 나올 확률은 다음과 같이 구할 수 있다.

$$\begin{aligned} P(\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(T)}) &= P(\mathbf{x}^{(1)}) \times P(\mathbf{x}^{(2)} | \mathbf{x}^{(1)}) \times \dots \times P(\mathbf{x}^{(T)} | \mathbf{x}^{(T-1)}, \dots, \mathbf{x}^{(1)}) \\ &= \prod_{t=1}^T P(\mathbf{x}^{(t)} | \mathbf{x}^{(t-1)}, \dots, \mathbf{x}^{(1)}) \end{aligned}$$

언어모델 (Language model)

- N그램: n그램이란 연속된 단어 덩어리이다.

예제: The students opened their _____

- 유니그램(unigram): the, students, opened, their
- 바이그램(bigram): the students, student opened, opened their
- 트라이그램(trigram): the students opened, students opened their
- 4그램: the students opened their

- 상이한 n그램이 어떤 빈도로 출현하는 지에 대한 통계량을 구하고 이를 이용해서 다음 단어를 예측한다.

언어모델 (Language model)

- 고전적 n그램 언어모델: $x(t+1)$ 이 이전의 $n-1$ 단어에 의해 결정된다.

$$P(x^{(t+1)} | x^{(t)}, \dots, x^{(1)}) = P(x^{(t+1)} | \overbrace{x^{(t)}, \dots, x^{(t-n+2)}}^{n-1 \text{ words}}) \quad (\text{assumption})$$

prob of a n-gram \rightarrow

prob of a (n-1)-gram \rightarrow

$$= \frac{P(x^{(t+1)}, x^{(t)}, \dots, x^{(t-n+2)})}{P(x^{(t)}, \dots, x^{(t-n+2)})} \quad (\text{definition of conditional prob})$$

$$\approx \frac{\text{count}(x^{(t+1)}, x^{(t)}, \dots, x^{(t-n+2)})}{\text{count}(x^{(t)}, \dots, x^{(t-n+2)})} \quad (\text{statistical approximation})$$

언어모델 (Language model)

- 예제: 4그램 언어모델

~~as the proctor started the clock, the~~ students opened their _____
discard condition on this

$$P(w|\text{students opened their}) = \frac{\text{count}(\text{students opened their } w)}{\text{count}(\text{students opened their})}$$

- 예를 들어 우리의 데이터베이스에서
 - "students opened their"가 1000번 나오고,
 - "students opened their books"가 400번 나오면,
 - > $P(\text{books}|\text{students opened their})=0.4$
 - "students opened their exams"가 100번 나오면,
 - > $P(\text{exams}|\text{students opened their})=0.1$
- ※ 만약 procter를 버리지 않았으면 더 추론하기 쉬웠을 것이다.

언어모델 (Language model)

- 신경망 모델의 등장 (고정 윈도우)

output distribution

$$\hat{y} = \text{softmax}(U\mathbf{h} + \mathbf{b}_2) \in \mathbb{R}^{|V|}$$

hidden layer

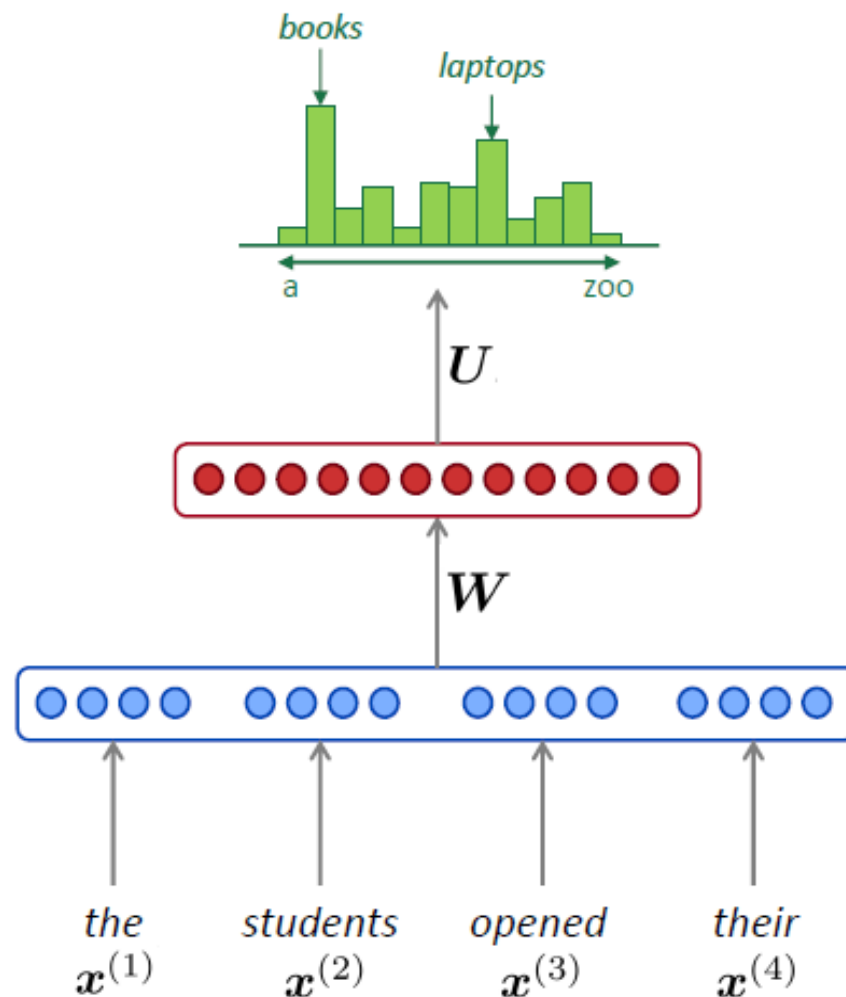
$$\mathbf{h} = f(\mathbf{W}\mathbf{e} + \mathbf{b}_1)$$

concatenated word embeddings

$$\mathbf{e} = [\mathbf{e}^{(1)}; \mathbf{e}^{(2)}; \mathbf{e}^{(3)}; \mathbf{e}^{(4)}]$$

words / one-hot vectors

$$\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \mathbf{x}^{(3)}, \mathbf{x}^{(4)}$$



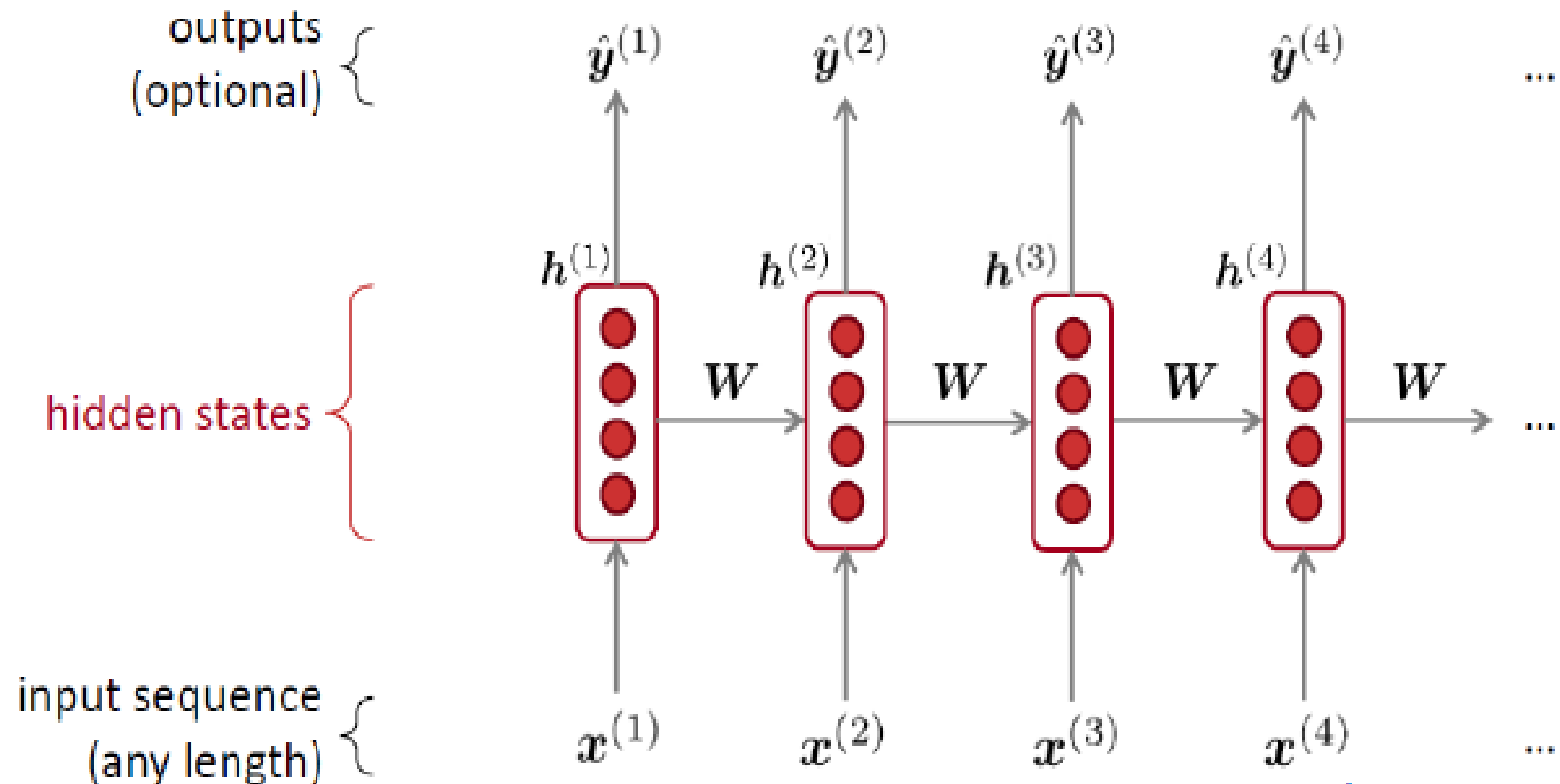
언어모델 (Language model)

- 신경망 모델의 등장 (고정 윈도우)의 문제
 - 고정 윈도우는 너무 작다.
 - 장기 의존성을 반영할 수 없다.
 - 윈도우는 클수록 좋다. -> 하지만 너무 많은 파라미터

=> 어떤 길이의 입력도 처리할 수 있는 신경망 구조의
필요성

언어모델 (Language model)

- 순환신경망: 동일한 W 가 반복적으로 적용



언어모델 (Language model)

A Simple 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

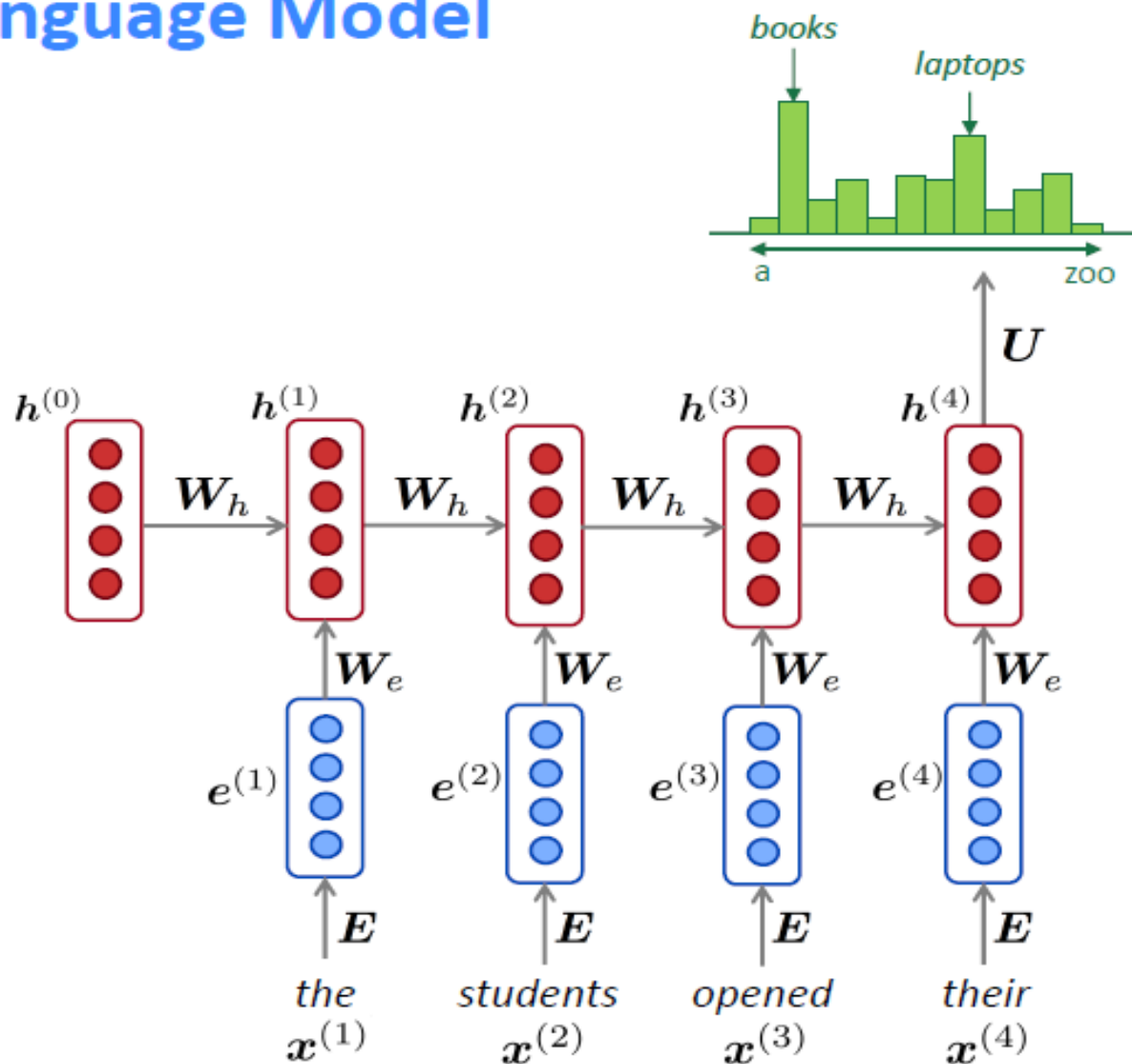
word embeddings

$$e^{(t)} = Ex^{(t)}$$

words / one-hot vectors

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

$$\hat{y}^{(4)} = P(x^{(5)} | \text{the students opened their})$$



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

언어모델 (Language model)

- RNN의 이점
 - 어떤 길이의 입력도 처리 가능
 - 많은 스텝의 정보를 처리
 - 입력이 길어져도 모델 크기가 증가하지 않음
 - 동일한 가중치가 매 스텝에서 반복 적용되므로, 입력의 처리의 대칭성 유지
- RNN의 문제점
 - 계산이 매우 느리다.
 - 실제로 많은 스텝에서 정보를 처리하는 것은 힘들다.

텐서플로우 2

- RNN 언어모델

https://www.tensorflow.org/tutorials/text/text_generation