자연어 처리 및 응용

The 13th KIAS CAC Summer School on
the Parallel Computing and Artificial Intelligence

한양대학교 컴퓨터소프트웨어학부
김은솔

≡ 매일경제 뉴스 | 경제 기업 사회 국제 부동산 증권 정치 IT·과학 문화

대기업들이 뛰어드는 '초거대 AI'는 무엇

임영신 기자 | 입력 : 2021.07.08 10:23:10 수정 : 2021.07.08 10:23:26

글로벌 초거대 AI 성능 비교

초거대 AI	개발사	주요 기능	파라미터 개수	발표 시점
RoBERTa	페이스북	언어 생성 · 번역 · 검색 · 기사작성 등	3억5500만	2019년 7월
GPT-2	오픈 AI	언어 생성 · 번역 · 검색 · 기사작성 등	15억	2019년 8월
T5	구글	언어 생성 · 번역 · 검색 · 기사작성 등	110억	2020년 2월
GPT-3	오픈 AI	기존 모든 기능의 고도화 · 프로그래밍	1750억	2020년 6월
하이퍼클로바	네이버	기존 모든 기능의 고도화 · 한국어 문장 생성 탁월	2040억	2021년 5월
우다오 2.0	베이징 지위안 인공지능연구원	기존 모든 기능의 고도화 · 중국어 문장 및 이미지 생성 탁월	1조7500억	2021년 6월
LG 초거대 AI	LG그룹	언어 · 이미지 이해 및 생성 · 데이터 추론	6000억	올 하반기(예정)
GPT-4	오픈AI	GPT-3 초월 전망	100조	2023년 (예정)

동아일보

국내 기업의 주요 초거대 인공지능(AI) 기술 자료: 각 사

	초거대 AI	특징
카카오브레인	코지피티(KoGPT)	▪ 한국어 특화 AI 언어모델 ▪ 구글 텐서 처리장치 활용, 연산속도 고도화
카카오브레인	민달리(minDALL-E)	▪ 1400만 장의 텍스트·이미지 세트 사전 학습 ▪ 텍스트 명령어 입력하면 실시간 이미지 생성
네이버	하이퍼클로바(HyperCLOVA)	▪ 2040억 개에 이르는 매개변수(파라미터) ▪ 학습 데이터의 한글 비중 97%, 한국어 집중 교육
LG	엑사원(EXAONE)	▪ 언어·이미지·영상 등을 다루는 멀티 모달리티 능력 ▪ 제조·연구·교육·금융 분야 상위 1% 전문가 목표

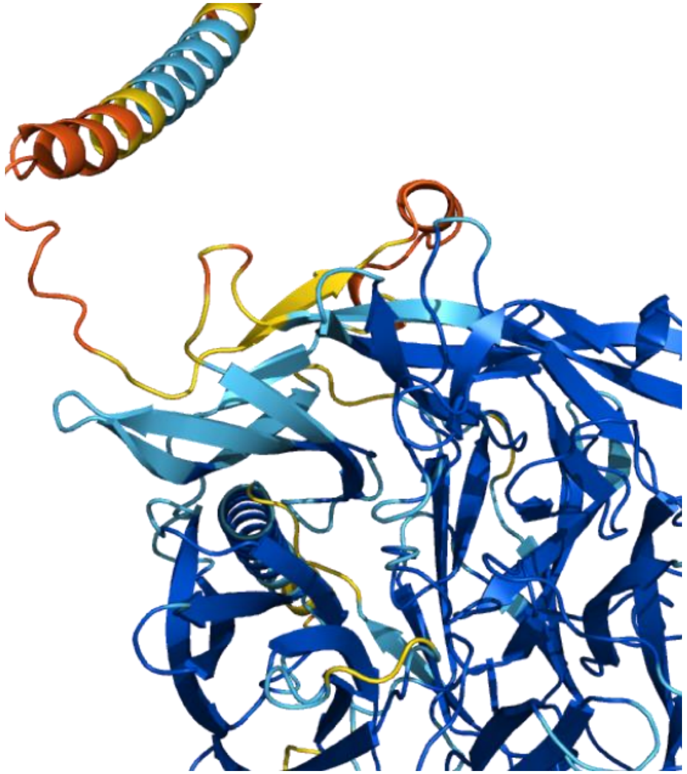
Transformer in Biological Science

Highly accurate protein structure prediction with AlphaFold

https://doi.org/10.1038/s41586-021-03819-2
Received: 11 May 2021
Accepted: 12 July 2021
Published online: 15 July 2021
Open access
Check for updates

John Jumper[1,4✉], Richard Evans[1,4], Alexander Pritzel[1,4], Tim Green[1,4], Michael Figurnov[1,4], Olaf Ronneberger[1,4], Kathryn Tunyasuvunakool[1,4], Russ Bates[1,4], Augustin Žídek[1,4], Anna Potapenko[1,4], Alex Bridgland[1,4], Clemens Meyer[1,4], Simon A. A. Kohl[1,4], Andrew J. Ballard[1,4], Andrew Cowie[1,4], Bernardino Romera-Paredes[1,4], Stanislav Nikolov[1,4], Rishub Jain[1,4], Jonas Adler[1], Trevor Back[1], Stig Petersen[1], David Reiman[1], Ellen Clancy[1], Michal Zielinski[1], Martin Steinegger[2,3], Michalina Pacholska[1], Tamas Berghammer[1], Sebastian Bodenstein[1], David Silver[1], Oriol Vinyals[1], Andrew W. Senior[1], Koray Kavukcuoglu[1], Pushmeet Kohli[1] & Demis Hassabis[1,4✉]

nature



Biological structure and function emerge from scaling unsupervised learning to 250 million protein sequences

Alexander Rives[a,b,1,2], Joshua Meier[a,1], Tom Sercu[a,1], Siddharth Goyal[a,1], Zeming Lin[b], Jason Liu[a], Demi Guo[c,3], Myle Ott[a], C. Lawrence Zitnick[a], Jerry Ma[d,e,3], and Rob Fergus[b]

PNAS

Contents

Word Embedding

Data Representation

Table of baby-name data
(baby-2010.csv)

name	rank	gender	year
Jacob	1	boy	2010
Isabella	1	girl	2010
Ethan	2	boy	2010
Sophia	2	girl	2010
Michael	3	boy	2010

Field names

One row (4 fields)

2000 rows all told

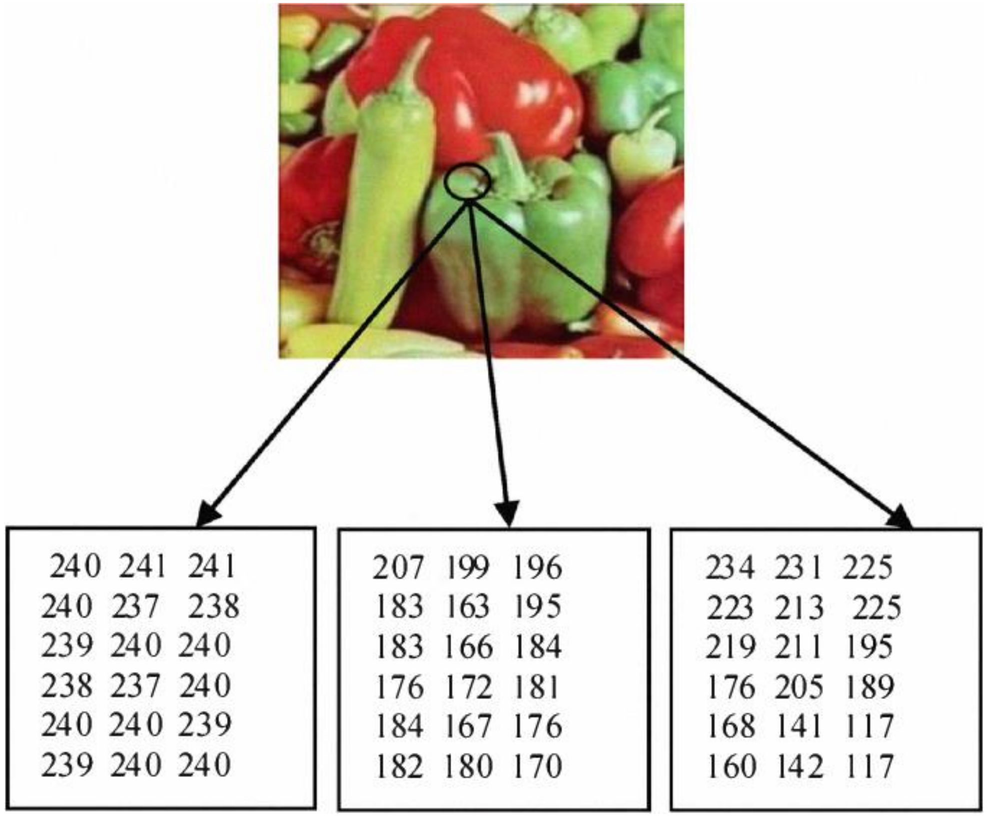


image from “https://web.stanford.edu/class/cs101/table-data.html”

image from “A new approach for LSB based image steganography using secret key”

Data Representation – Text

- Conventional Word Representations
  - 이미지, 음성과 달리 언어 데이터는 discrete
  - One-hot encoding
    - 데이터에 포함된 단어로 사전을 만들고, 이를 기반으로 one-hot encoding을 하여 단어를 표현
    - Discrete, Sparse
- All vectors are orthogonal
  - There are no natural notion of similarity for one-hot vectors

Word	One-hot encoding
economic	000010...
growth	001000...
has	100000...
slowed	000001...

Word Embedding

- Assumption: Distributional semantics (hypothesis)
    - *Linguistic items with similar distributions have similar meanings*
    - Representing words by their context

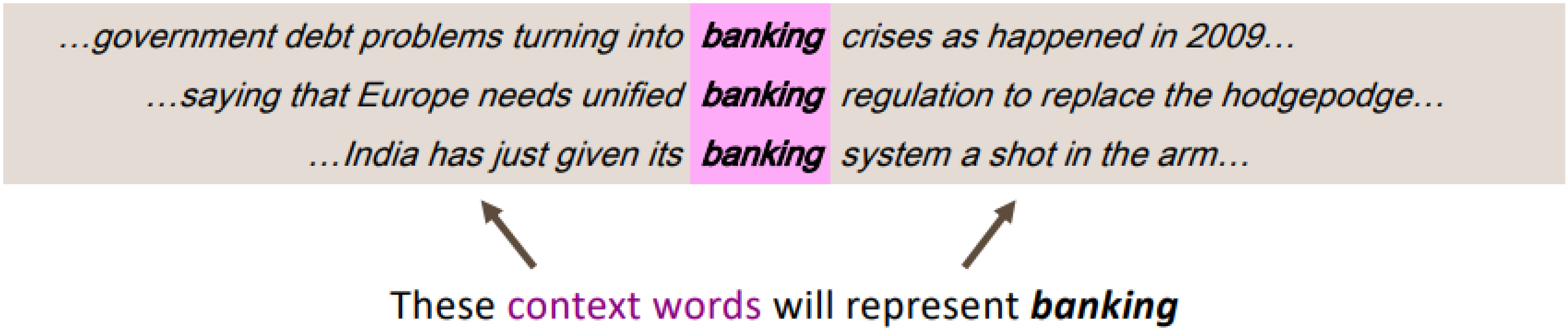


image from https://web.stanford.edu/class/archive/cs/cs224n/cs224n.1214/slides/cs224n-2021-lecture01-wordvecs1.pdf

Word Embedding

- Build a dense vector for each word
- similar to vectors of words that appear in similar contexts

$$\textit{banking} = \begin{pmatrix} 0.286 \\ 0.792 \\ -0.177 \\ -0.107 \\ 0.109 \\ -0.542 \\ 0.349 \\ 0.271 \end{pmatrix}$$

image from https://web.stanford.edu/class/archive/cs/cs224n/cs224n.1214/slides/cs224n-2021-lecture01-wordvecs1.pdf

Word2Vec

- Efficient Estimation of Word Representations in Vector Space
  - T. Mikolov et al., ICLR Workshop, 2013
- Distributed Representations of Words and Phrases and their Compositionality
  - T. Mikolov et al., 2013, NeurIPS

Distributed Representations of Words and Phrases and their Compositionality

Tomas Mikolov
Google Inc.
Mountain View
mikolov@google.com

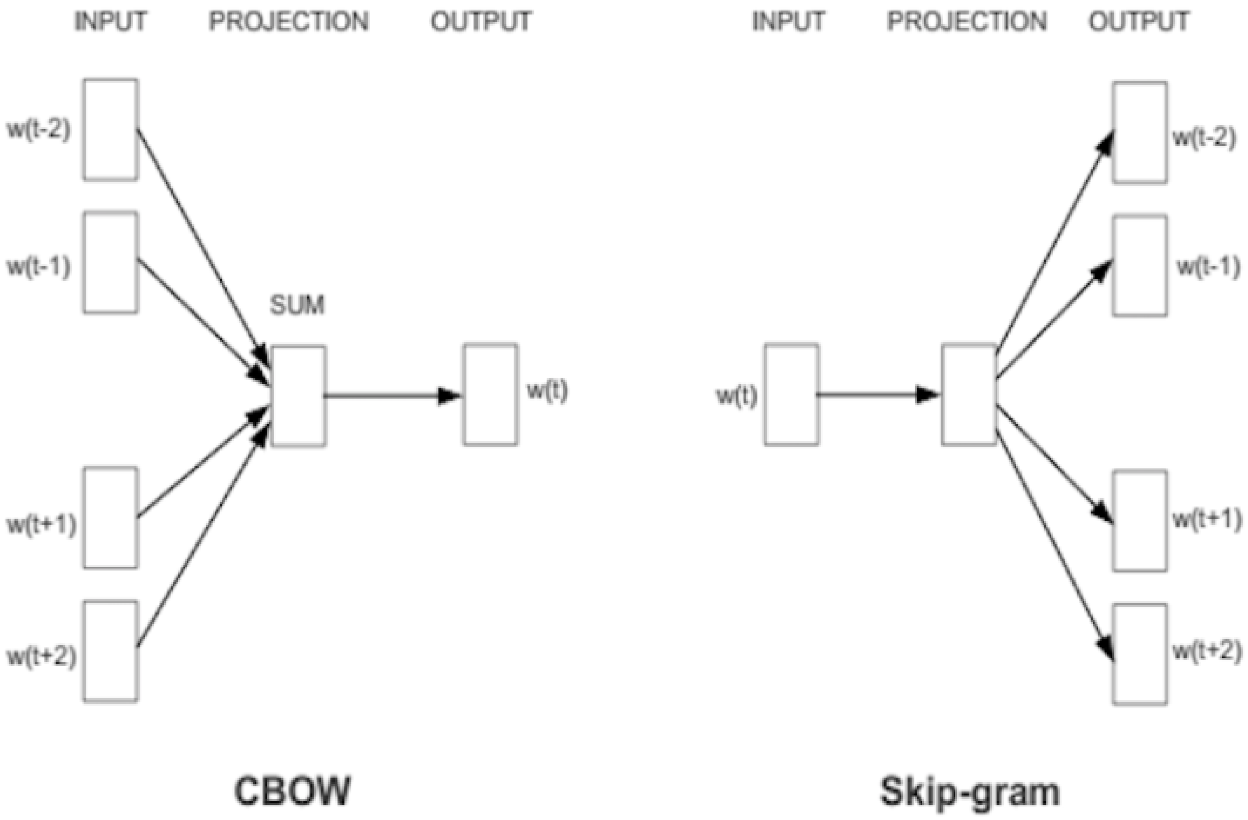
Ilya Sutskever
Google Inc.
Mountain View
ilyasu@google.com

Kai Chen
Google Inc.
Mountain View
kai@google.com

Greg Corrado
Google Inc.
Mountain View
gcorrado@google.com

Jeffrey Dean
Google Inc.
Mountain View
jeff@google.com

Word2Vec



- Key Idea
  - Find word representations that are useful for predicting the surrounding words
  - Use the similarity of the word vectors to calculate the probability

Word2Vec

	w_{t-2}	w_{t-1}	w_t	w_{t+1}	w_{t+2}	
The	**training**	**objective**	**of**	**the**	**Skip-gram**	**model**
	context words		center word	context words		

Word2Vec

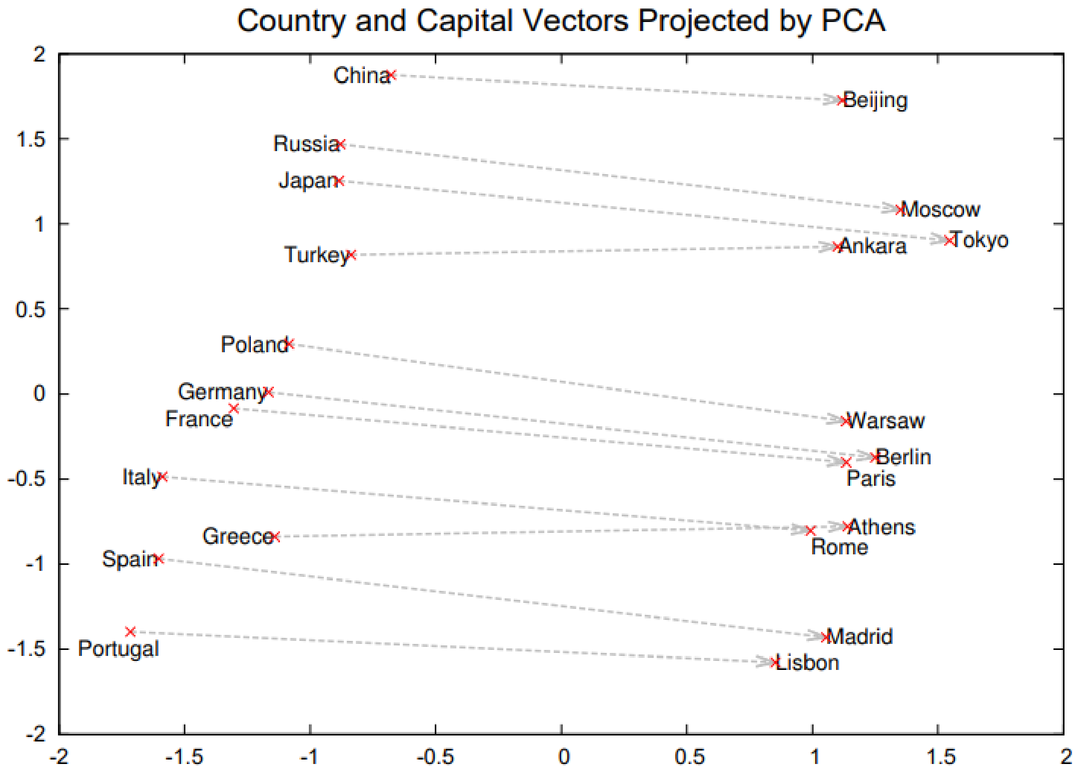
	w_{t-2}	w_{t-1}	w_t	w_{t+1}	w_{t+2}	
The	**training**	**objective**	**of**	**the**	**Skip-gram**	**model**
	context words		center word	context words		

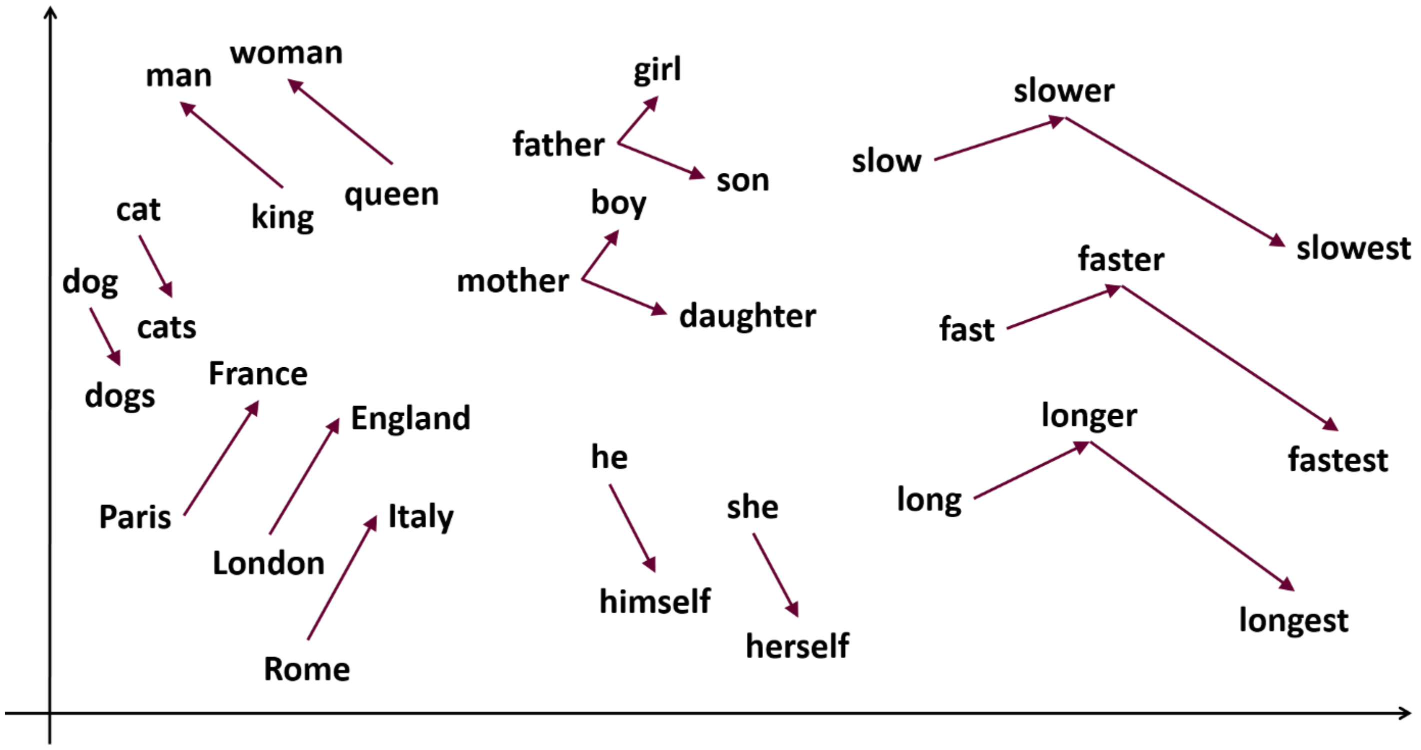
training objective function

$$\frac{1}{T}\sum_{t=1}^{T}\sum_{-c\leq j\leq c, j\neq 0}\log p(w_{t+j}|w_t)$$

$$p(w_O|w_I) = \frac{\exp\left({v'_{w_O}}^{\top} v_{w_I}\right)}{\sum_{w=1}^{W}\exp\left({v'_w}^{\top} v_{w_I}\right)}$$

Interesting Results of the Word2Vec





https://medium.com/analytics-vidhya/implementing-word2vec-in-tensorflow-44f93cf2665f

Another Word (Neural) Embedding

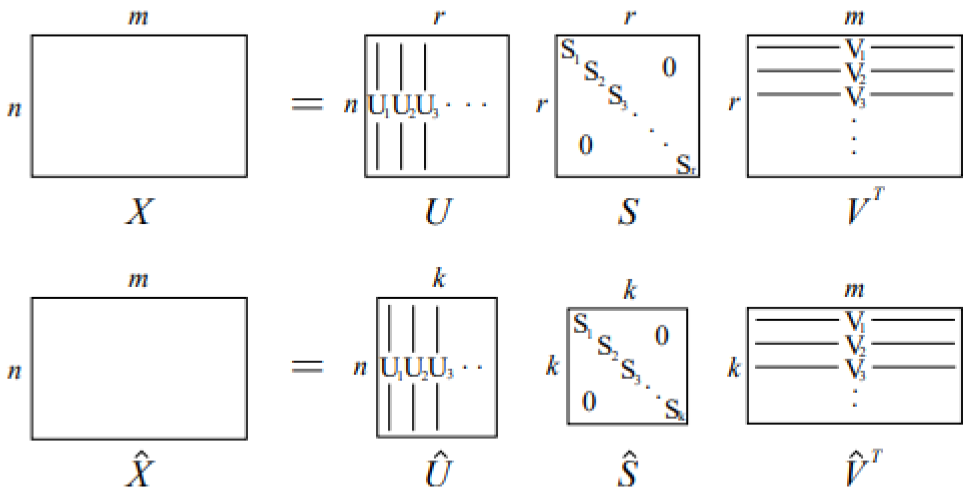
- Using co-occurrence information

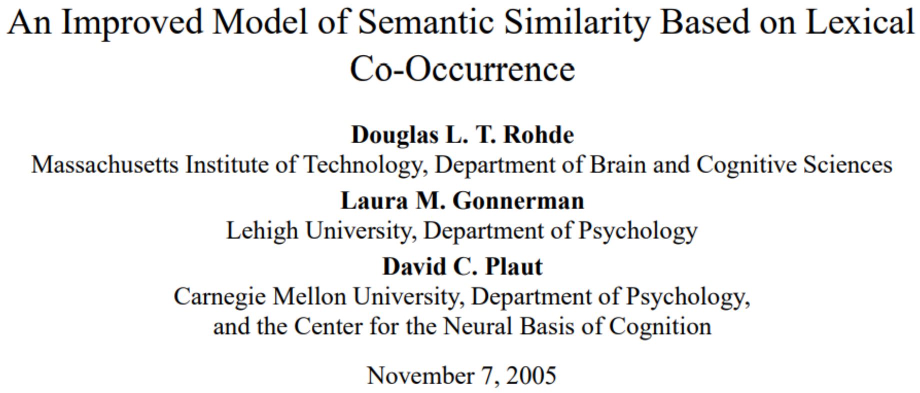
	a	as	chuck	could	how	if	much	wood	woodch.	would	,	.	?	a	as	chuck	could	how	if	much	wood	woodch.	would	,	.	?
a	13	24	12	3	9	20	22	31	16	23	18	0	7	13	7	31	26	0	14	4	21	50	9	16	7	7
as	7	8	15	11	0	5	9	25	10	0	3	0	17	24	8	2	3	0	9	10	10	20	13	11	0	0
chuck	31	2	5	20	5	14	6	9	36	15	12	0	0	12	15	5	6	0	9	8	30	10	2	11	9	12
could	26	3	6	0	0	16	2	4	30	9	14	0	0	3	11	20	0	0	0	6	23	2	1	0	8	8
how	0	0	0	0	0	0	0	0	0	0	0	0	0	9	0	5	0	0	3	10	9	7	8	4	0	0
if	14	9	9	0	3	0	8	11	16	15	20	0	2	20	5	14	16	0	0	3	14	18	0	0	5	5
much	4	10	8	6	10	3	0	8	5	0	2	0	9	22	9	6	2	0	8	0	20	18	15	10	0	0
wood	21	10	30	23	9	14	20	7	26	5	11	0	8	31	25	9	4	0	11	8	7	26	20	14	10	10
woodch.	50	20	10	2	7	18	18	26	13	20	16	0	5	16	10	36	30	0	16	5	26	13	10	18	9	9
would	9	13	2	1	8	0	15	20	10	0	0	0	4	23	0	15	9	0	15	0	5	20	0	17	3	0
,	16	11	11	0	4	0	10	14	18	17	0	0	3	18	3	12	14	0	20	2	11	16	0	0	4	4
.	7	0	9	8	0	5	0	10	9	3	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
?	7	0	12	8	0	5	0	10	9	0	4	0	0	7	17	0	0	0	2	9	8	5	4	3	0	0

Rohde et al., 2005

Co-occurrence based word vectors

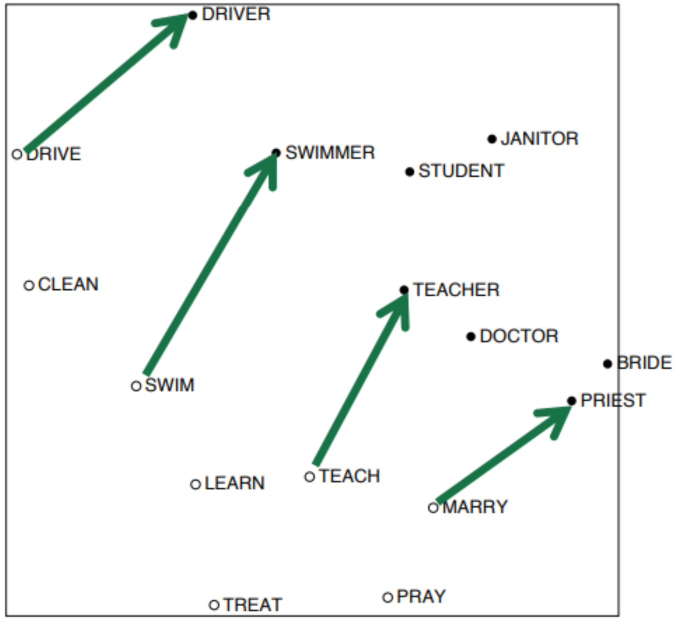
- Singular Value Decomposition of co-occurrence matrix X
- Factorize X into $U\Sigma V^{\top}$
  - U, V are orthogonal




An Improved Model of Semantic Similarity Based on Lexical Co-Occurrence

Douglas L. T. Rohde
Massachusetts Institute of Technology, Department of Brain and Cognitive Sciences
Laura M. Gonnerman
Lehigh University, Department of Psychology
David C. Plaut
Carnegie Mellon University, Department of Psychology, and the Center for the Neural Basis of Cognition

November 7, 2005



GloVe

GloVe: Global Vectors for Word Representation

Jeffrey Pennington, Richard Socher, Christopher D. Manning
Computer Science Department, Stanford University, Stanford, CA 94305
jpennin@stanford.edu, richard@socher.org, manning@stanford.edu

- Key idea
  - Ratios of co-occurrence probabilities can encode meaning components

	x = solid	x = gas	x = water	x = random
$P(x\|\text{ice})$	large	small	large	small
$P(x\|\text{steam})$	small	large	large	small
$\frac{P(x\|\text{ice})}{P(x\|\text{steam})}$	large	small	~1	~1

https://web.stanford.edu/class/archive/cs/cs224n/cs224n.1214/slides/cs224n-2021-lecture02-wordvecs2.pdf

GloVe

GloVe: Global Vectors for Word Representation

Jeffrey Pennington, Richard Socher, Christopher D. Manning
Computer Science Department, Stanford University, Stanford, CA 94305
jpennin@stanford.edu, richard@socher.org, manning@stanford.edu

- Key idea
  - Ratios of co-occurrence probabilities can encode meaning components

	x = solid	x = gas	x = water	x = fashion
$P(x \mid \text{ice})$	1.9×10^{-4}	6.6×10^{-5}	3.0×10^{-3}	1.7×10^{-5}
$P(x \mid \text{steam})$	2.2×10^{-5}	7.8×10^{-4}	2.2×10^{-3}	1.8×10^{-5}
$\frac{P(x \mid \text{ice})}{P(x \mid \text{steam})}$	8.9	8.5×10^{-2}	1.36	0.96

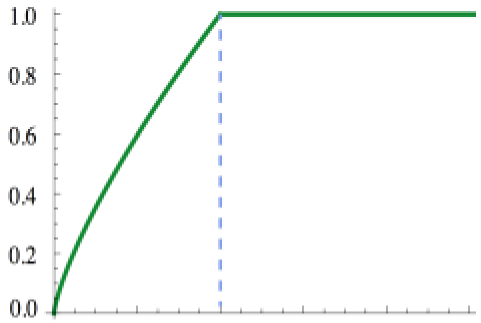
GloVe

- Training Objective

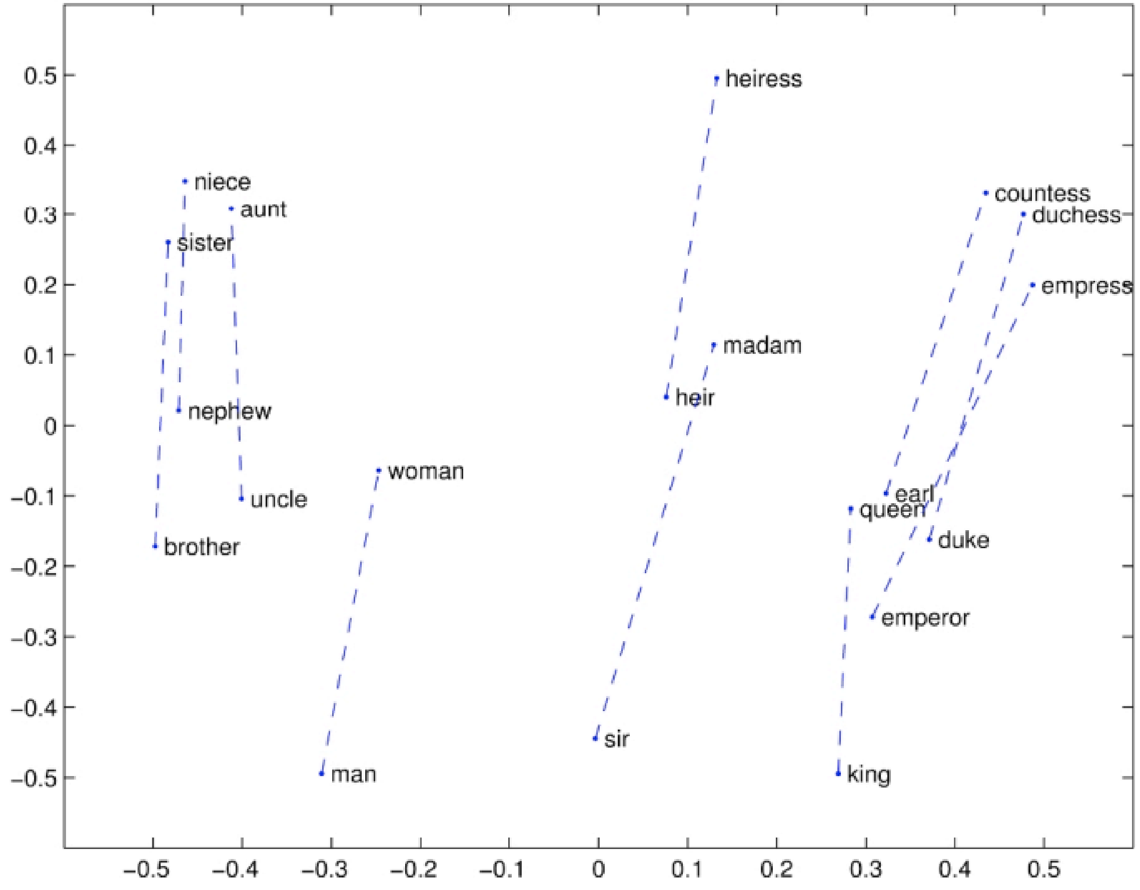
$$w_i \cdot w_j = \log P(i|j)$$

$$J = \sum_{i,j=1}^{V} f\left(X_{ij}\right)\left(w_i^T \tilde{w}_j + b_i + \tilde{b}_j - \log X_{ij}\right)^2$$

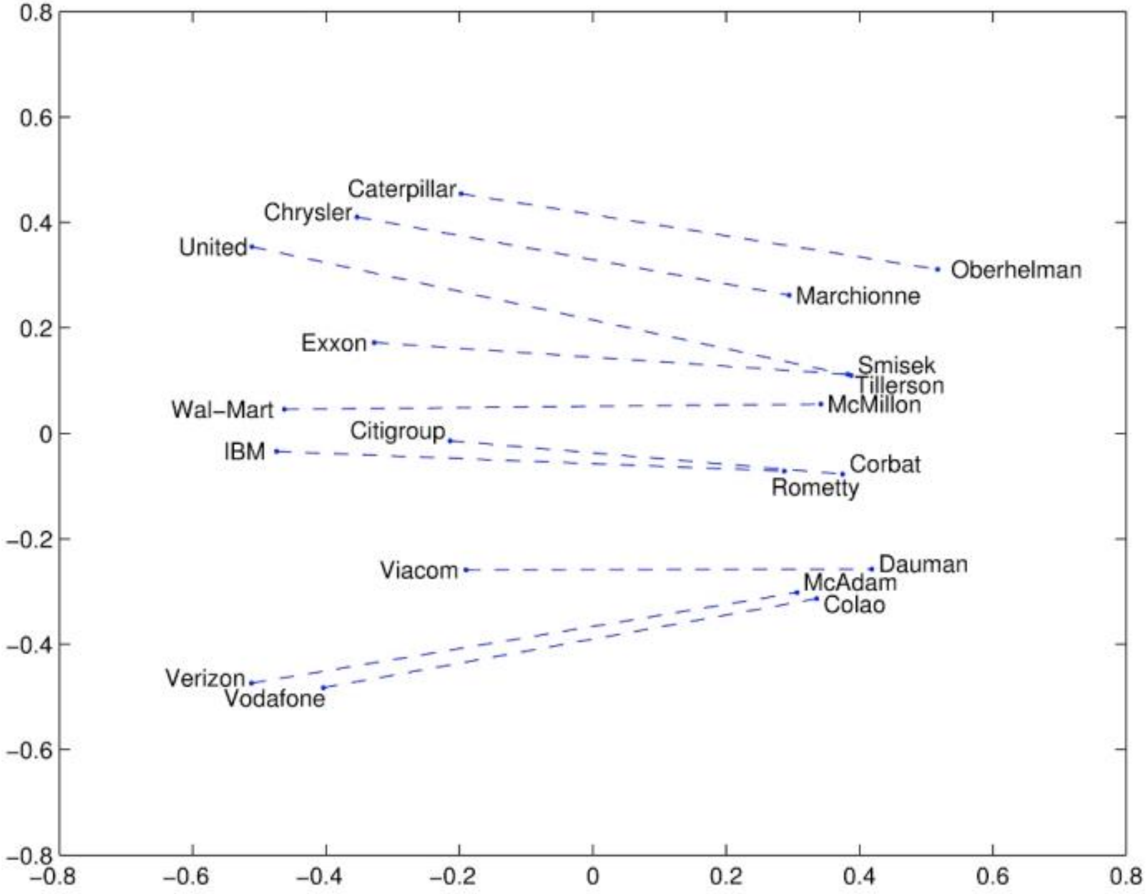
$f \sim$



GloVe



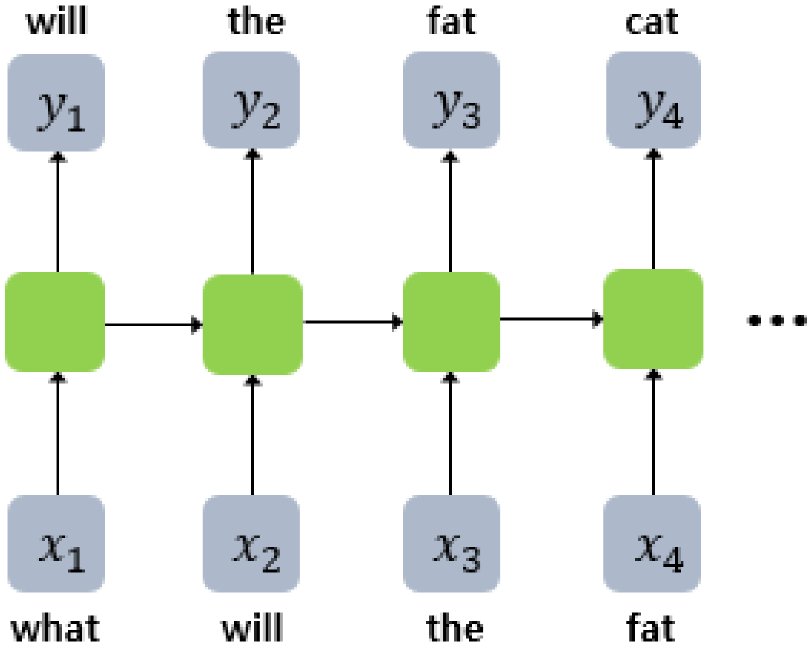
heiress
niece
aunt
sister
countess
duchess
empress
madam
heir
nephew
woman
uncle
earl
queen
brother
duke
emperor
sir
man
king
0.5
0.4
0.3
0.2
0.1
0
−0.1
−0.2
−0.3
−0.4
−0.5

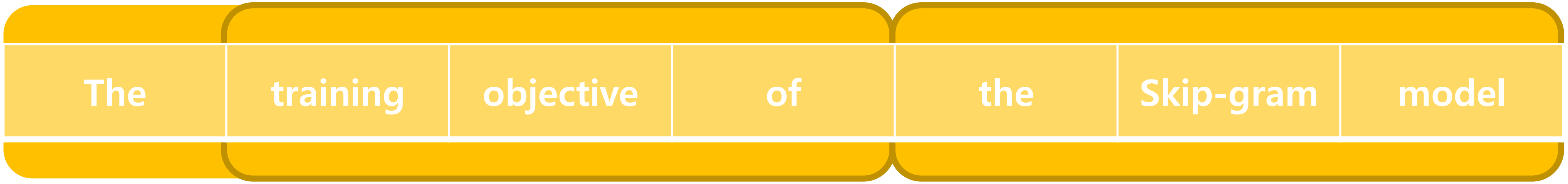


Caterpillar
Chrysler
United
Oberhelman
Marchionne
Exxon
Smisek
Tillerson
Wal–Mart
McMillon
Citigroup
IBM
Corbat
Rometty
Viacom
Dauman
McAdam
Colao
Verizon
Vodafone
0.8
0.6
0.4
0.2
0
−0.2
−0.4
−0.6
−0.8

Language Models

Language Models

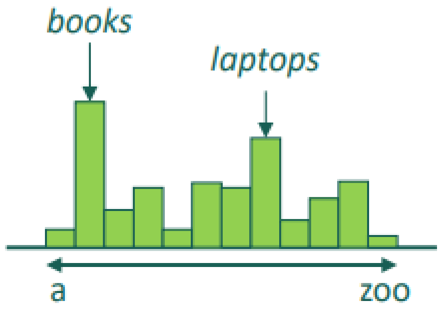




Language Models

- Language modeling is the task of predicting what word comes next.

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



$$V=\{w_1,w_2,\dots,w_{|V|}\}$$

$$P(x^{1},\dots,x^{t})=P(x^{1})\times P(x^{2}|x^{1})\times\cdots\times P(x^{t}|x^{t-1},\dots,x^{1})$$

P(This is a sentence) = P(This) x P(is | This) x P(a | This is) x P(sentence| This is a)

n-gram Language Models

- Modeling with Markov assumption
  - x^t depends only on the preceding (n-1) words

$$P(x^{t+1}|x^t,\ldots,x^1) = P(x^{t+1}|x^t,\ldots,x^{t-n+2})$$

- For example, if n=3

P(This is a sentence from AAA) = P(This)

x P(is | This)
x P(a | This is)
x P(sentence| ~~This~~ is a)
x P(from | ~~This is~~ a sentence)
x P(AAA | ~~This is a~~ sentence from)

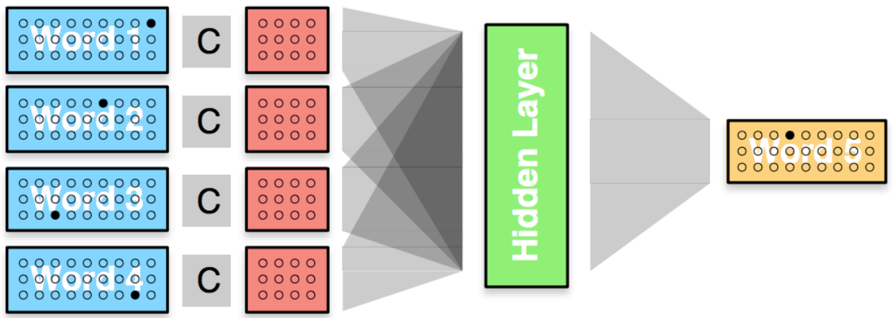
calculate by counting the phrases in large corpus of text

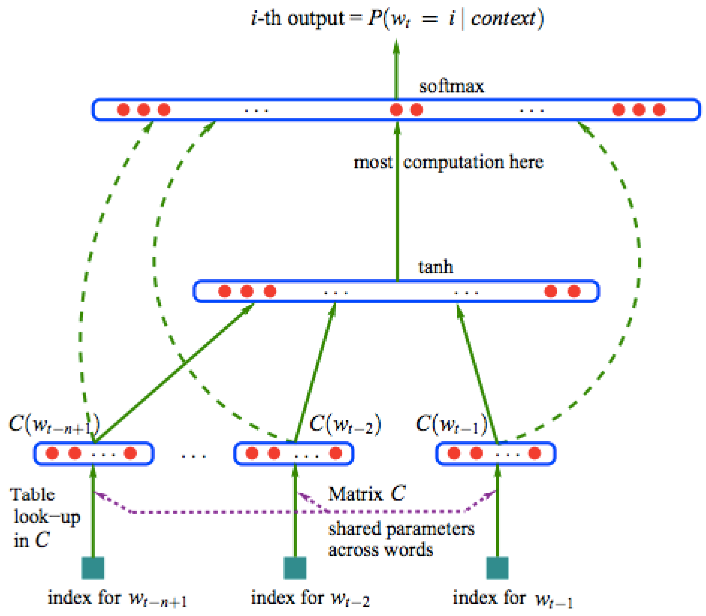
Neural Word Embedding

- Neural network의 hidden vector 값을 이용하여 단어의 의미를 표현하는 기법
- Neural network language model (NNLM, 2003)

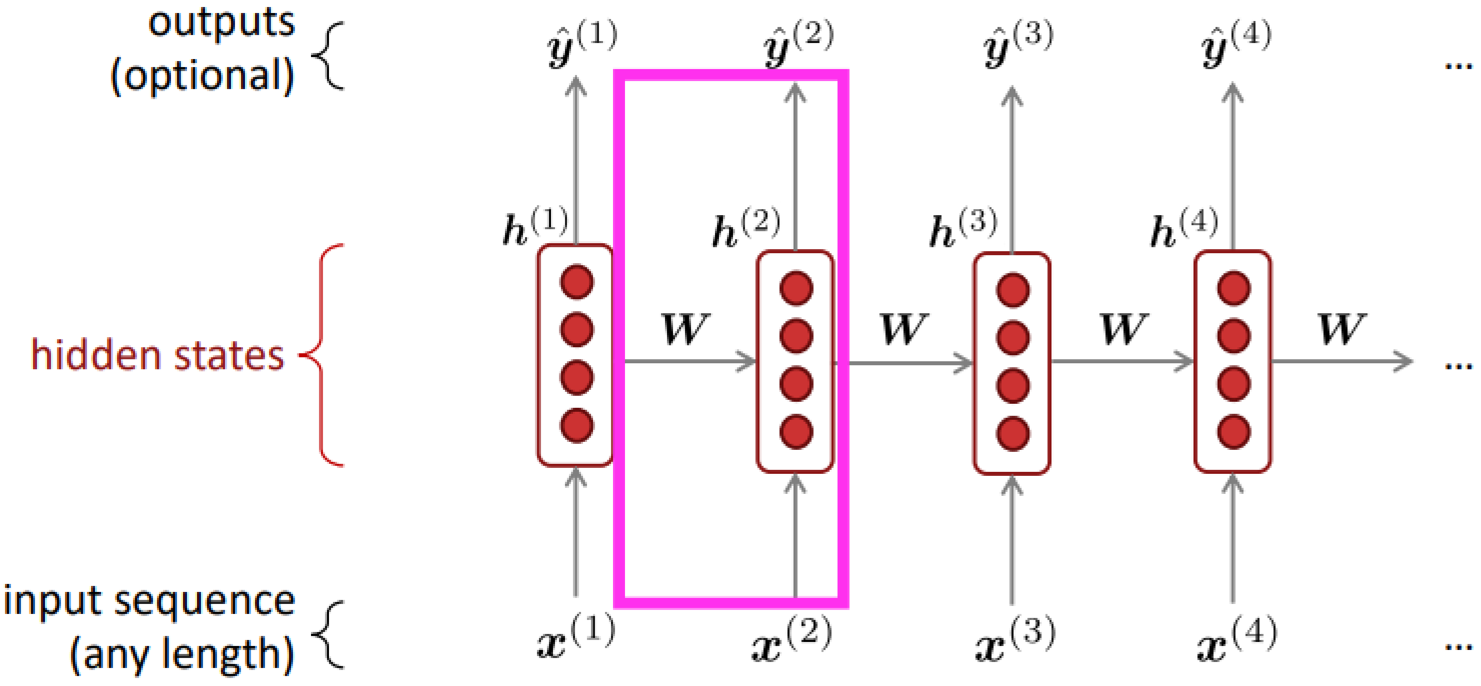
$$p(W) = \sum_i p(w_i|w_1, \ldots, w_{i-1})$$

$$p(w_i|w_1, \ldots, w_{i-1}) \simeq p(w_i|w_{i-4}, w_{i-3}, w_{i-2}, w_{i-1})$$

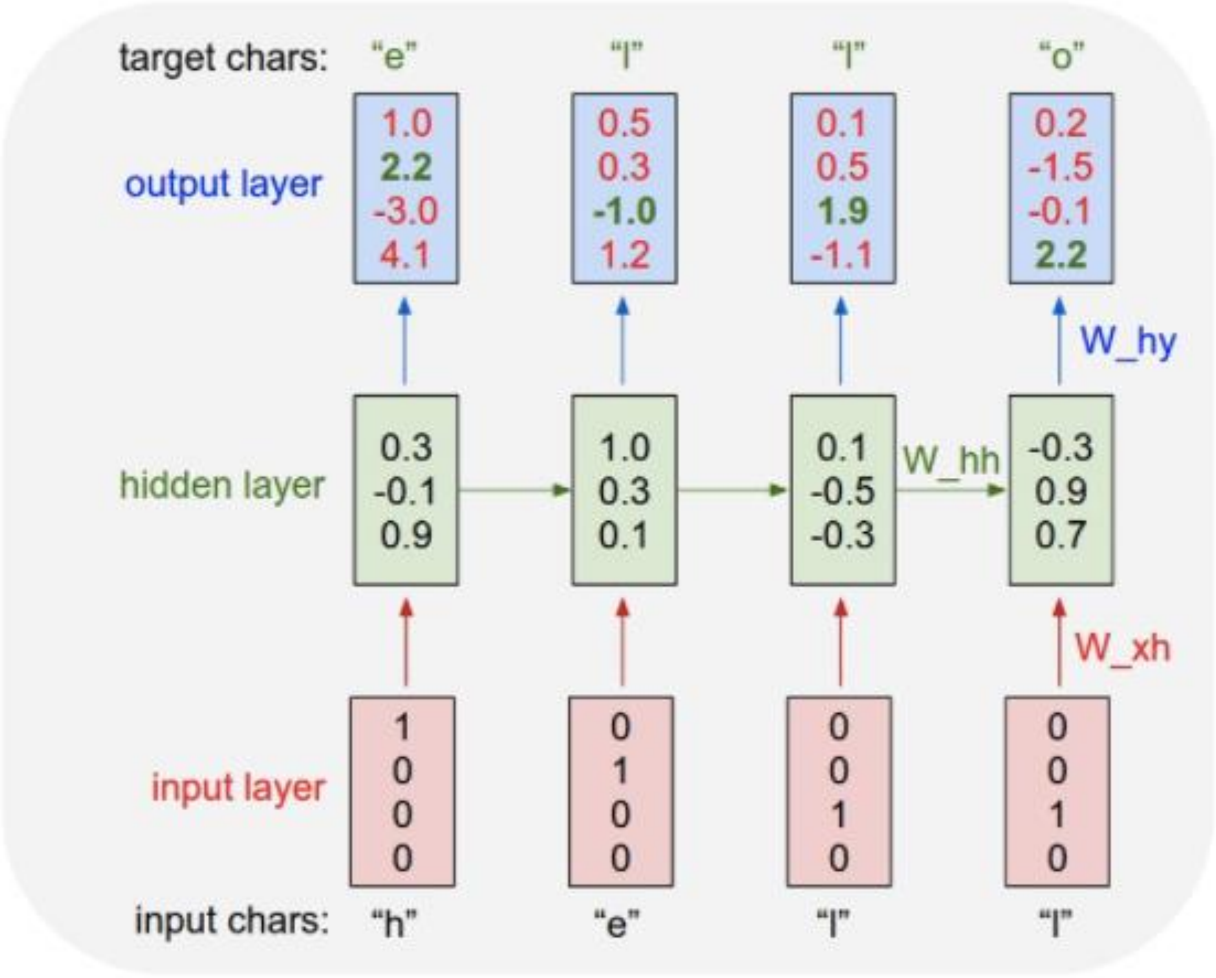




Recurrent Neural Network (RNN)



An illustrative example



Recurrent Neural Network (RNN)

output distribution

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

hidden states

$$\boldsymbol{h}^{(t)} = \sigma\left(\boldsymbol{W}_h\boldsymbol{h}^{(t-1)} + \boldsymbol{W}_e\boldsymbol{e}^{(t)} + \boldsymbol{b}_1\right)$$

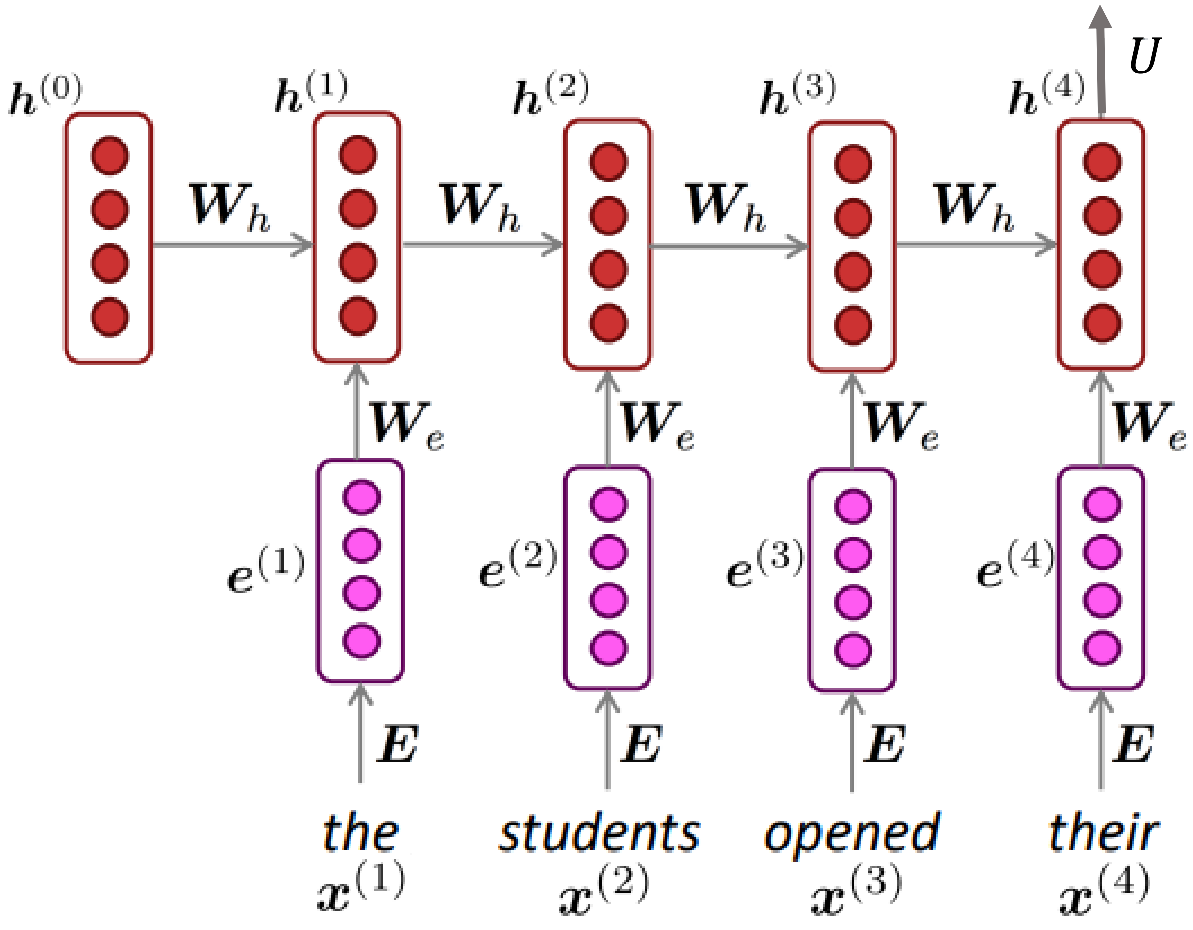
$\boldsymbol{h}^{(0)}$ is the initial hidden state

word embeddings

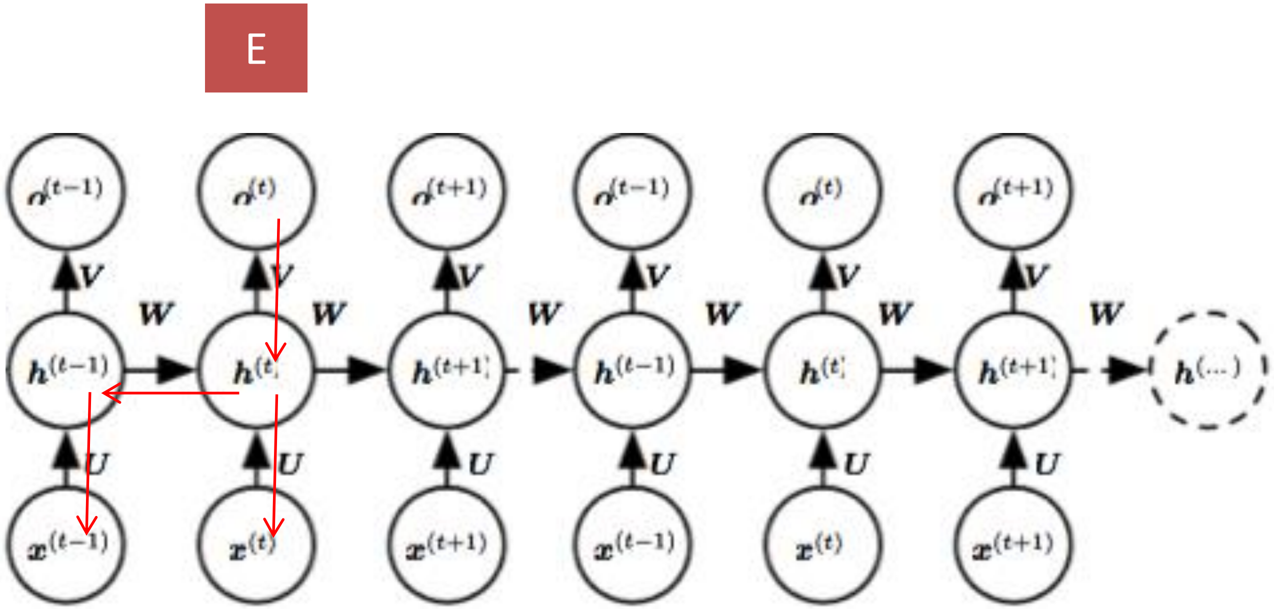
$$\boldsymbol{e}^{(t)} = \boldsymbol{E}\boldsymbol{x}^{(t)}$$

words / one-hot vectors

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

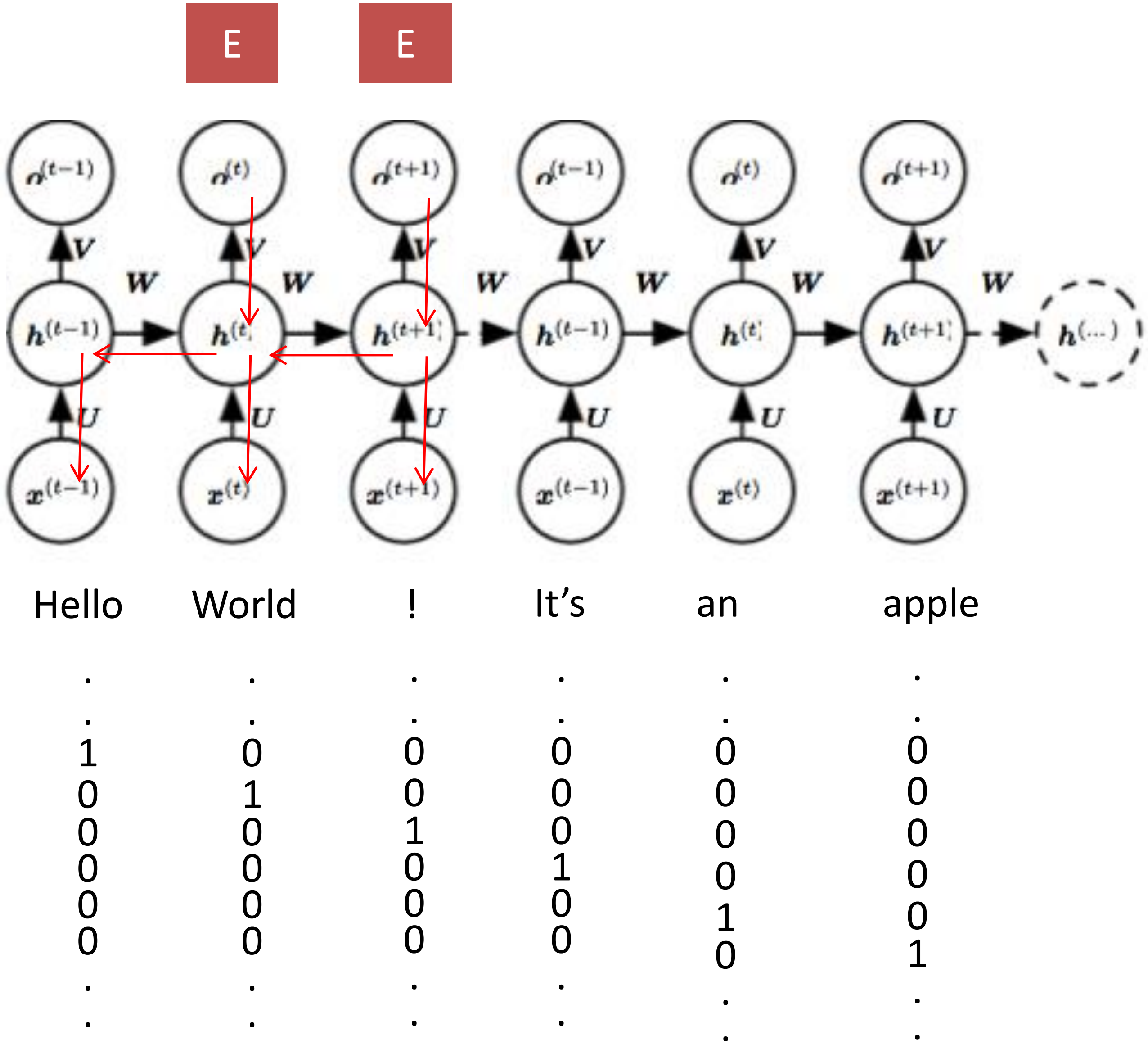


Process – Next word prediction

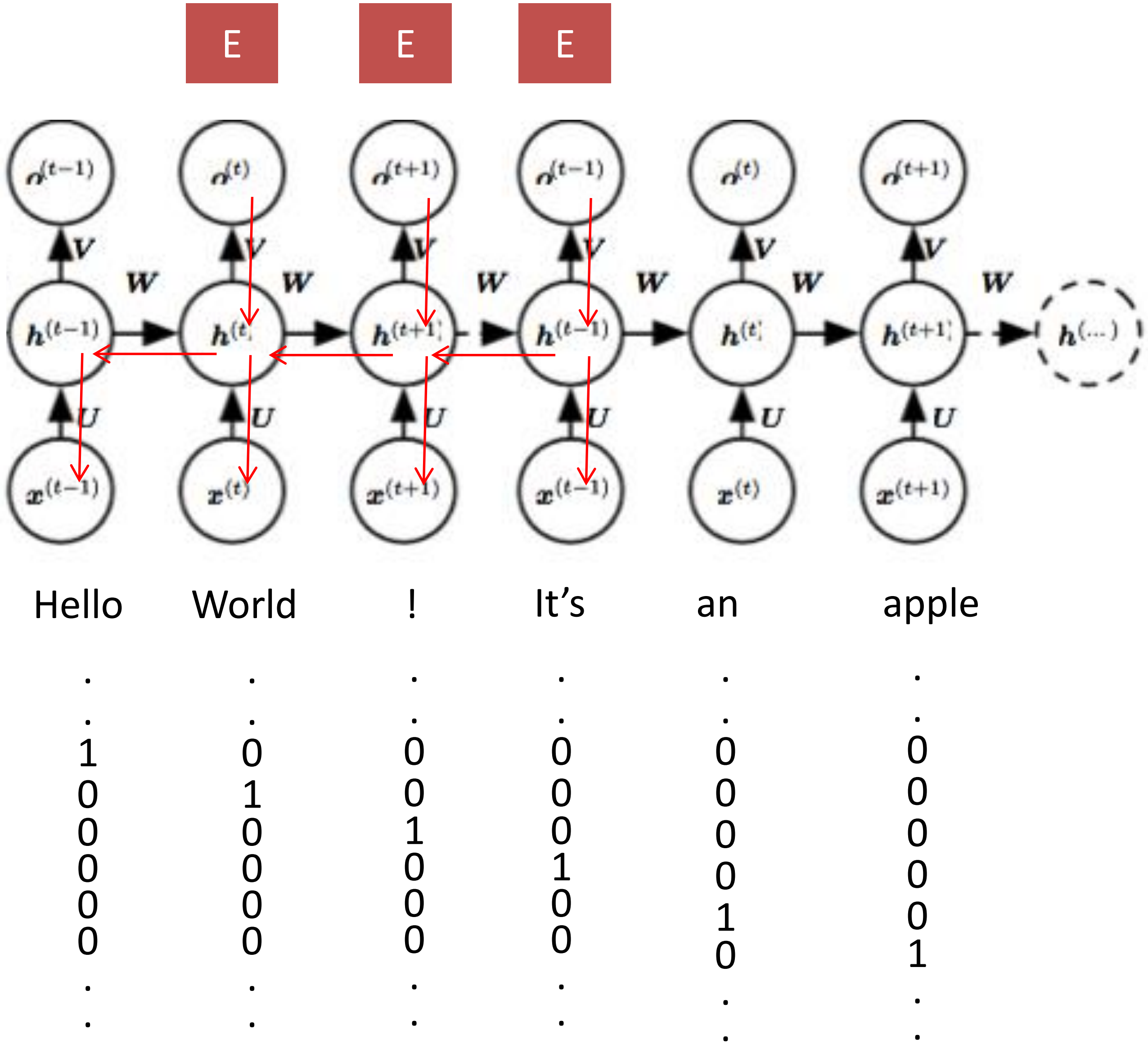


Hello	World	!	It's	an	apple
.	.	.	.	.	.
.	.	.	.	.	.
1	0	0	0	0	0
0	1	0	0	0	0
0	0	1	0	0	0
0	0	0	1	0	0
0	0	0	0	1	0
0	0	0	0	0	1
.	.	.	.	.	.
.	.	.	.	.	.

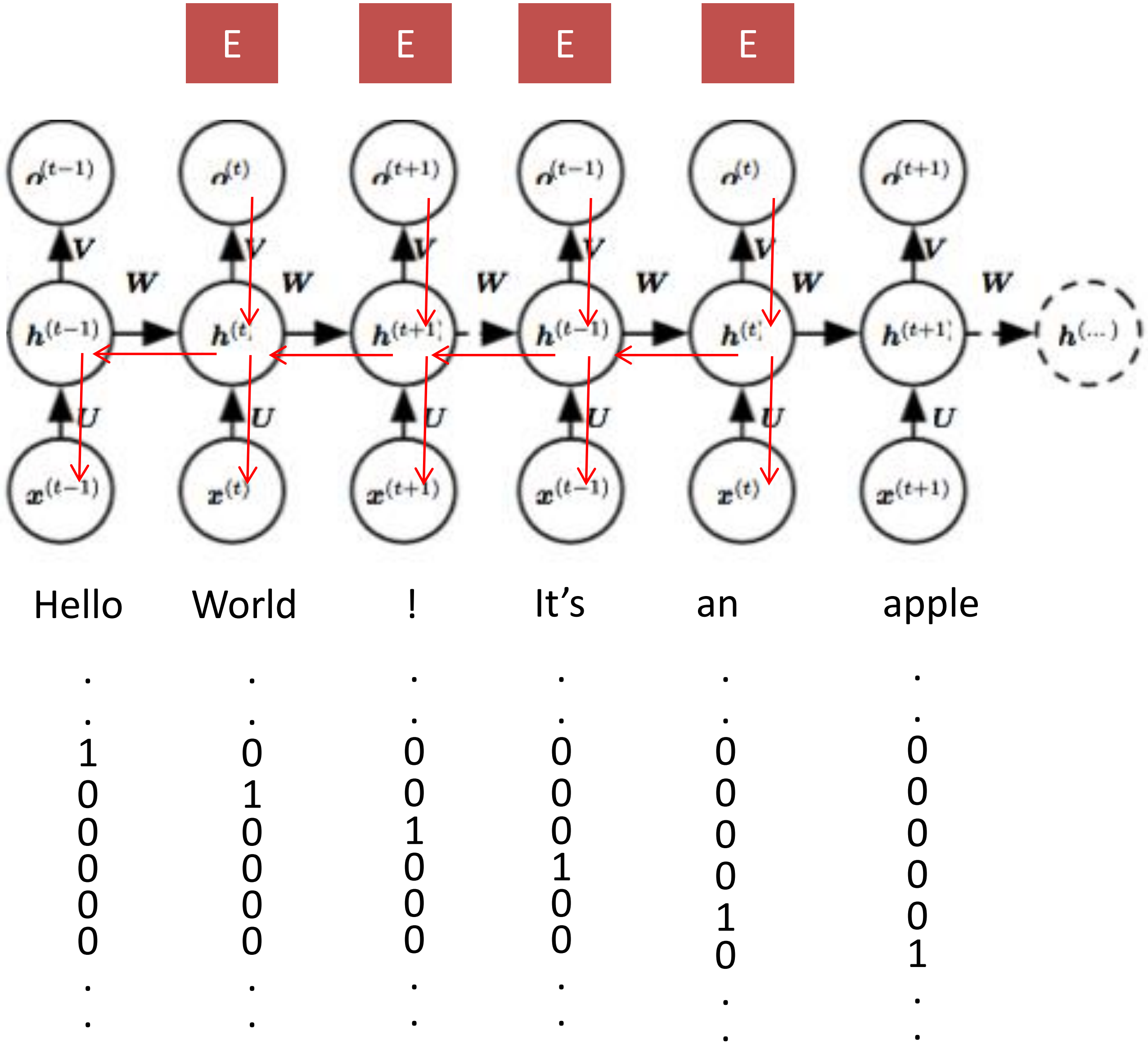
Process – Next word prediction



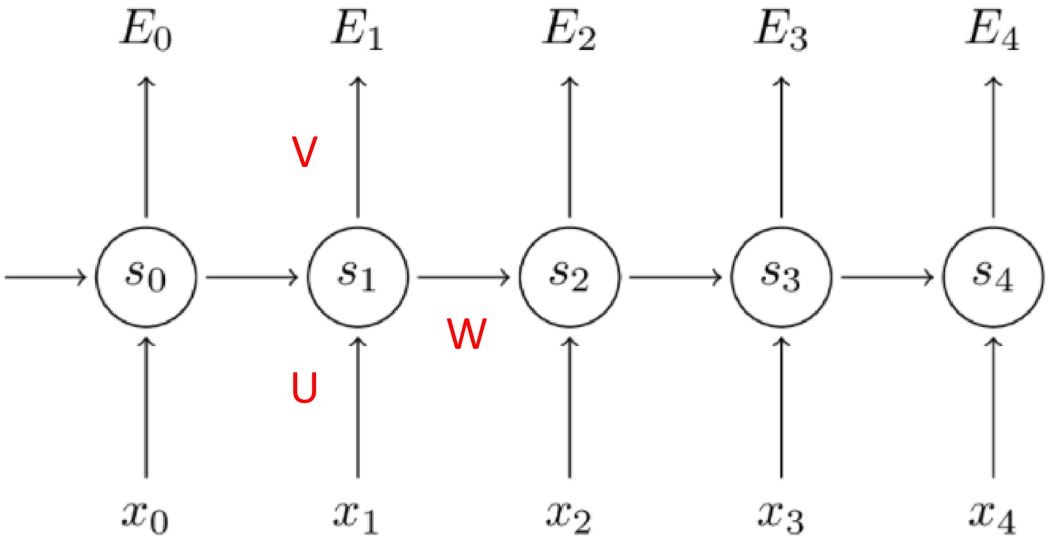
Process – Next word prediction



Process – Next word prediction



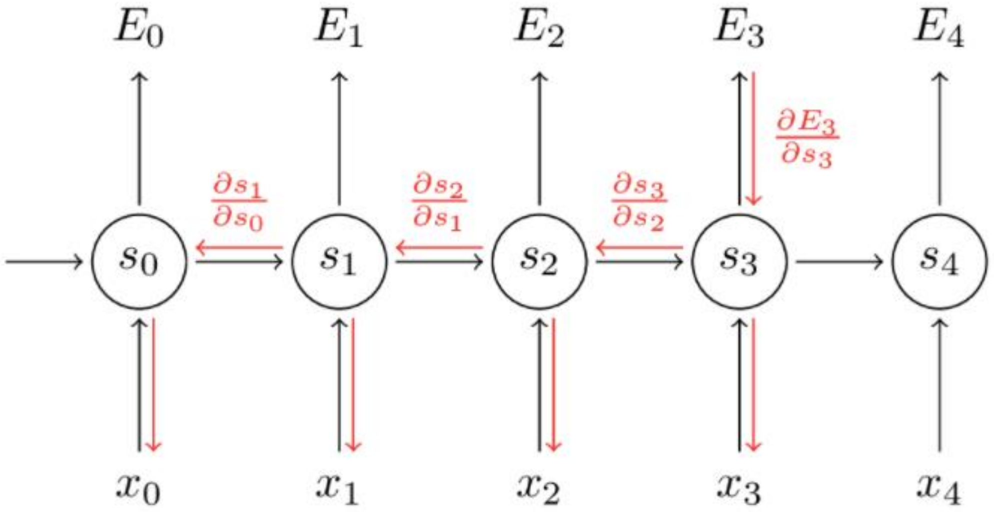
Model



$$s_t = \tanh(Ux_t + Ws_{t-1})$$
$$\hat{y}_t = softmax(Vs_t)$$

$$E(y_t, \hat{y}_t) = -y_t \log \hat{y}_t$$
$$E(y, \hat{y}) = -\sum_t E_t(y_t, \hat{y}_t)$$
$$= -\sum_t -y_t \log \hat{y}_t$$

Learning



$$\begin{aligned}\frac{\partial E_3}{\partial V} &= \frac{\partial E_3}{\partial \hat{y}_3}\frac{\partial \hat{y}_3}{\partial V}\\ &= \frac{\partial E_3}{\partial \hat{y}_3}\frac{\partial \hat{y}_3}{\partial z_3}\frac{\partial z_3}{\partial V}\\ &= (\hat{y}_3 - y_3) \otimes s_3\end{aligned}$$

$$\frac{\partial E_3}{\partial W} = \frac{\partial E_3}{\partial \hat{y}_3}\frac{\partial \hat{y}_3}{\partial s_3}\frac{\partial s_3}{\partial W}$$

$$\frac{\partial E_3}{\partial W} = \sum_{k=0}^{3} \frac{\partial E_3}{\partial \hat{y}_3}\frac{\partial \hat{y}_3}{\partial s_3}\frac{\partial s_3}{\partial s_k}\frac{\partial s_k}{\partial W}$$

RNN Applications

Automatic Text Generation

```
PANDARUS:
Alas, I think he shall be come approached and the day
When little srain would be attain'd into being never fed,
And who is but a chain and subjects of his death,
I should not sleep.

Second Senator:
They are away this miseries, produced upon my soul,
Breaking and strongly should be buried, when I perish
The earth and thoughts of many states.

DUKE VINCENTIO:
Well, your wit is in the care of side and that.

Second Lord:
They would be ruled after this chamber, and
my fair nues begun out of the fact, to be conveyed,
Whose noble souls I'll have the heart of the wars.

Clown:
Come, sir, I will make did behold your worship.

VIOLA:
I'll drink it.
```

http://karpathy.github.io/2015/05/21/rnn-effectiveness/

RNN Applications

Automatic Image Caption Generation

"man in black shirt is playing guitar."

"construction worker in orange safety vest is working on road."

"two young girls are playing with lego toy."

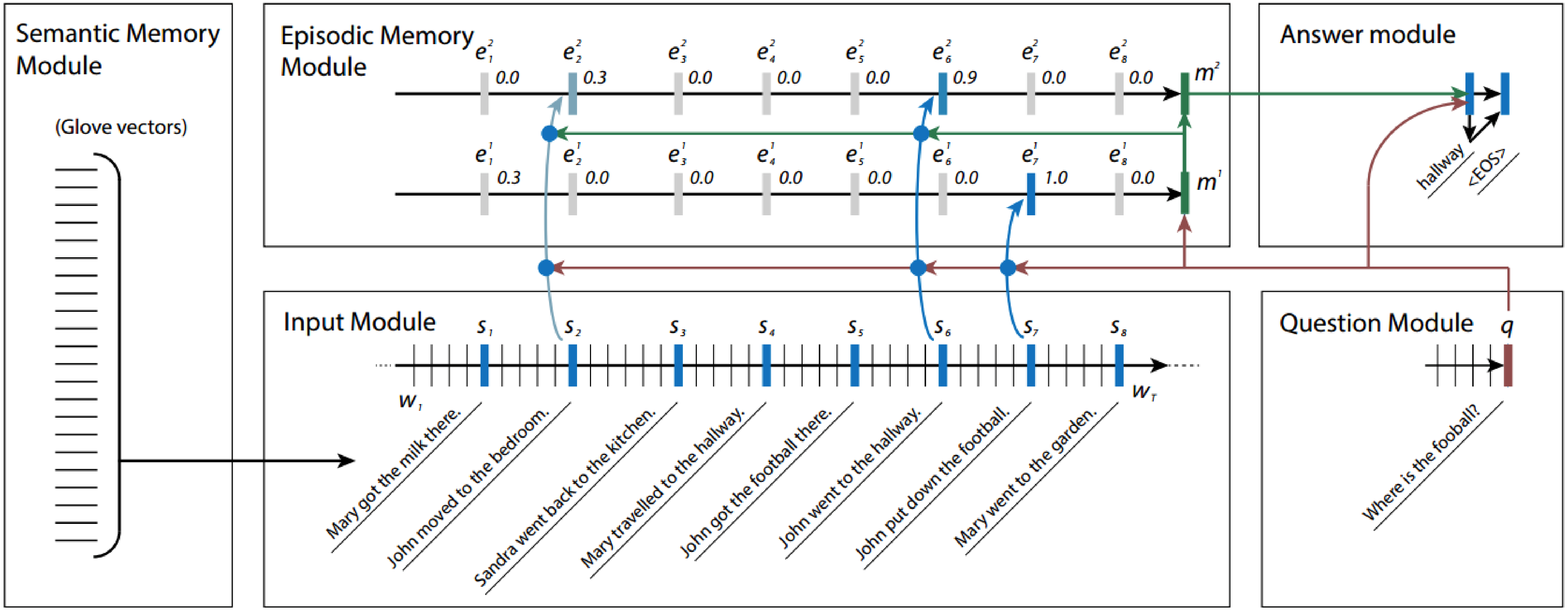
"girl in pink dress is jumping in air."

"black and white dog jumps over bar."

"young girl in pink shirt is swinging on swing."

http://cs.stanford.edu/people/karpathy/deepimagesent/rankingdemo/

RNN Applications– Question Answering



RNN Applications – Machine Translation