삼성전기 AI전문가 양성과정 - 프로젝트 실습 (비영상)

자연어처리를 위한 Language Model

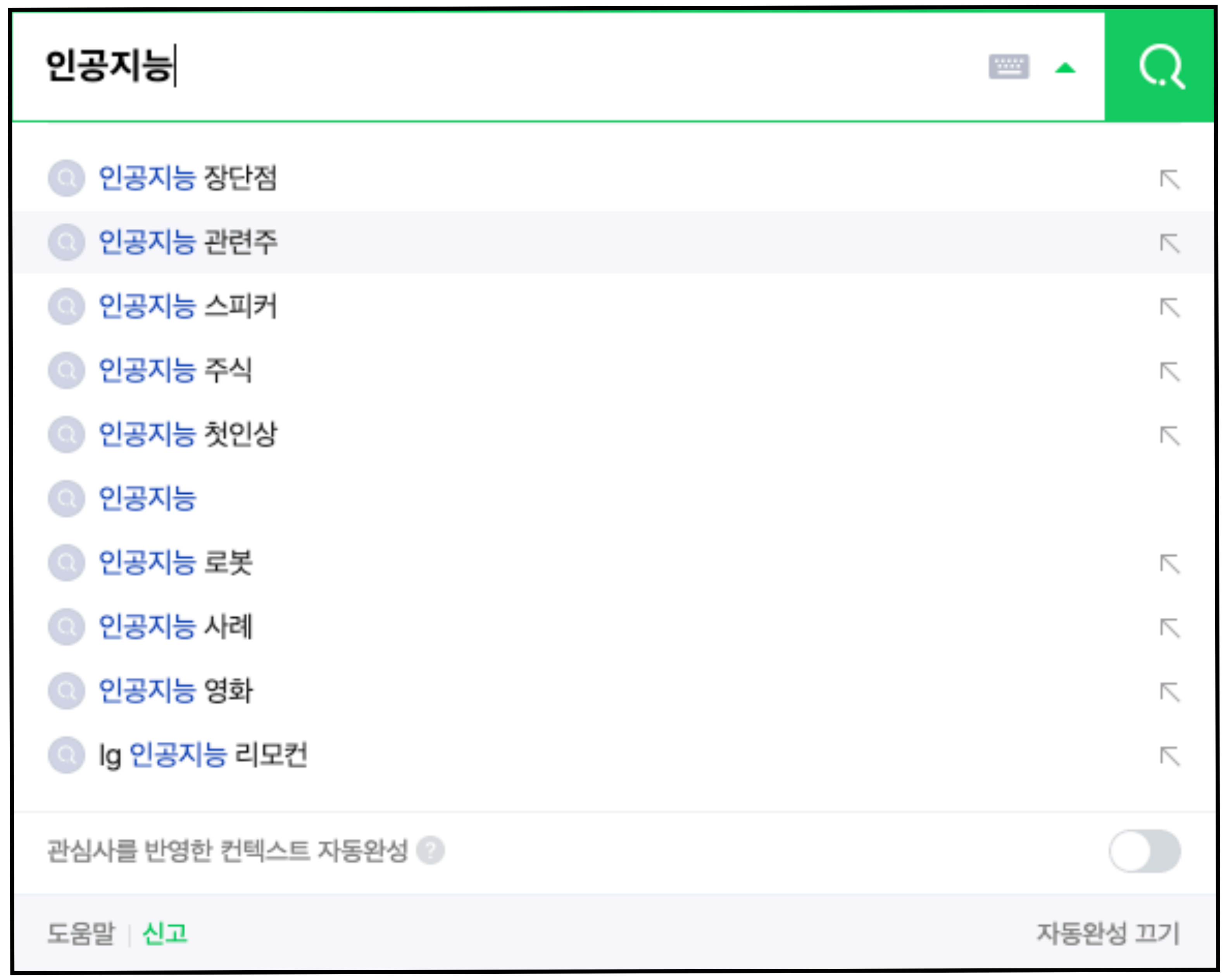
현청천

2022.02.28

What is Language Model

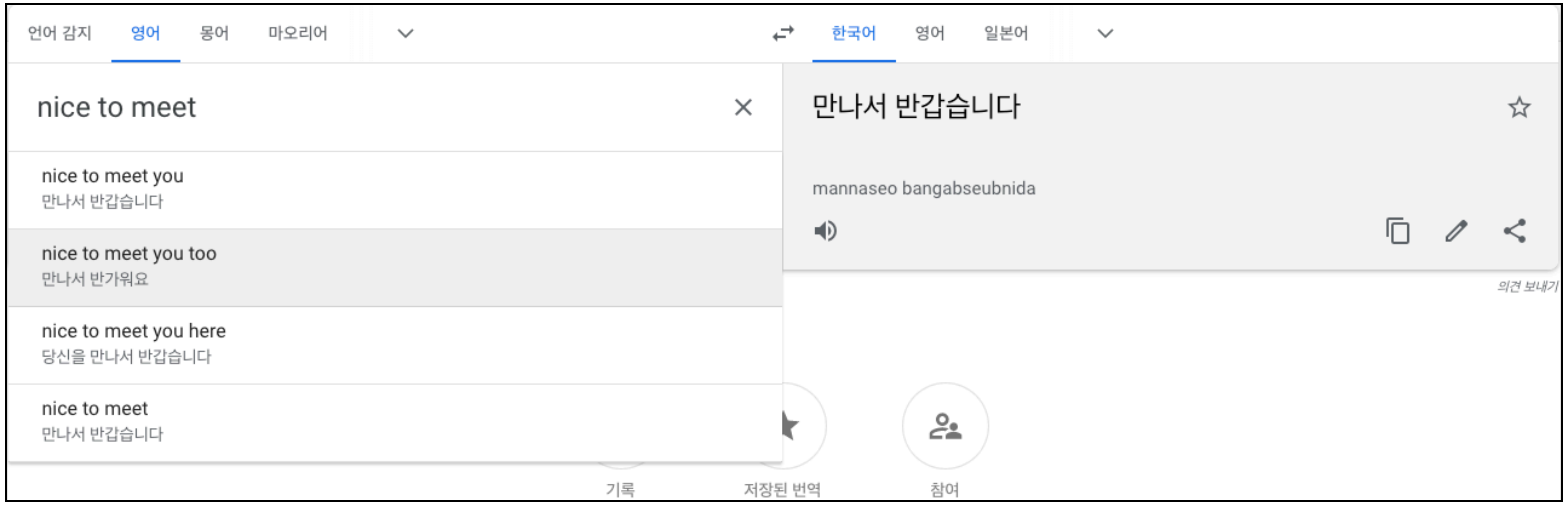
Language Model은 언어의 확률분포를 추정하는 것

What is Language Model



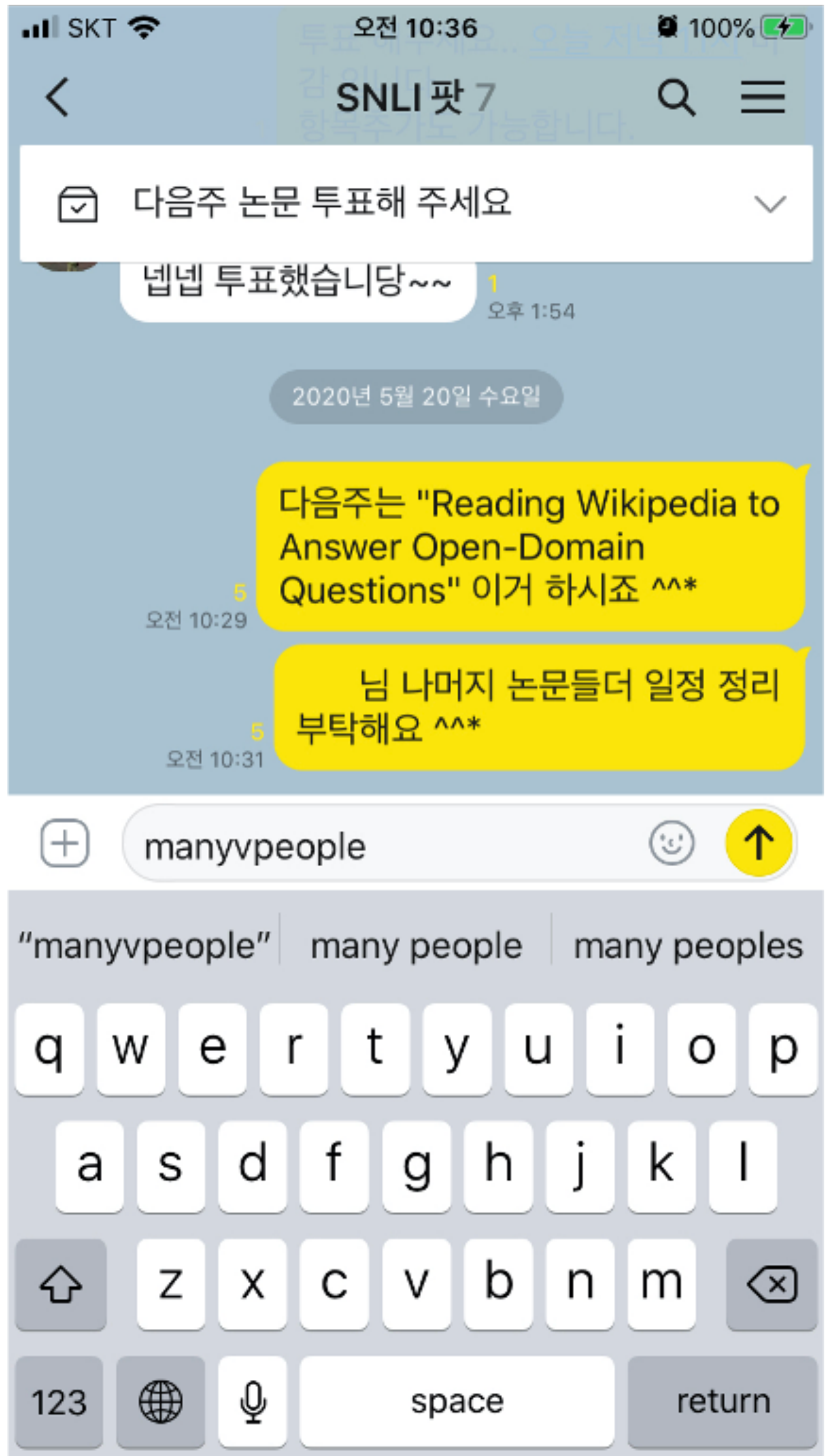
자동완성

What is Language Model



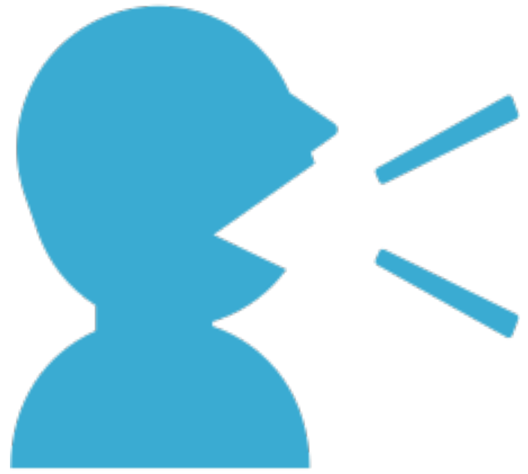
자동완성

What is Language Model



오타

What is Language Model



아버지가방에들어가신다

아버지가 방에 들어가신다

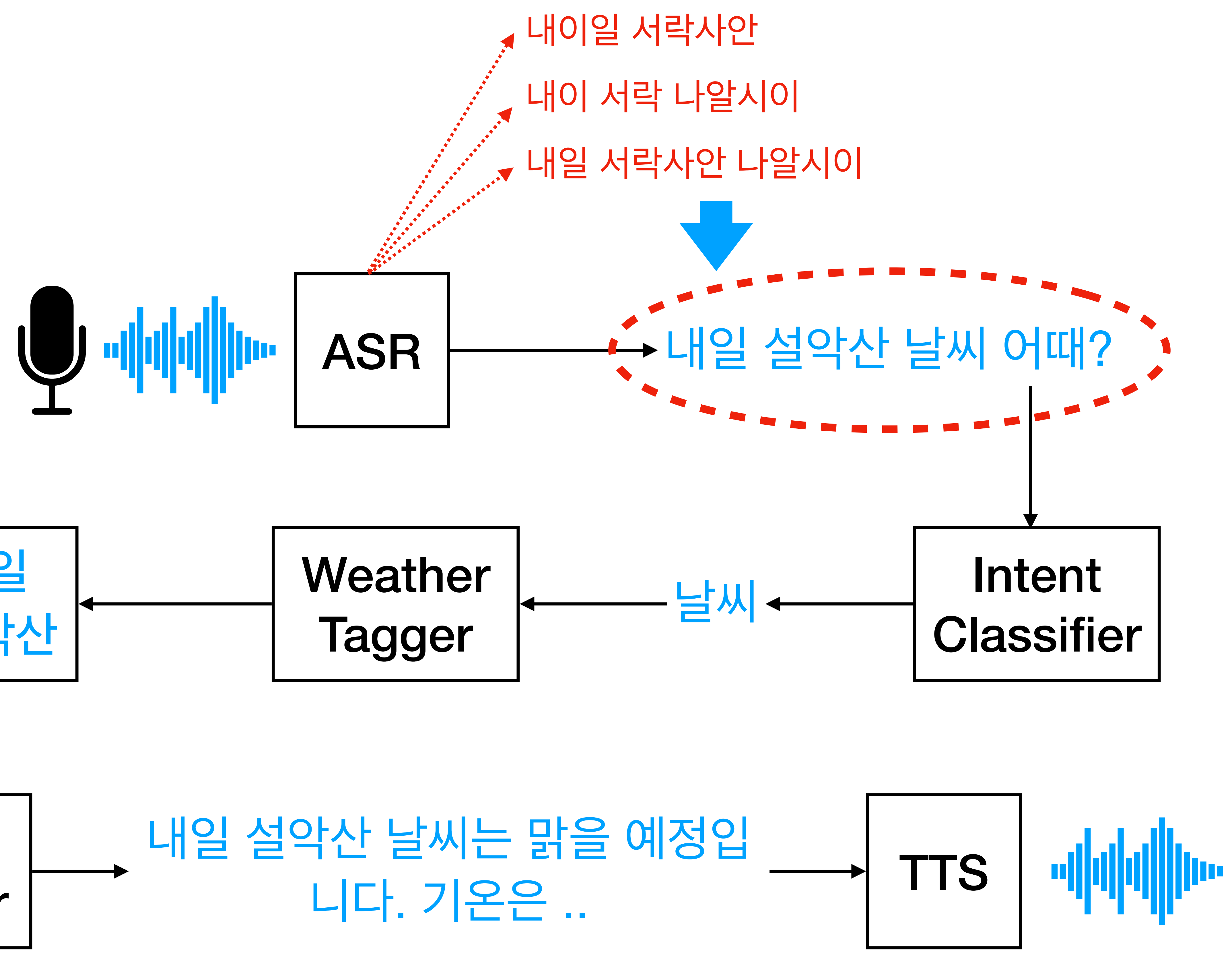
아버지 가방에 들어가신다

아버지가방 들어가신다

아버지가 방에 들어가신다

사람간의 대화

What is Language Model



음성인식에서 Language Model을 이용해 음성을 문자로 변환

What is Language Model

자연어에서 발생할 확률

p(그는 사과를 보자 배고픔을 느꼈다) $>$ p(그는 사과를 보자 외로움을 느꼈다)

p(그녀는 운동을 열심히 한다) $>$ p(그녀는 운동을 떳떳이 한다)

What is Language Model

실제 언어의 확률분포를 아는 것은 어려움

좋은 근사치를 제공하는 Language Model을 정의 할 수 있음

N-gram Language Model

48봉지 2620개의 M&M의 컬러 분포

372	544	369	483	481	371

이 데이터로부터 확률 분포를 추론하는 방법은?

$$p(color) = \frac{count(color)}{N}, \quad N = \sum_{color} count(color)$$

N-gram Language Model

Word sequence로부터 확률 분포를 추론하는 방법

$$s = (w^{(1)}, w^{(2)}, \ldots, w^{(n)})$$

많은 text corpus가 있다면

이 corpus로부터 확률 분포를 추론 할 수 있음

N-gram Language Model

$p(s = w^{(1)}, w^{(2)}, \ldots, w^{(n)})$

$p(s = the\ cat\ slept\ quietly)$

$p(w^{(1)} = the, w^{(2)} = cat, w^{(3)} = slept, w^{(4)} = quietly)$

$p(quietly \,|\, the\ cat\ slept) \cdot p(slept \,|\, the\ cat) \cdot p(cat \,|\, the) \cdot p(the)$

$$p(w^{(1)}, w^{(2)}, \ldots, w^{(n)}) = \prod_{i=1}^{n} p(w^{(i)} \,|\, w^{(1)}, \ldots, w^{(i-1)})$$

N-gram Language Model

Independent Assumption

단어의 분포는 고정된 몇 개의 이전 단어에 의존함

$$p(w^{(i)} \mid w^{(i)}, w^{(2)}, \ldots, w^{(i-1)}) \dashrightarrow p(w^{(i)} \mid w^{(i-n+1)}, w^{(i-n+2)}, \ldots, w^{(i-1)})$$

Trigram: $p(w^{(i)} \mid w^{(1)}, w^{(2)}, \ldots, w^{(i-1)}) \approx p(w^{(i)} \mid w^{(i-2)}, w^{(i-1)})$

bigram: $p(w^{(i)} \mid w^{(1)}, w^{(2)}, \ldots, w^{(i-1)}) \approx p(w^{(i)} \mid w^{(i-1)})$

unigram: $p(w^{(i)} \mid w^{(1)}, w^{(2)}, \ldots, w^{(i-1)}) \approx p(w^{(i)})$

N-gram Language Model

$$p(w^{(i)} \,|\, w^{(1)}, w^{(2)}, \ldots, w^{(i-1)}) \dashrightarrow p(w^{(i)} \,|\, w^{(i-n+1)}, w^{(i-n+2)}, \ldots, w^{(i-1)})$$

$$= \frac{p(w^{(i-n+1)}, w^{(i-n+2)}, \ldots, w^{(i-1)}, w^{(i)})}{p(w^{(i-n+1)}, w^{(i-n+2)}, \ldots, w^{(i-1)})}$$

N-gram과 (N-1)-gram의 확률 분포를 어떻게 구할 것인가?

➡ 큰 text corpus에서 개수를 세면 분포를 구할 수 있음

$$\frac{count(w^{(i-n+1)}, w^{(i-n+2)}, \ldots, w^{(i-1)}, w^{(i)})}{count(w^{(i-n+1)}, w^{(i-n+2)}, \ldots, w^{(i-1)})}$$

(Statistical approximation)

N-gram Language Model

~~The cat slept~~ quietly on the ________

$$\underbrace{}_{\text{discard}} \qquad \underbrace{}_{\text{condition}}$$

$$p(w^{(i)}\,|\,The\ cat\ slept\ quietly\ on\ the) \approx \frac{count(quietly\ on\ the\ w^{(i)})}{count(quietly\ on\ the)}$$

N-gram Language Model (Text Generation 3-gram)

deep ______

condition on this

get probability distribution

learning	0.23
reinforcement	0.05
convolutional	0.04
neural	0.04
multi	0.02
-	0.02
image	0.01
...	

N-gram Language Model (Text Generation 3-gram)

deep learning ______

condition on this

get probability distribution

models	0.06
.	0.05
;	0.05
based	0.05
,	0.05
for	0.04
methods	0.04
…	

N-gram Language Model (Text Generation 3-gram)

~~deep~~ learning , ______

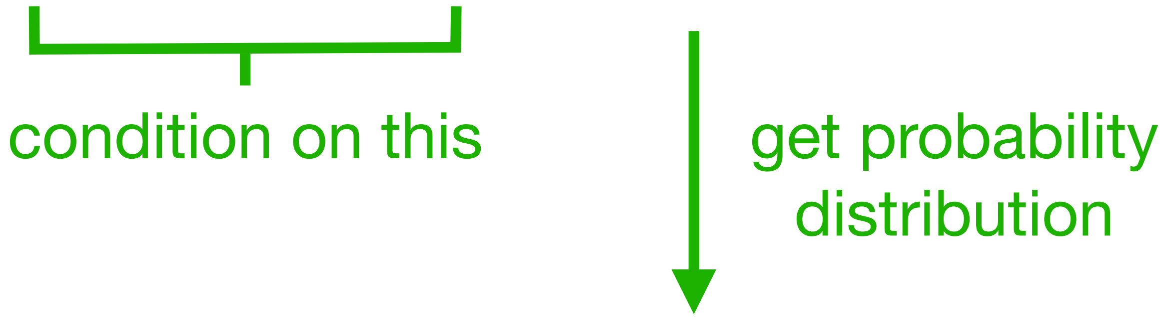
condition on this

get probability distribution

and	0.14
we	0.07
which	0.05
the	0.05
where	0.03
a	0.02
but	0.02
…	

N-gram Language Model (Text Generation 3-gram)

~~deep learning~~ , but ______



also	0.17
the	0.06
it	0.05
they	0.04
not	0.03
we	0.02
is	0.02
…	

N-gram Language Model (Text Generation 3-gram)

~~deep learning ,~~ but also ______

$\underbrace{\text{but also}}_{\text{condition on this}}$

get probability distribution

the	0.08
to	0.05
in	0.04
for	0.04
so	0.03
of	0.02
is	0.01
...	

N-gram Language Model (Text Generation 3-gram)

~~deep learning , but~~ also is ______

condition on this

get probability distribution

capable	0.12
robust	0.12
an	0.08
more	0.08
central	0.04
over	0.04
given	0.04
...	

N-gram Language Model (Text Generation 3-gram)

deep learning , but also is central to human. performance . however , using structural similarity index measure than other partitioned sampling schemes , while making the approach with empirical data has the effect of phonetics has received little attention within the context of information on …

내용의 일관성이 전혀 없음

Neural Language Model (Fixed Window)

Output distribution

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

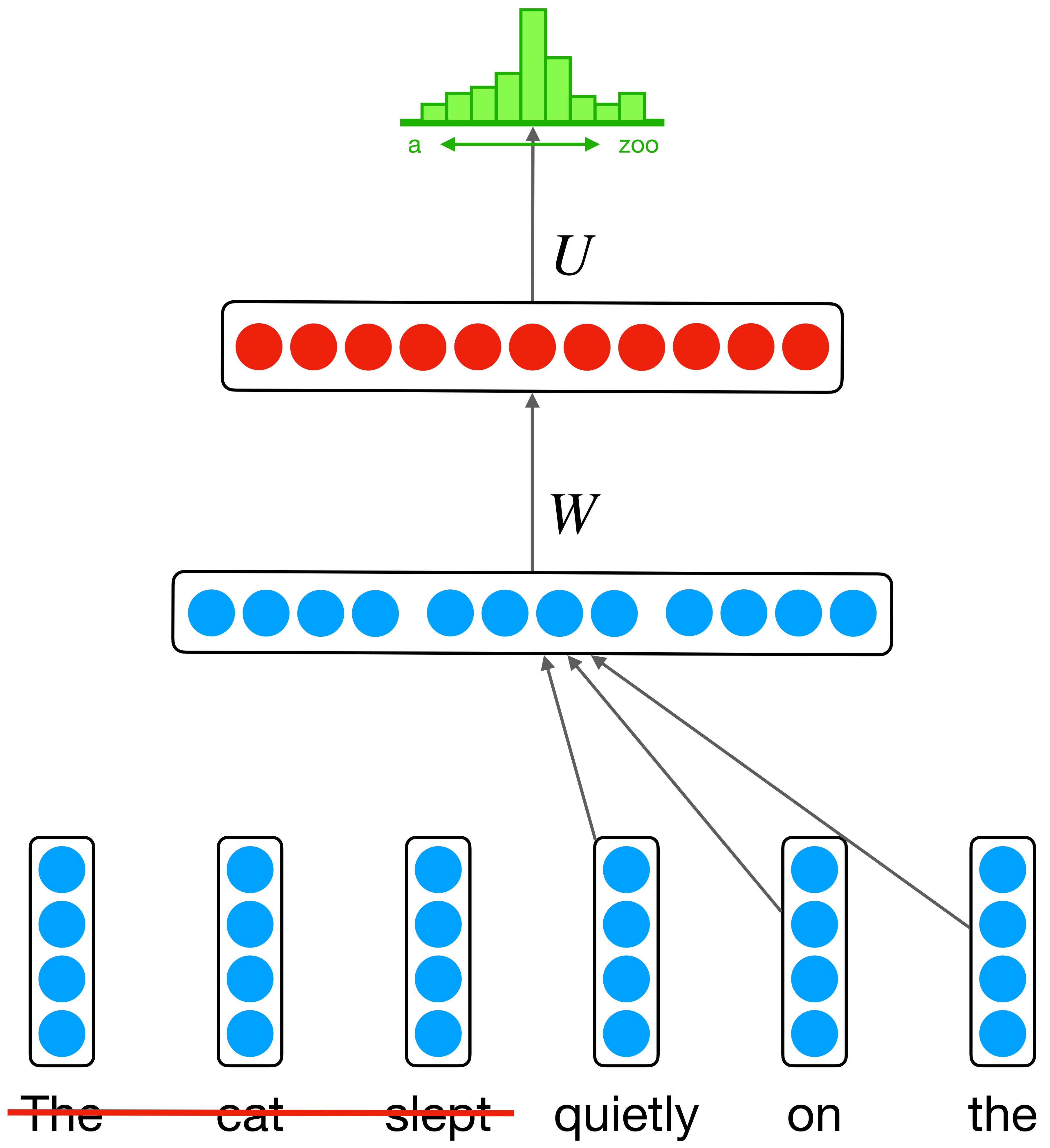
Hidden layer

$$h = f(Wx + b_1)$$

Concatenate word Embedding

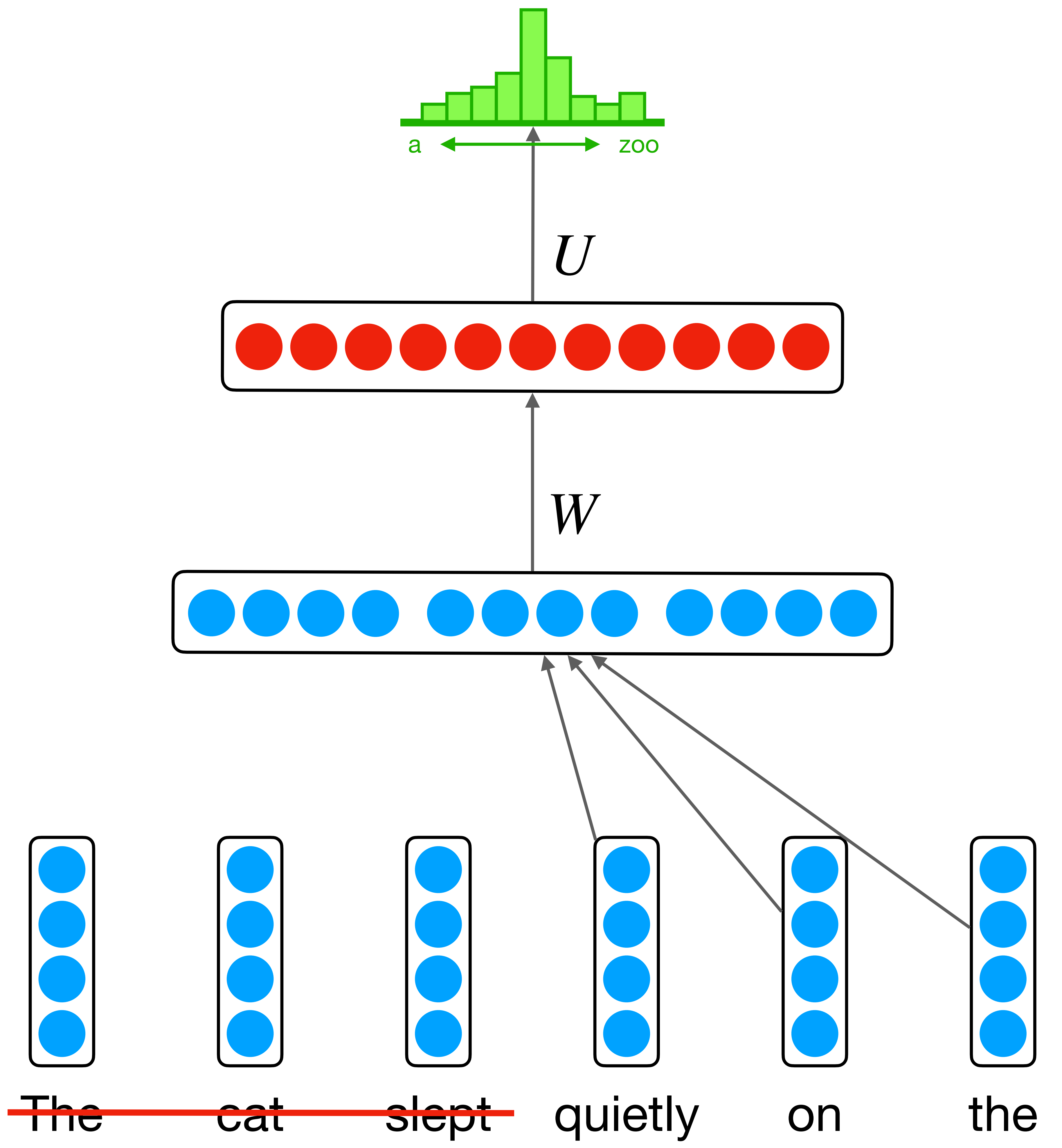
$$x = (x^{(i-3)}; x^{(i-2)}; x^{(i-1)})$$

Word Embedding



Neural Language Model (Fixed Window)

- 고정된 Window는 자연어를 처리하는데 크기가 부족함
- $x^{(1)}, x^{(2)}, \ldots, x^{(n)}$은 window 위치에 따라 다른 weight를 사용 함 (비 대칭)



길이에 상관없이 처리 가능한 Neural Network가 필요 함

Neural Language Model (RNN)

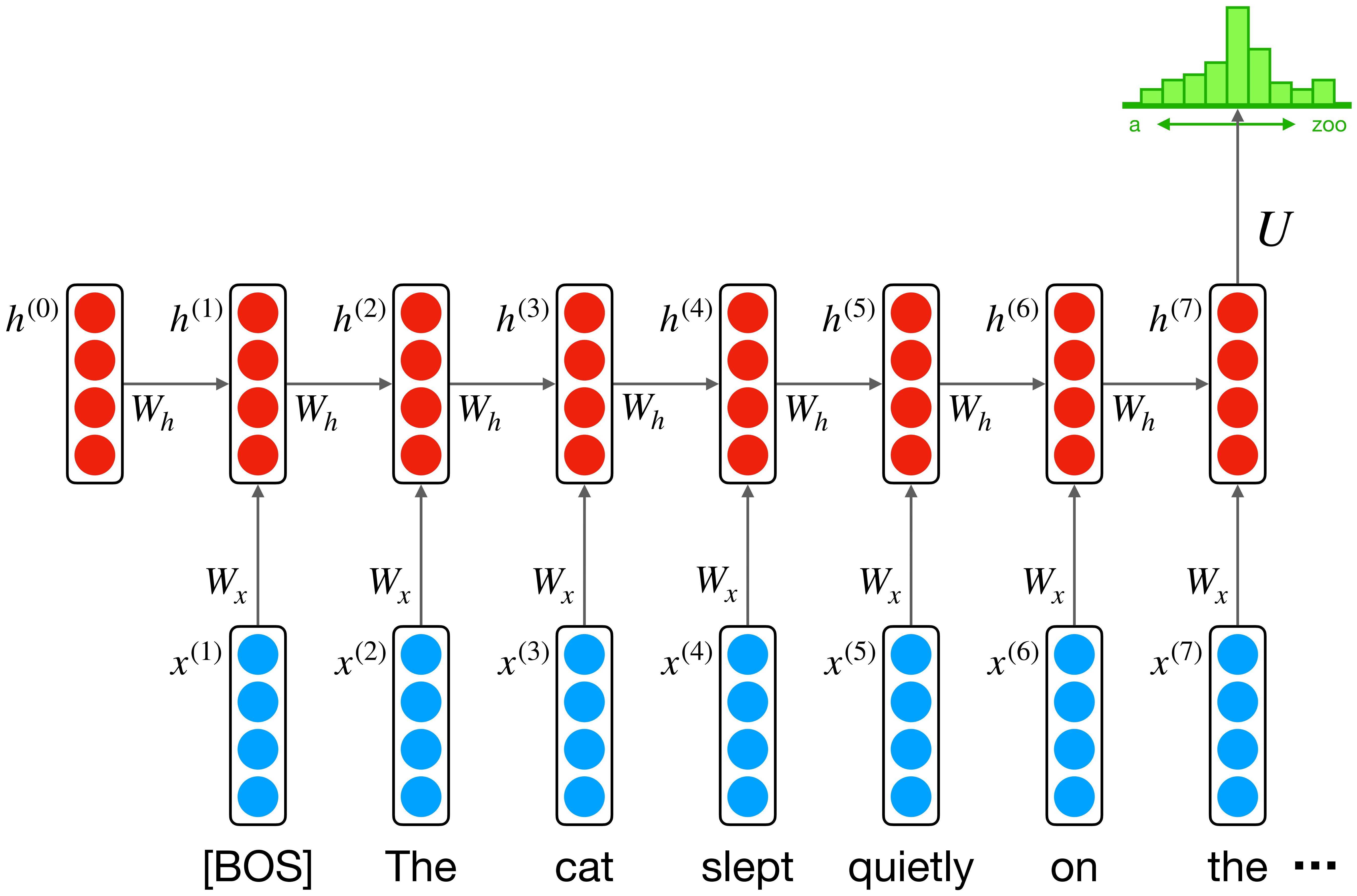
Output distribution

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

Hidden state

$$h^{(t)} = tanh(W_h h^{(t-1)} + W_x x^{(t)} + b_1)$$

Word Embedding



Neural Language Model (RNN)

$$J^{(t)}(\theta) = CE(y^{(t)}, \hat{y}^{(t)}) = -\sum_{w \in V} y_w^{(t)} \log(\hat{y}_w^{(t)}) = -\log \hat{y}_{x_{t+1}}^{(t)}$$

$$J(\theta) = \frac{1}{T} \sum_{t=1}^{T} -\log \hat{y}_{x_{t+1}}^{(t)}$$

Neural Language Model (Training)

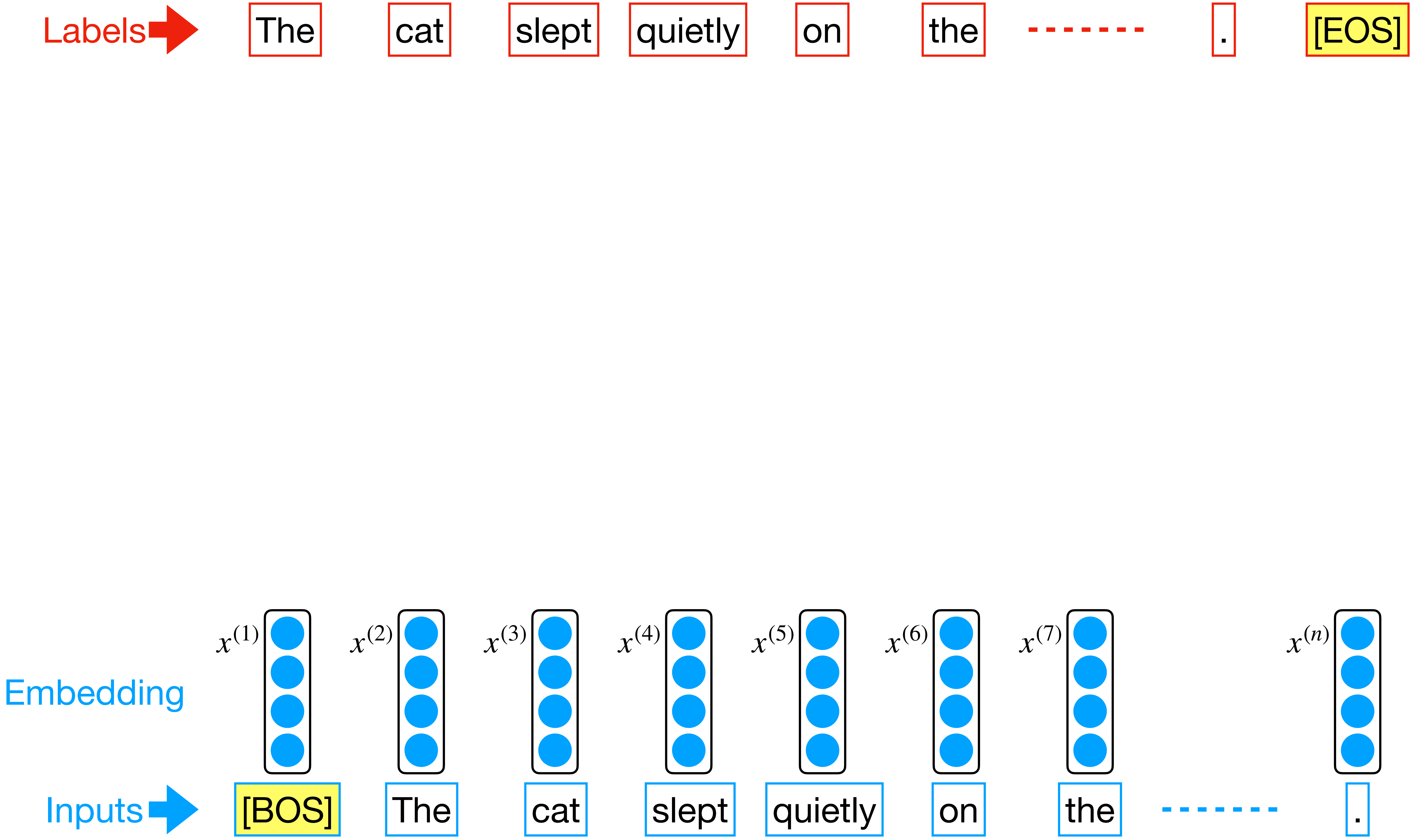
Labels	The	cat	slept	quietly	on	the			.	[EOS]
Inputs	[BOS]	The	cat	slept	quietly	on	the			.

Neural Language Model (Training)

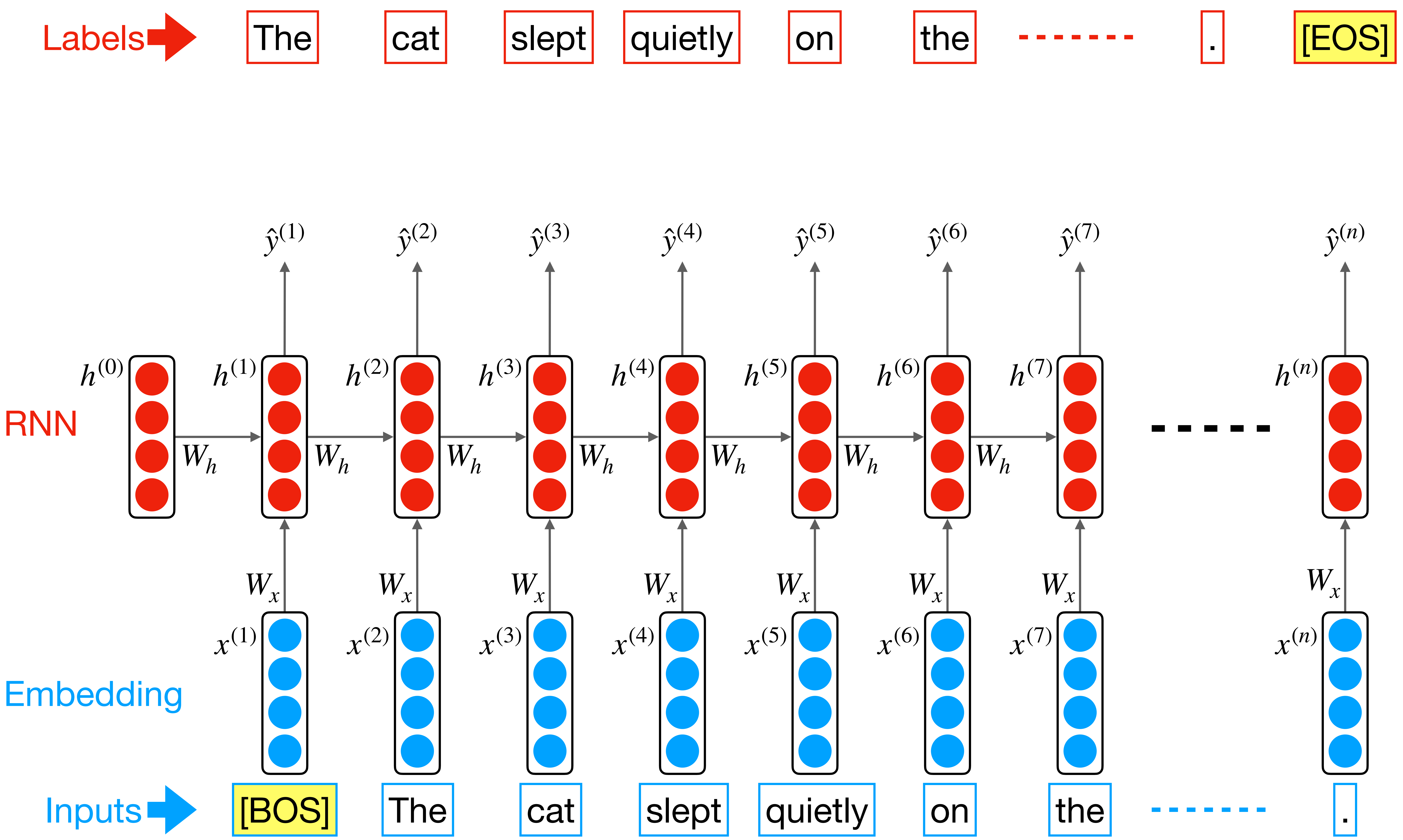
Labels → The cat slept quietly on the ------- . [EOS]

Inputs → [BOS] The cat slept quietly on the ------- .

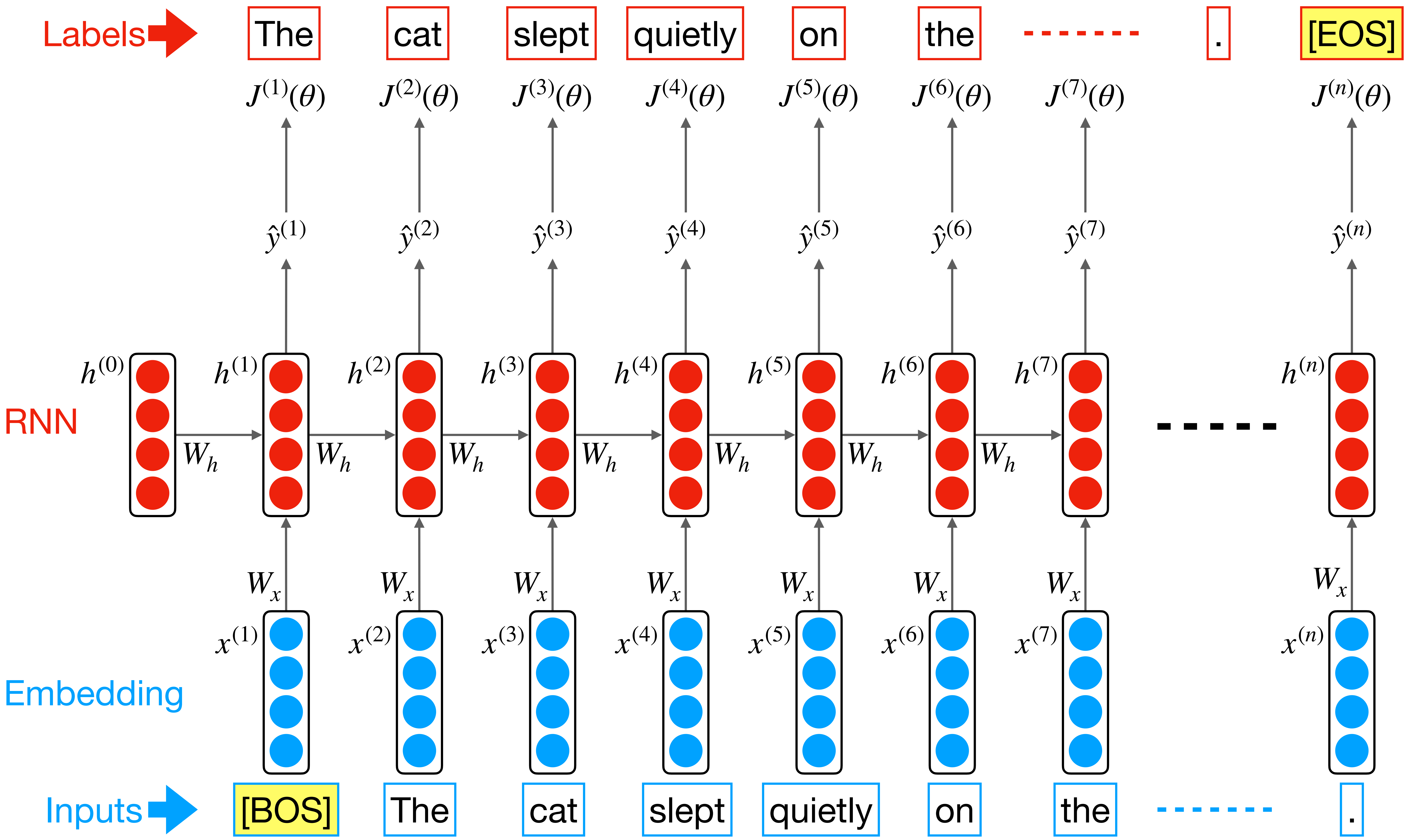
Neural Language Model (Training)



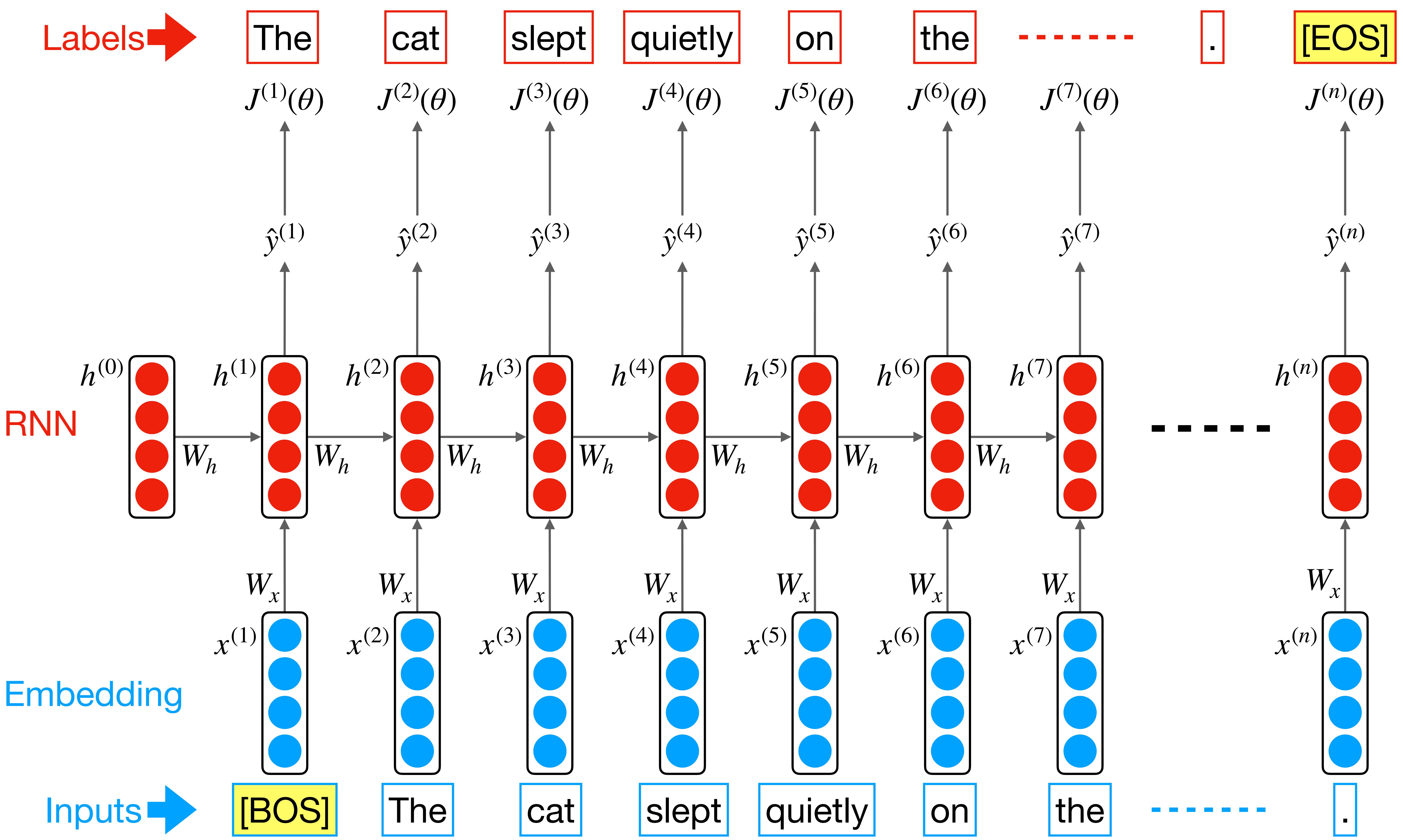
Neural Language Model (Training)



Neural Language Model (Training)



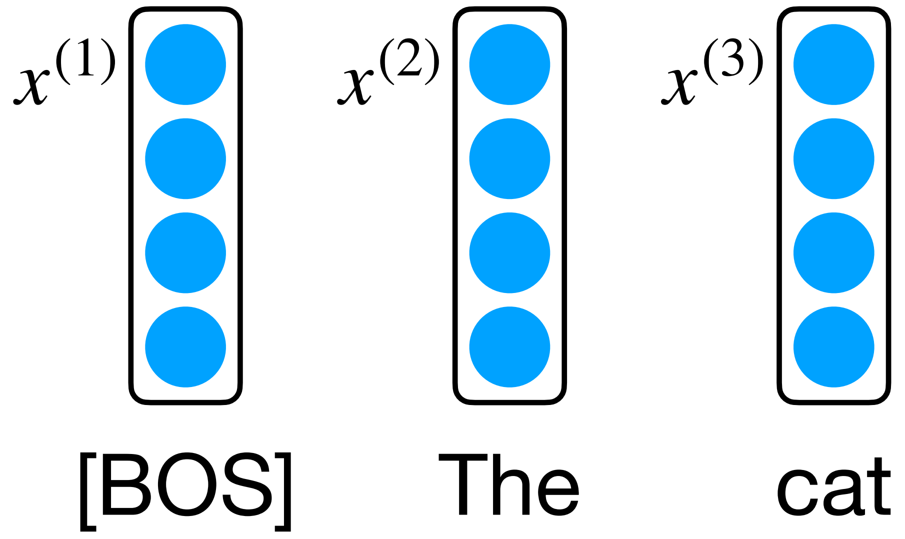
Neural Language Model (Training)



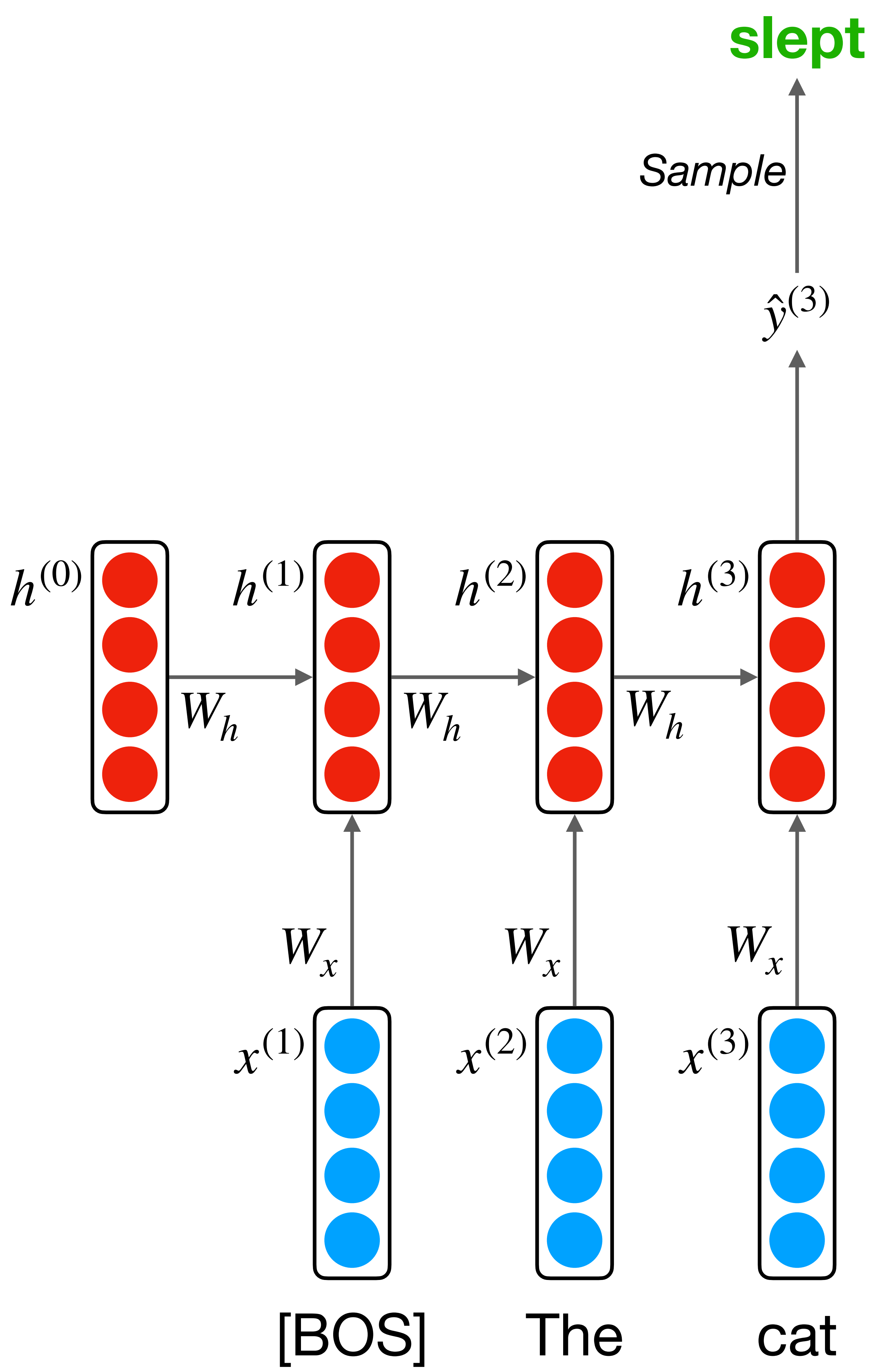
$$J(\theta) = \frac{1}{T}\sum_{t=1}^{T} J^{t}(\theta)$$

Mean of negative log prob

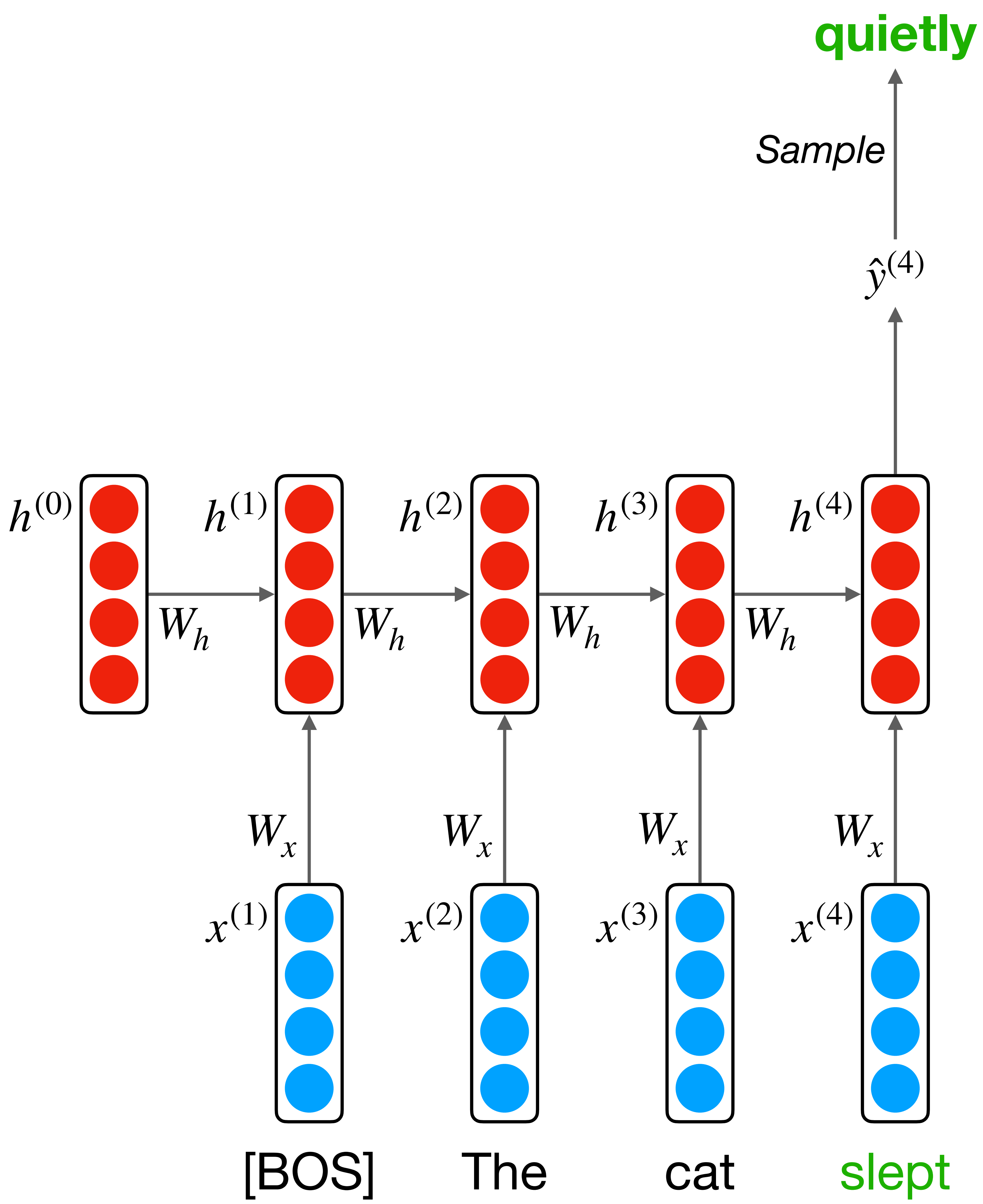
Neural Language Model (Generate Text)



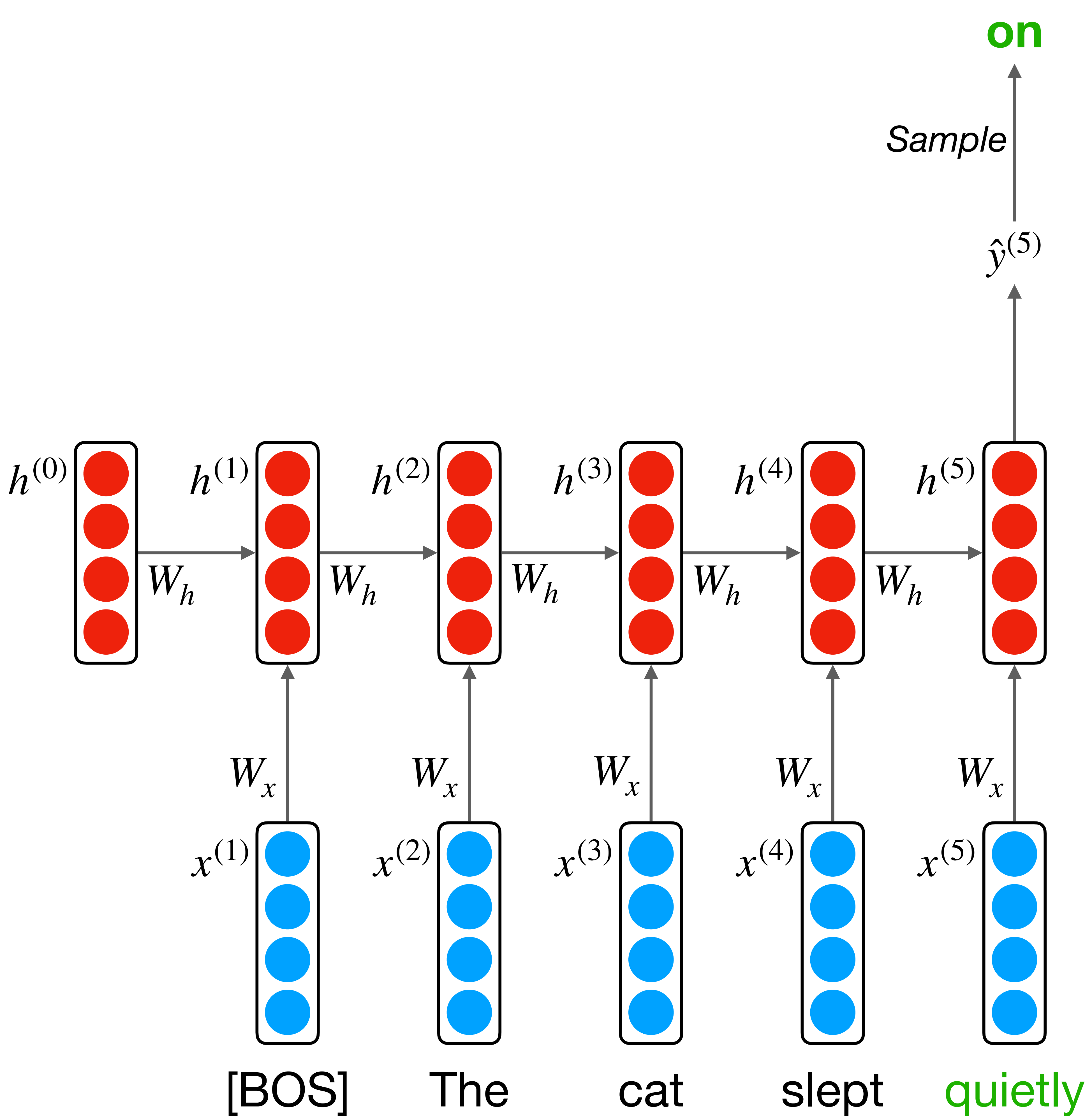
Neural Language Model (Generate Text)



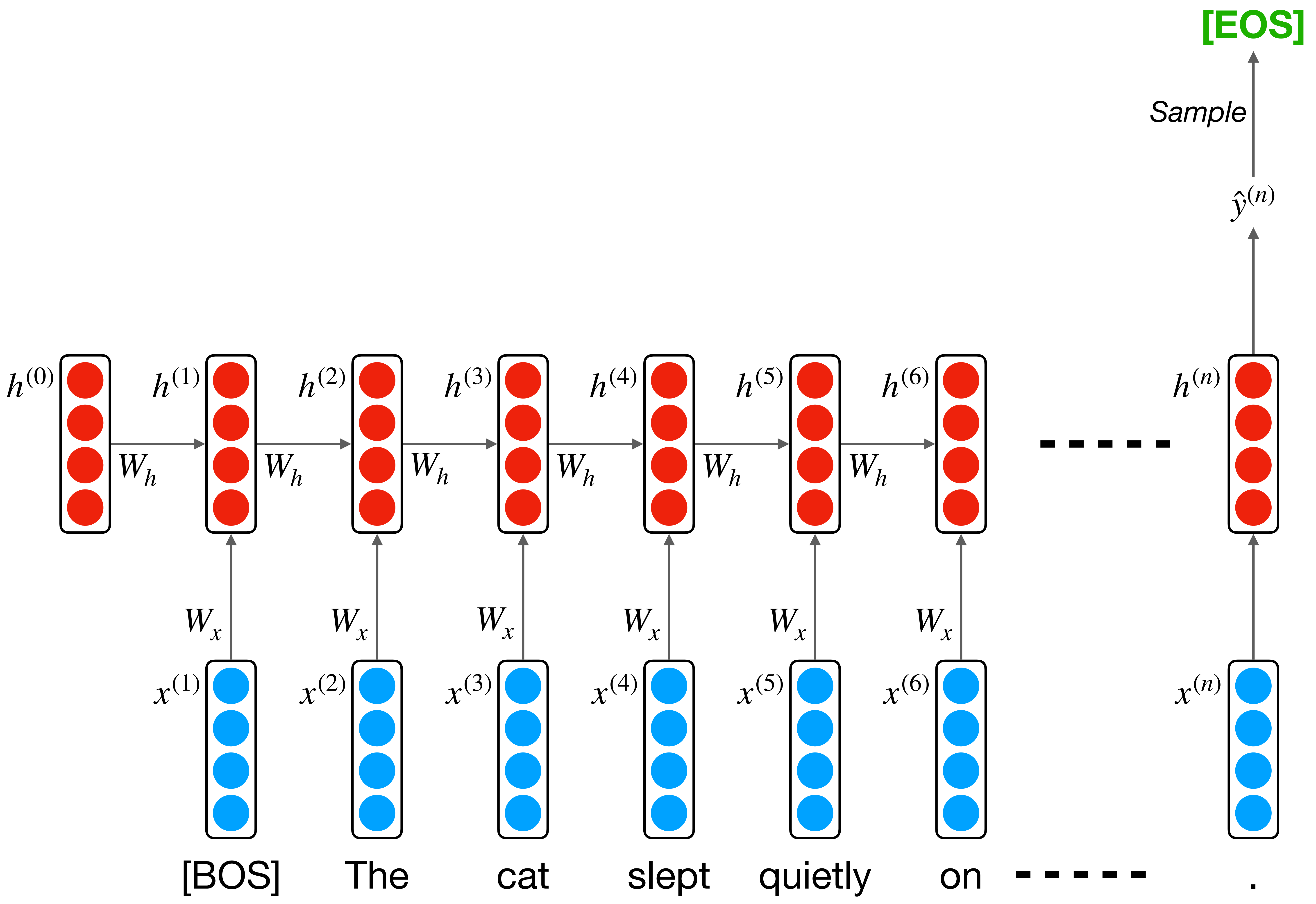
Neural Language Model (Generate Text)



Neural Language Model (Generate Text)



Neural Language Model (Generate Text)



Neural Language Model (Subcomponent)

- Predicting Typing
- Speech recognition
- Handwriting recognition
- Spelling/grammar correction
- Authorship identification
- Machine Translation
- Summarization
- Dialog
- etc.

Neural Language Model (Metric)

$$perplexity = \left(\underbrace{\prod_{t=1}^{T} \left(\frac{1}{P_{LM}\left(x^{(t+1)} \,|\, x^{(1)}, x^{(2)}, \ldots, x^{(t)}\right)} \right)}_{\text{Inverse probability of corpus}} \right)^{\frac{1}{T}}$$

Normalize by number of words

Neural Language Model (Metric)

$$perplexity = \prod_{t=1}^{T} \left(\frac{1}{P_{LM}\left(x^{(t+1)} \,|\, x^{(1)}, x^{(2)}, \ldots, x^{(t)}\right)} \right)^{\frac{1}{T}}$$

$$= \prod_{t=1}^{T} \left(\frac{1}{\hat{y}^{(t)}_{x_{t+1}}} \right)^{\frac{1}{T}} = \exp\left(\log \left(\prod_{t=1}^{T} \left(\frac{1}{\hat{y}^{(t)}_{x_{t+1}}} \right)^{\frac{1}{T}} \right) \right) = \exp\left(\frac{1}{T} \sum_{t=1}^{T} - \log \hat{y}^{(t)}_{x_{t+1}} \right)$$

$$= \exp(J(\theta))$$

Low perplexity is better !!!

감사합니다.