Lecture 9:
Natural Language Processing

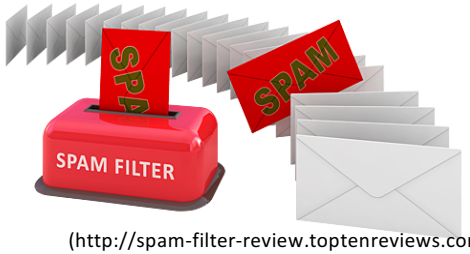
Data Science, Fall 2018

Hong-Han Shuai

Thanks to Prof. Hung-yi Lee from NTU for the slides.

Language Technology

spam detection



(http://spam-filter-review.toptenreviews.com/)

Part-of-speech Tagging

John saw the saw.

PN V D N

Name Entity Recognition

這 位 是 明 金 城 武

Name of People

Sentiment Analysis

這部電影太糟了

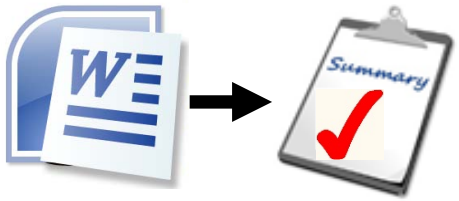
Negative (負雷)

Translation

“Machine learning ……”

“機器學習 ……”

Summarization



document summary

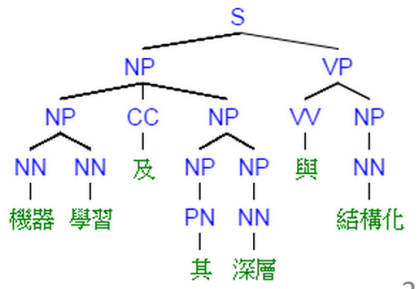
Retrieval

Google 台灣

Speech Recognition

大家好……

Syntactic Analysis



Do machine really understand human language?

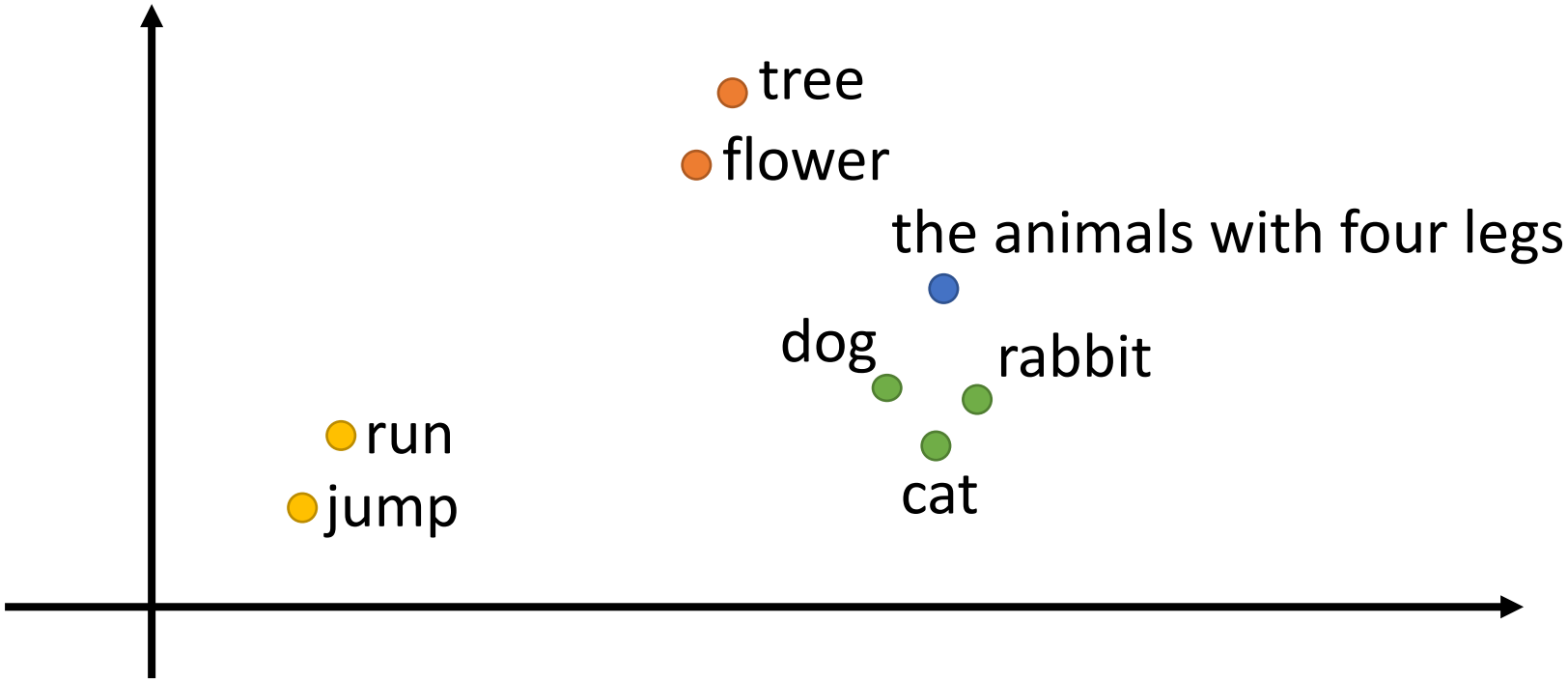
http://cse3521.artifice.cc/chinese-room.html

“A word is known by the company it keeps”

John Rupert Firth

Meaning Representation

Do machine know the meaning of a word or word sequence?

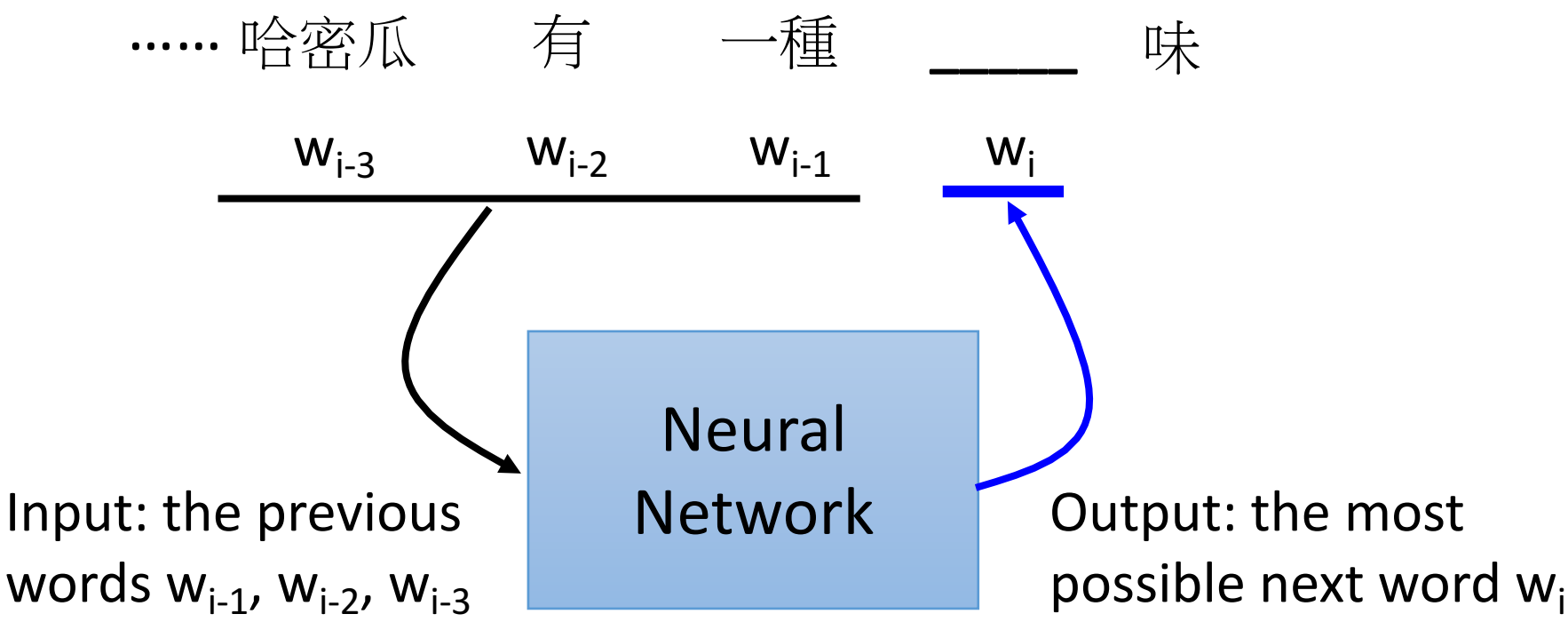


Meaning of Word

Predicting the next word

噓 hsiaoyoshye: 麻煩這系列的請到政黑或其他地方討論好嗎?這裡是八 04/27 00:40
推 lrfnc: 仙 04/27 00:51
推 headiron: 樂 04/27 00:52
推 victorshu: 園 04/27 00:57

Fill in the Blank



Each word should be represented as a feature vector.

Fill in the Blank

1-of-N Encoding

lexicon = {apple, bag, cat, dog, elephant}

apple = [1 0 0 0 0]

bag = [0 1 0 0 0]

cat = [0 0 1 0 0]

dog = [0 0 0 1 0]

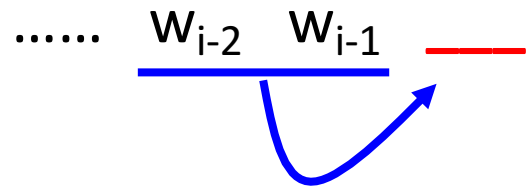
elephant = [0 0 0 0 1]

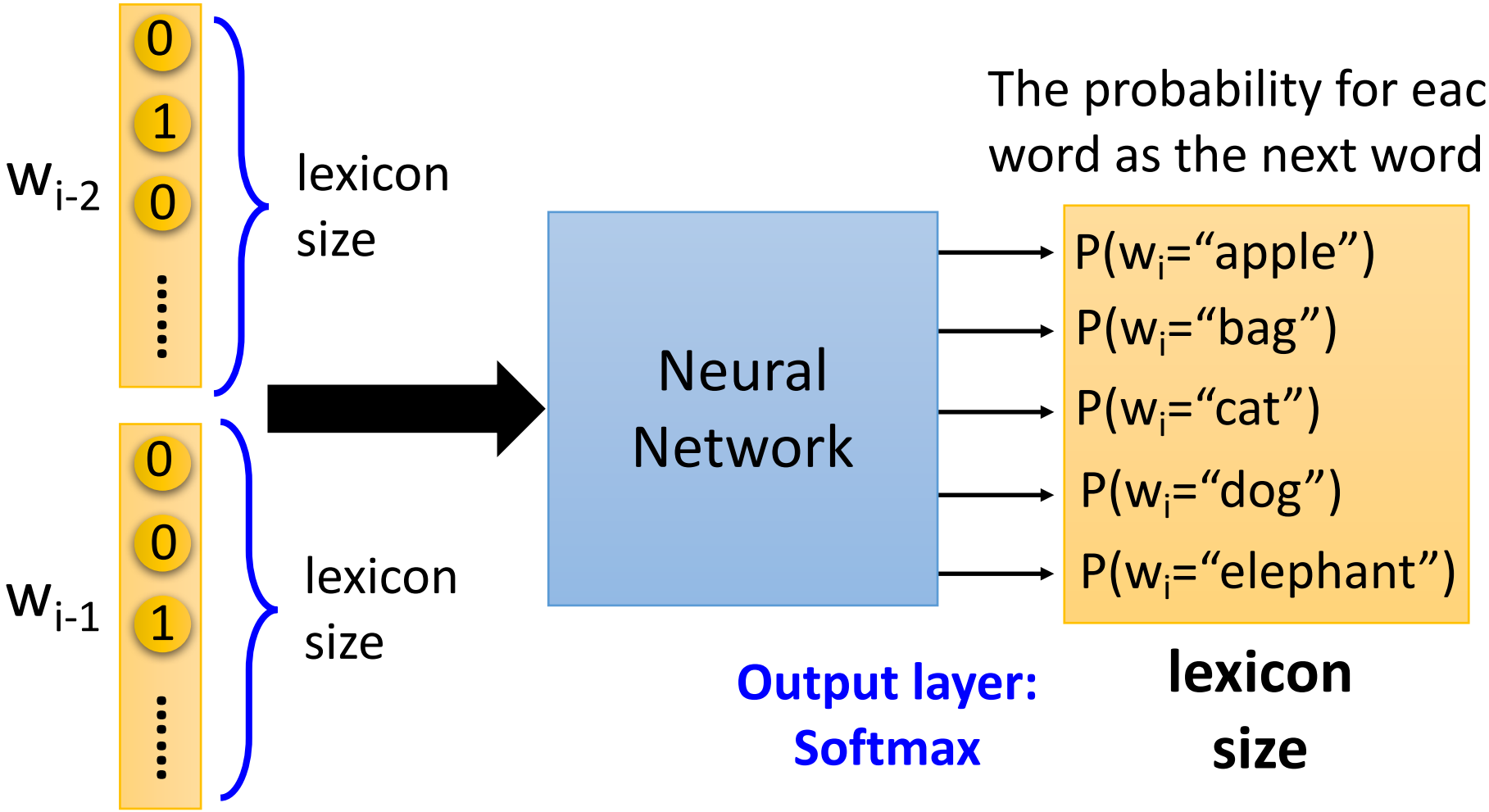
The vector is lexicon size.

Each dimension corresponds to a word in the lexicon

The dimension for the word is 1, and others are 0

Fill in the Blank





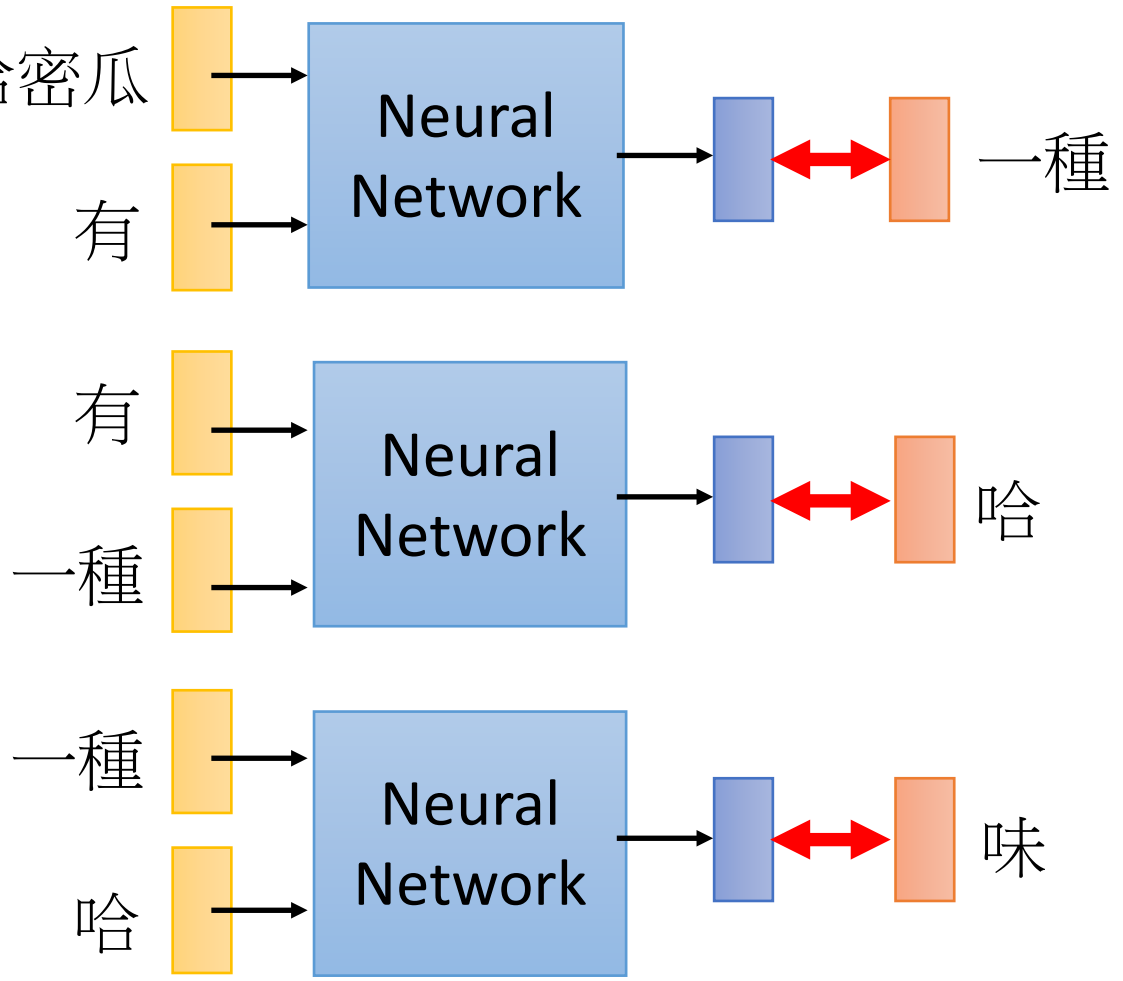
Fill in the Blank

- Training:

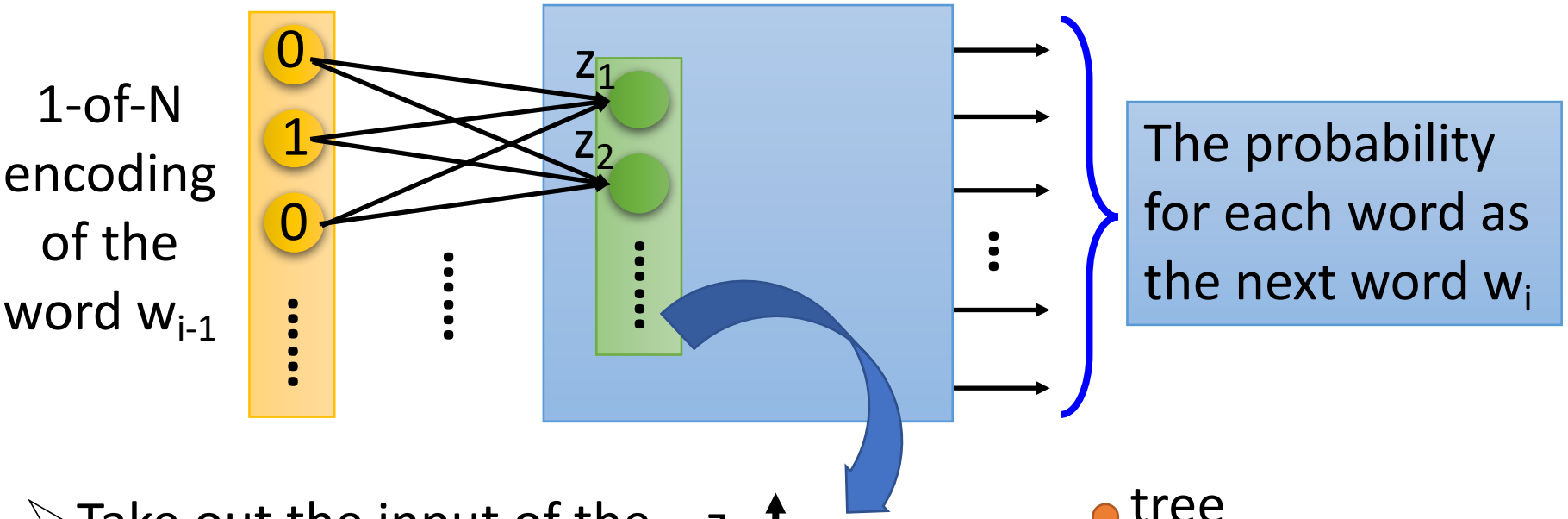
Collect data:

哈密瓜 有 一種 哈 味
不爽 不要 買
公道價 八萬 一
……

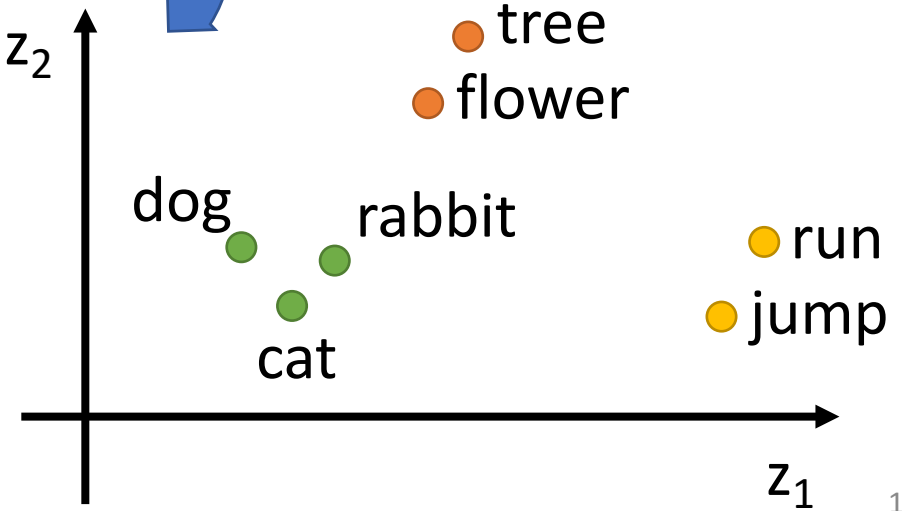
Minimizing cross entropy



Word Vector

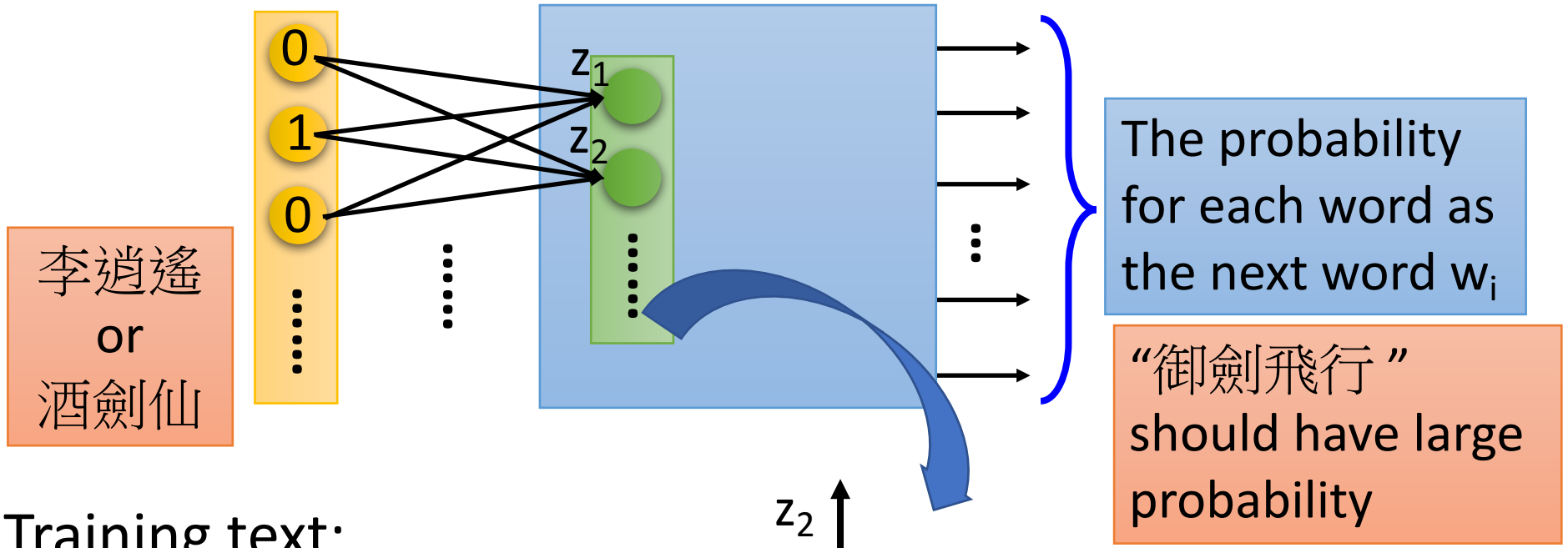


- Take out the input of the neurons in the first layer
- Use it to represent a word w
- Word vector, word embedding feature: V(w)



Word Vector

You shall know a word by the company it keeps



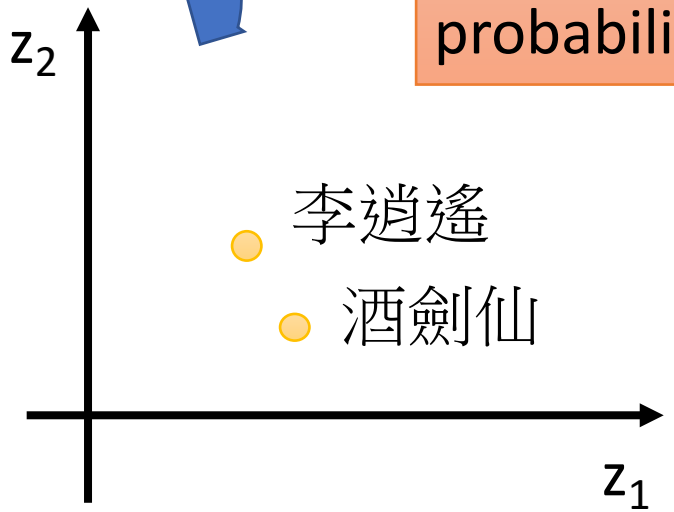
Training text:

…… 李逍遙 御劍飛行 ……

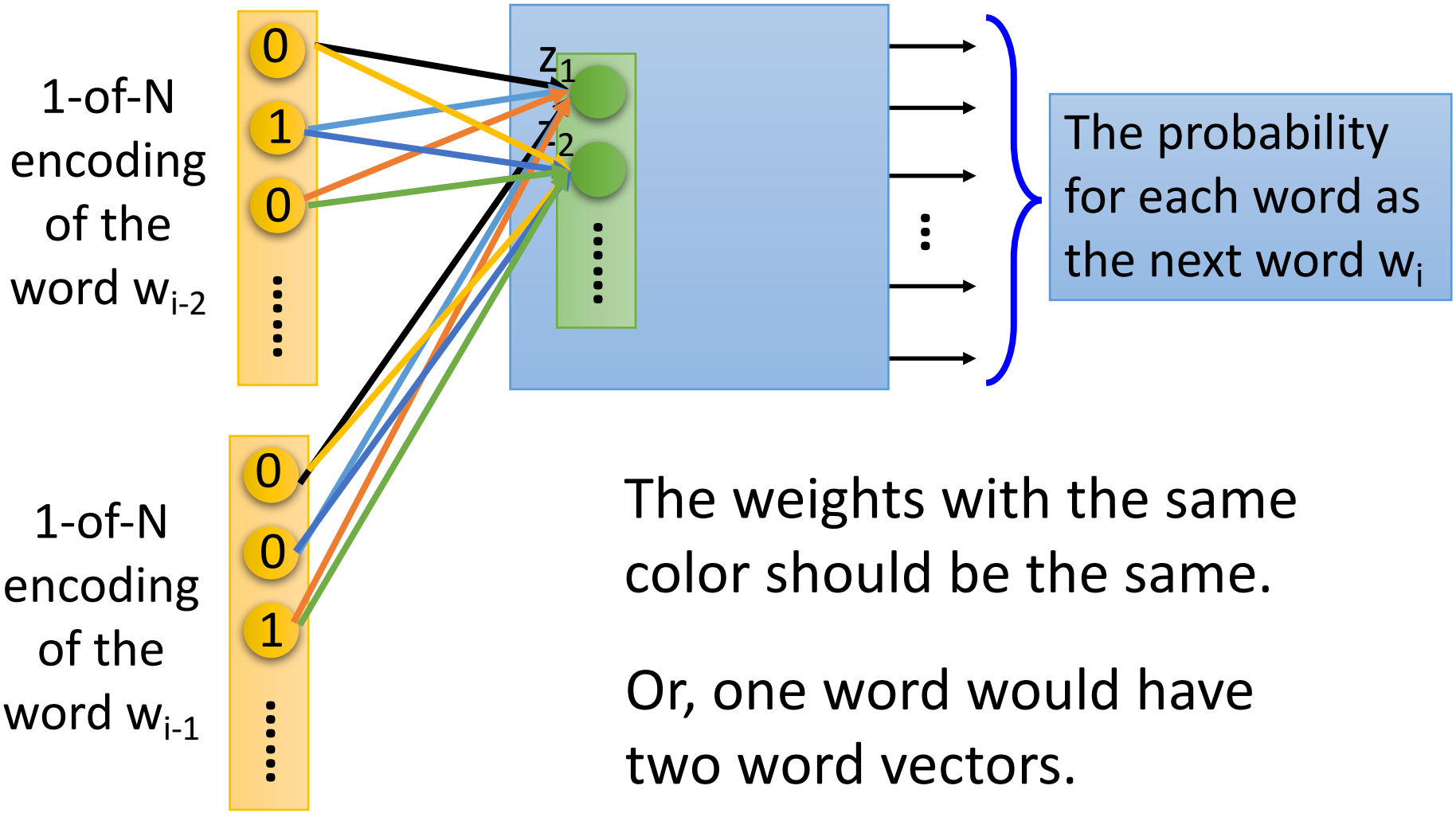
w_{i-1} w_i

…… 酒劍仙 御劍飛行 ……

w_{i-1} w_i



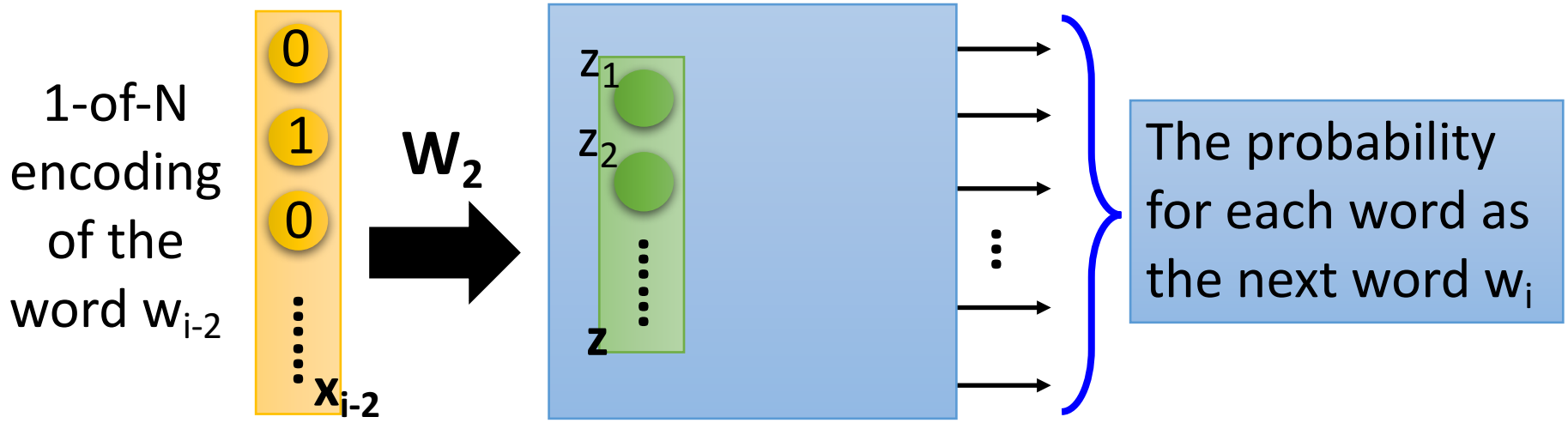
Word Vector – Sharing Parameters



The weights with the same color should be the same.

Or, one word would have two word vectors.

Word Vector – Sharing Parameters



The length of $\mathbf{x}_{i-1}$ and $\mathbf{x}_{i-2}$ are both $|V|$.

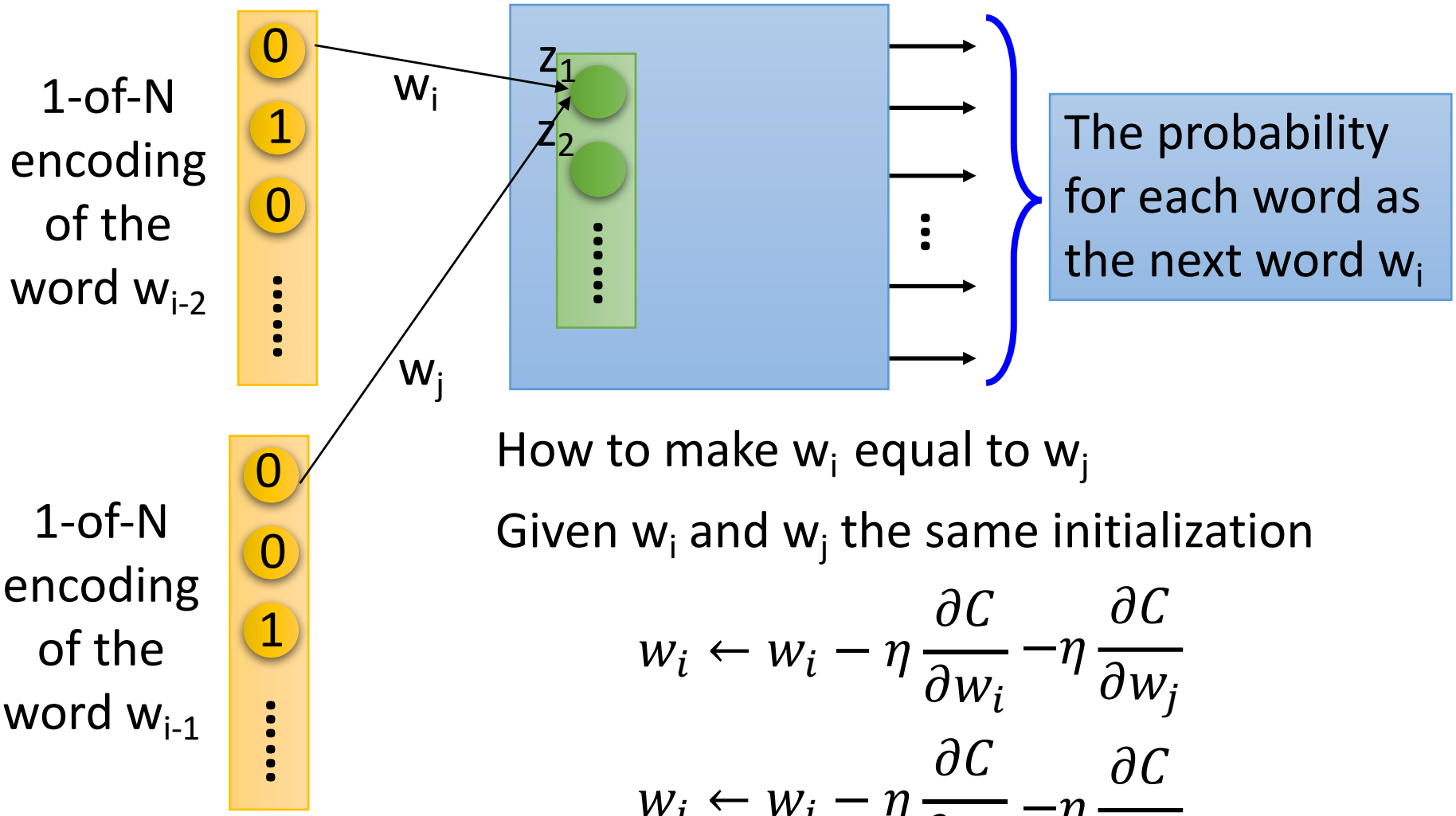
The length of $\mathbf{z}$ is $|Z|$.

$$\mathbf{z} = \mathbf{W}_1 \mathbf{x}_{i-1} + \mathbf{W}_2 \mathbf{x}_{i-2}$$

The weight matrix $\mathbf{W}_1$ and $\mathbf{W}_2$ are both $|Z|X|V|$ matrices.

$$\mathbf{W}_1 = \mathbf{W}_2 = \mathbf{W} \Rightarrow \mathbf{z} = \mathbf{W}(\mathbf{x}_{i-2} + \mathbf{x}_{i-1})$$

Word Vector – Sharing Parameters



How to make w_i equal to w_j

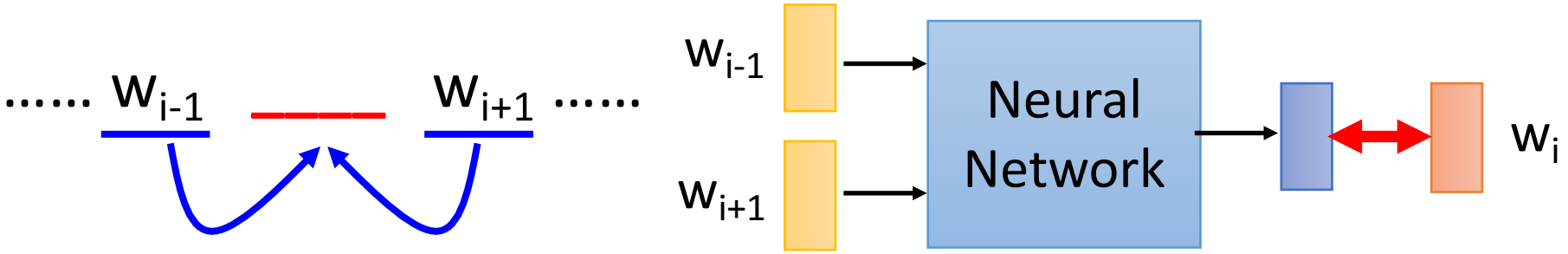
Given w_i and w_j the same initialization

$$w_i \leftarrow w_i - \eta \frac{\partial C}{\partial w_i} - \eta \frac{\partial C}{\partial w_j}$$

$$w_j \leftarrow w_j - \eta \frac{\partial C}{\partial w_j} - \eta \frac{\partial C}{\partial w_i}$$

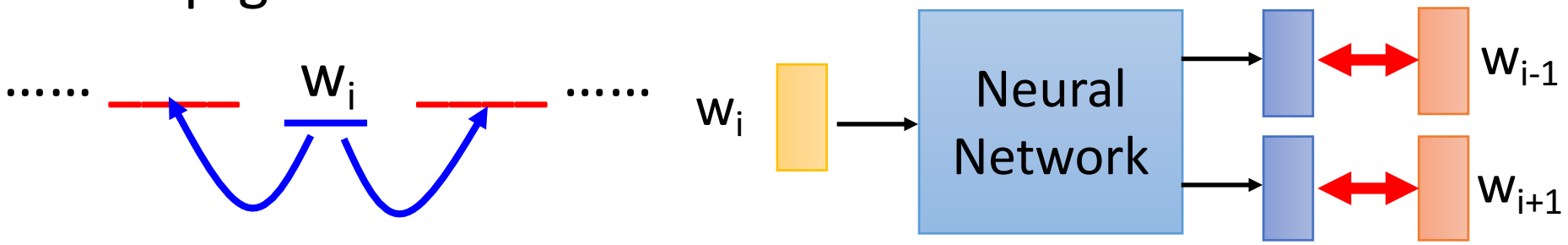
Word Vector – Various Architectures

- Continuous bag of word (CBOW) model



predicting the word given its context

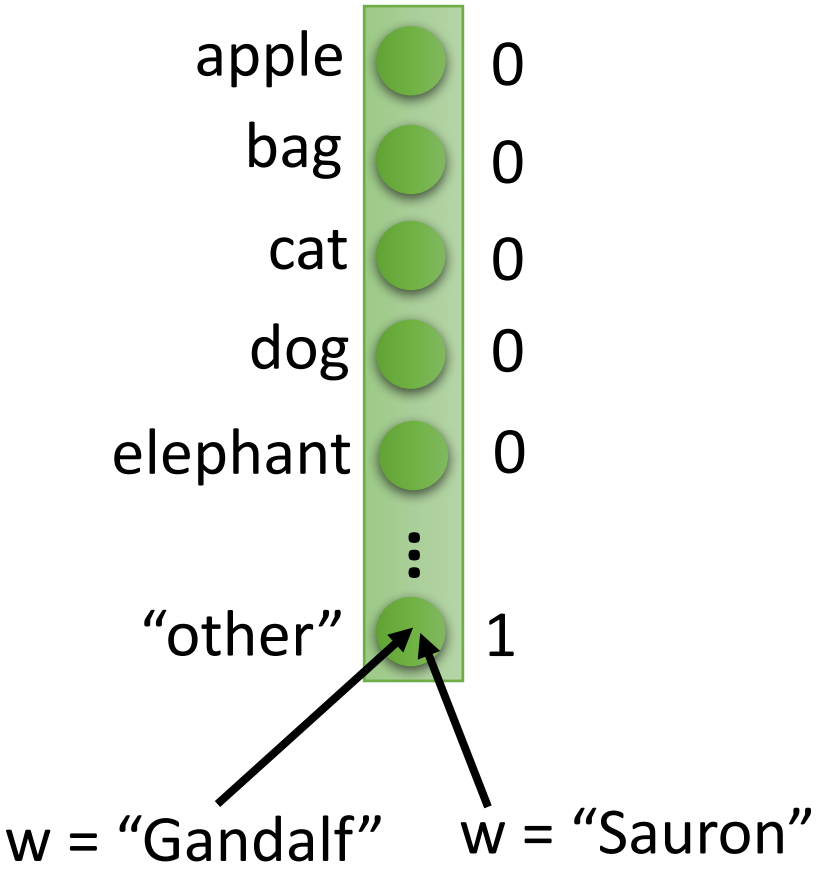
- Skip-gram



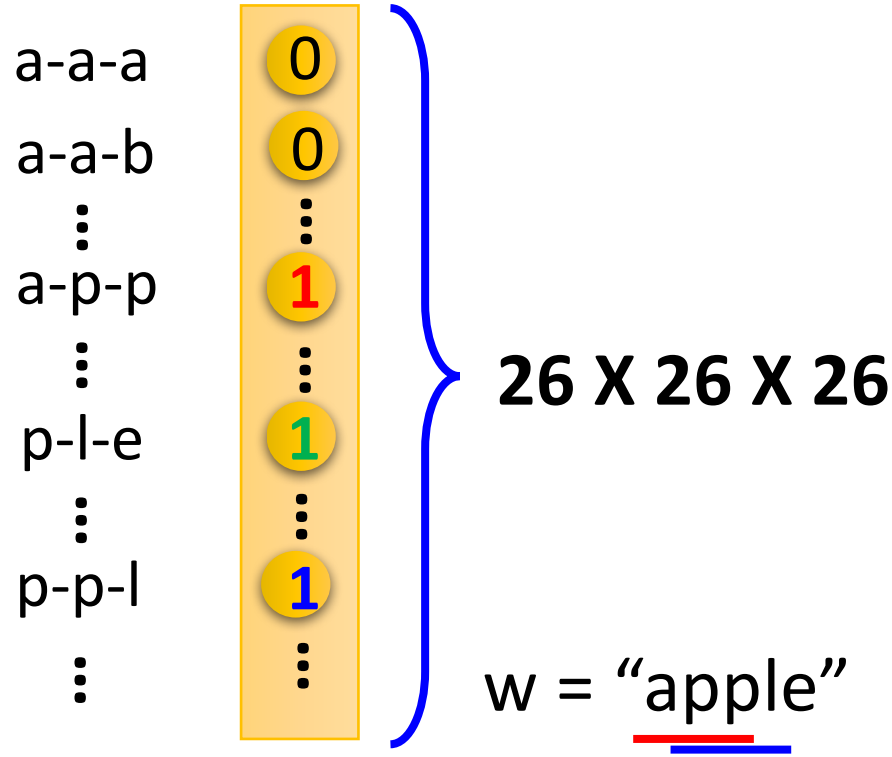
predicting the context given a word

Beyond 1-of-N encoding

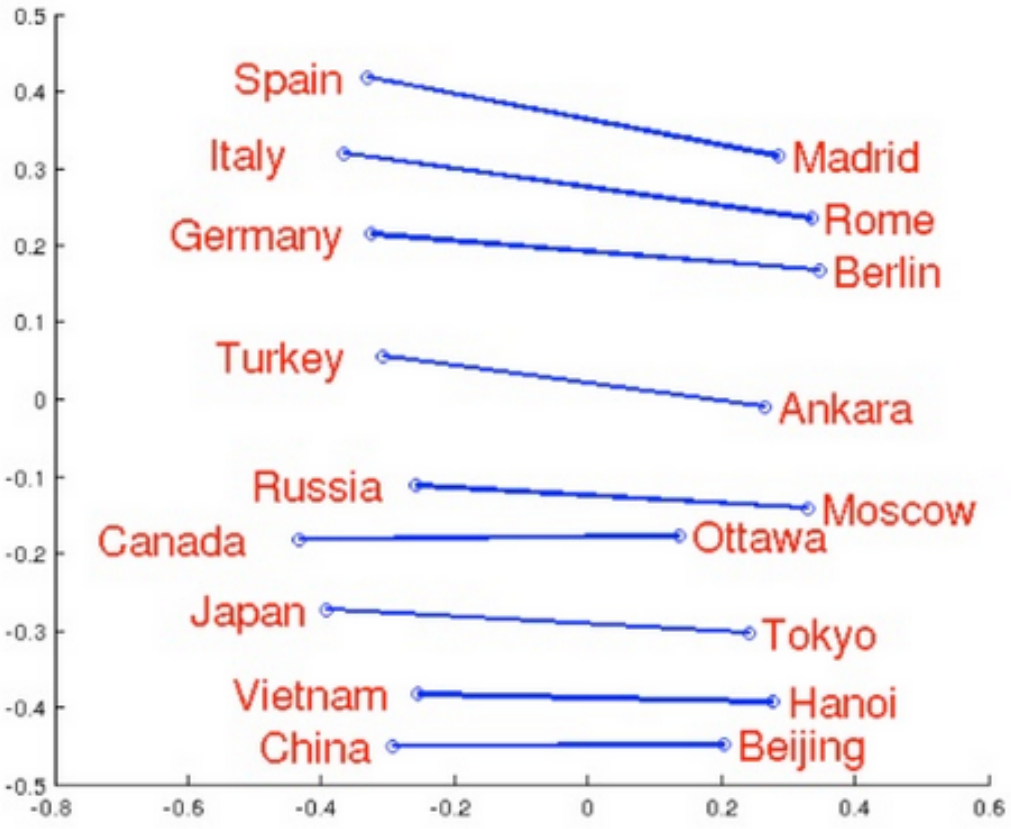
Dimension for "Other"

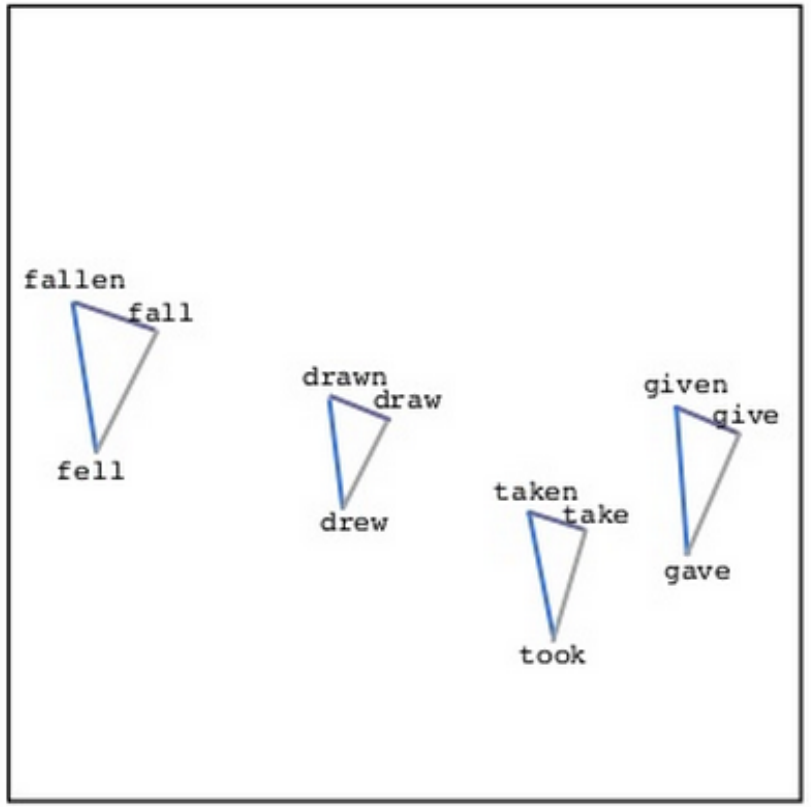


Word hashing



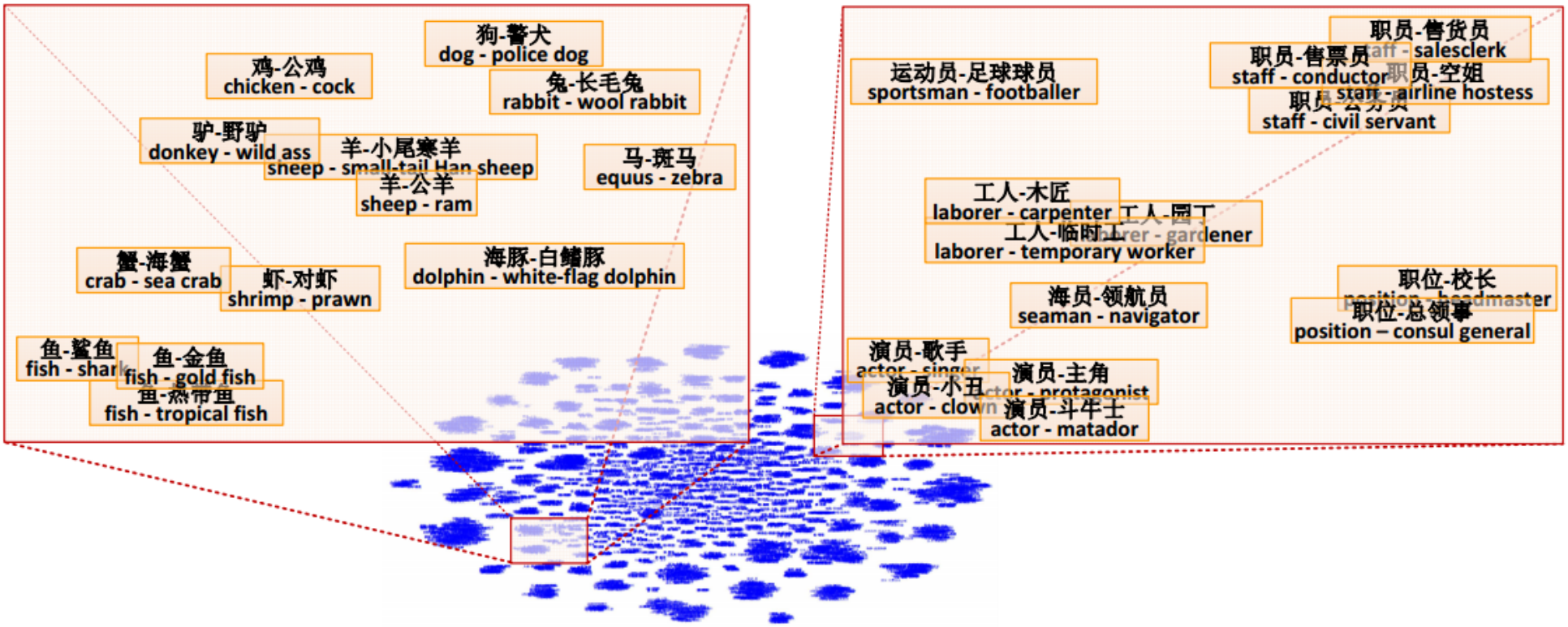
Word Vector





Source: http://www.slideshare.net/hustwj/cikm-keynotenov2014

Word Vector



Fu, Ruiji, et al. "Learning semantic hierarchies via word embeddings."*Proceedings of the 52th Annual Meeting of the Association for Computational Linguistics: Long Papers*. Vol. 1. 2014.

Word Vector

$$V(Germany) \approx V(Berlin) - V(Rome) + V(Italy)$$

- Characteristics

$$V(hotter) - V(hot) \approx V(bigger) - V(big)$$
$$V(Rome) - V(Italy) \approx V(Berlin) - V(Germany)$$
$$V(king) - V(queen) \approx V(uncle) - V(aunt)$$

- Solving analogies

Rome : Italy = Berlin : ?

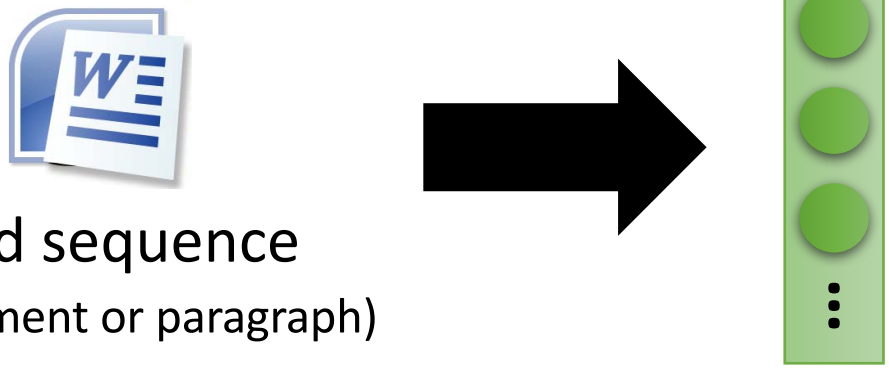
Compute $V(Berlin) - V(Rome) + V(Italy)$

Find the word w with the closest V(w)

Meaning of Word Sequence

Meaning of Word Sequence

- word sequences with different lengths → the vector with the same length
  - The vector representing the meaning of the word sequence
  - A word sequence can be a document or a paragraph



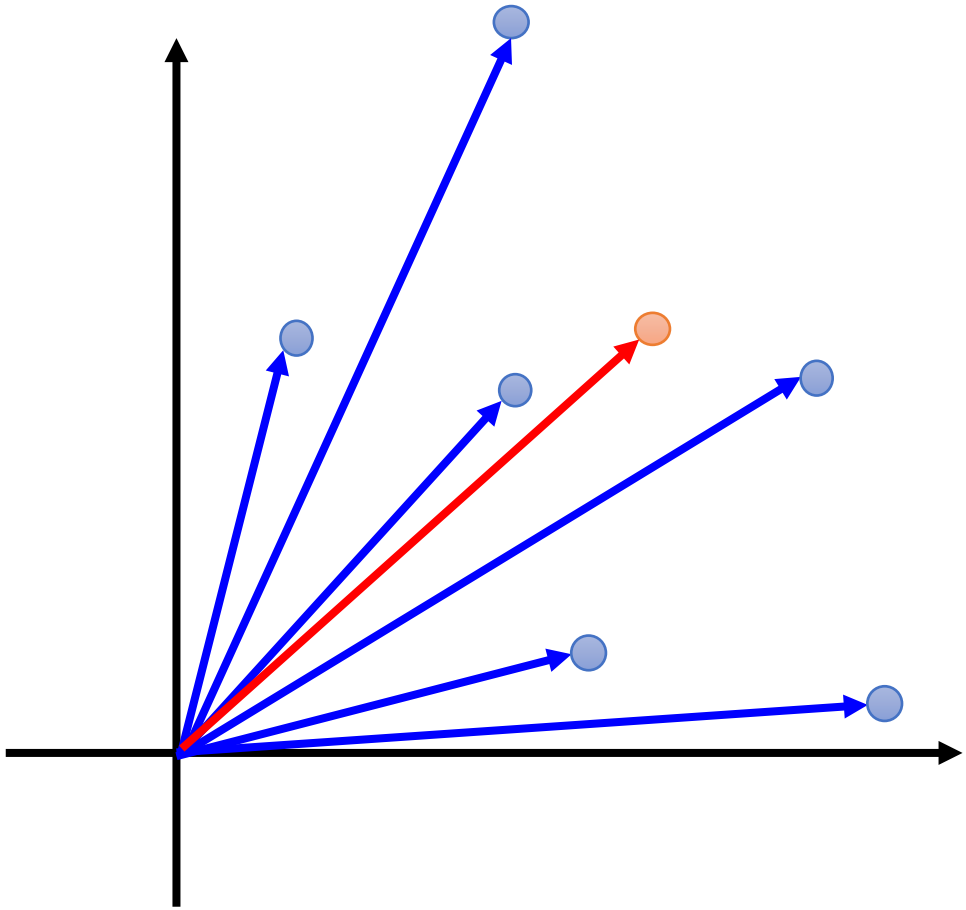
word sequence

(a document or paragraph)

Outline

Deep Structured Semantic Model (DSSM)	• Application: Information Retrieval (IR)
Recursive Neural Network	• Application: Sentiment Analysis, Sentence Relatedness
Unsupervised	• Paragraph Vector • Sequence-to-sequence auto-encoder

Information Retrieval (IR)



Vector Space Model

The documents are vectors in the space.

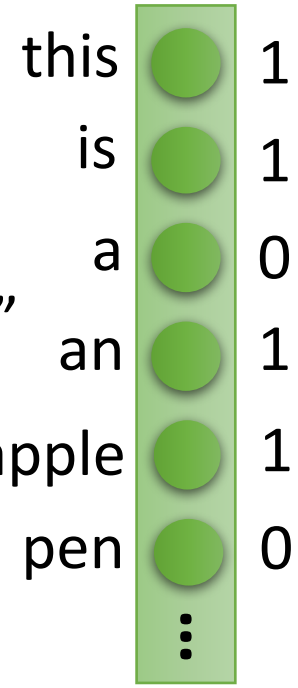
The query is also a vector.

How to use a vector to represent word sequences

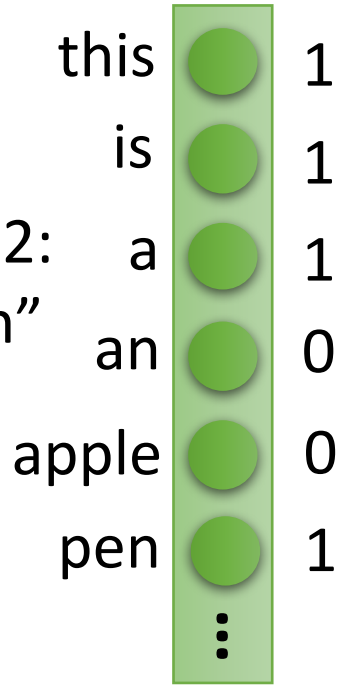
Information Retrieval (IR)

Bag-of-word

word string s1:
"This is an apple"



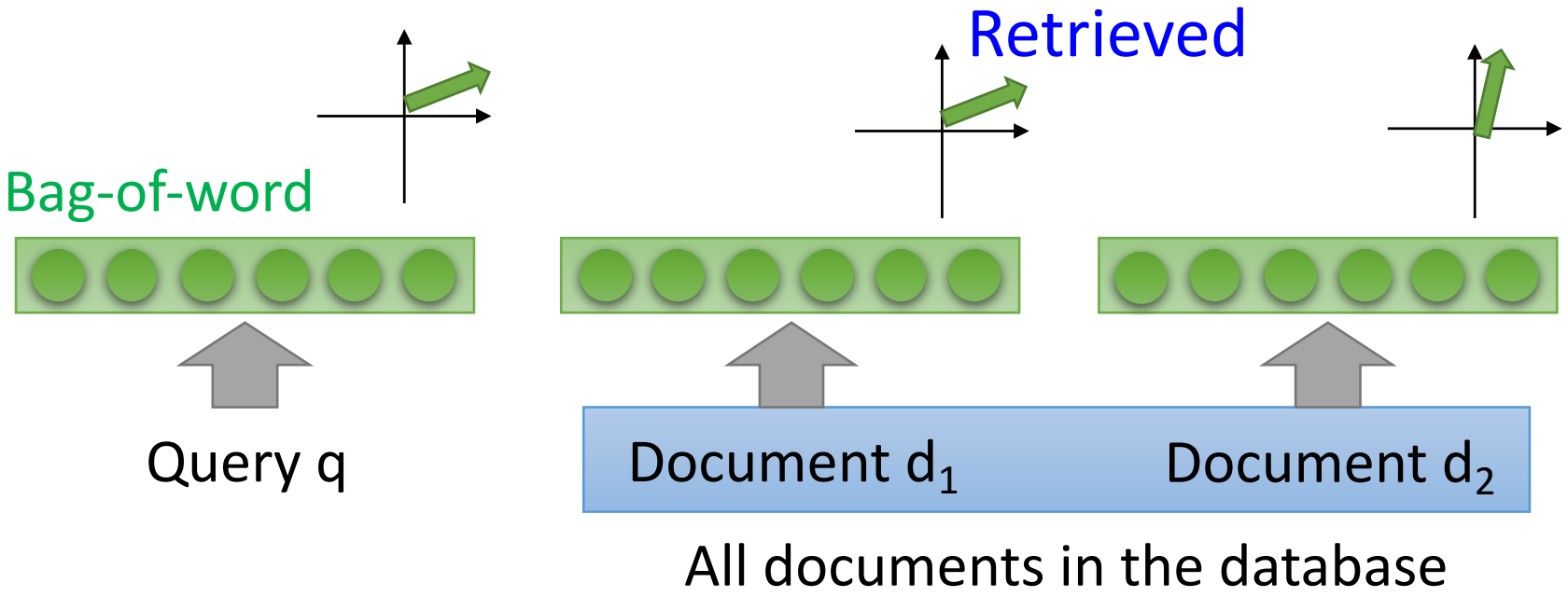
word string s2:
"This is a pen"



Weighted by IDF

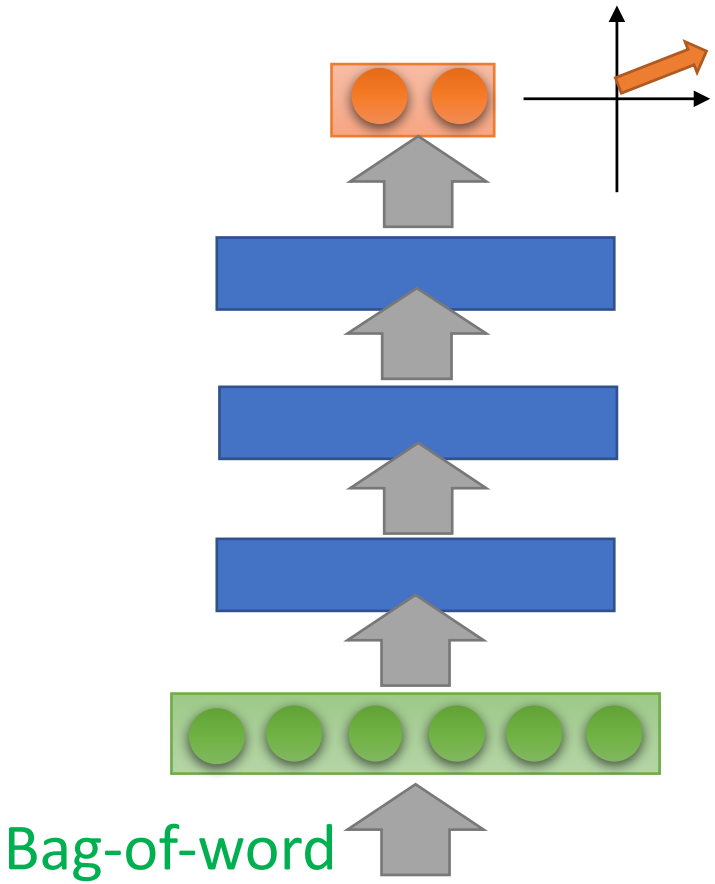
Information Retrieval (IR)

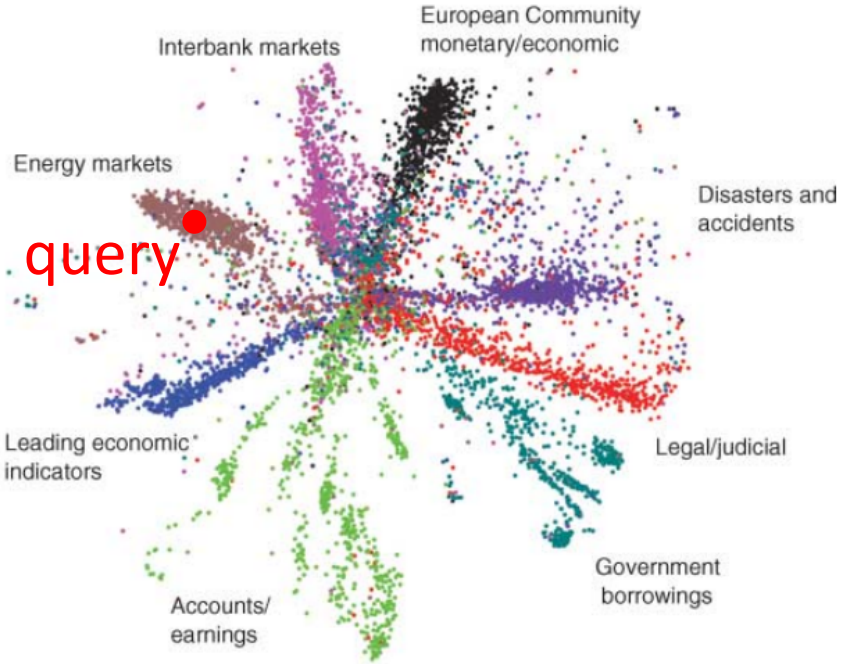
Vector Space Model + Bag-of-word



- All the words are treated as discrete tokens.
- Never considered: Different words can have the same meaning, and the same word can have different meanings.

IR - Semantic Embedding





Reference: Hinton, Geoffrey E., and Ruslan R. Salakhutdinov. "Reducing the dimensionality of data with neural networks." *Science* 313.5786 (2006): 504-507

How to achieve that? (No target ……)

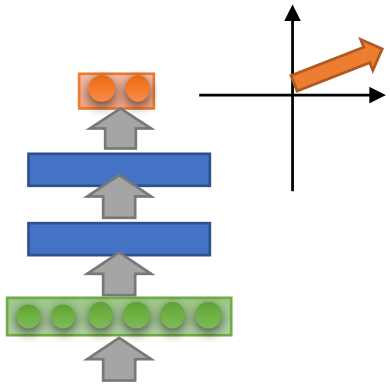
DSSM

Click-through data: $q_1 \longrightarrow d_1 : +$ $d_2 : -$

$q_2 \longrightarrow d_3 : -$ $d_4 : +$

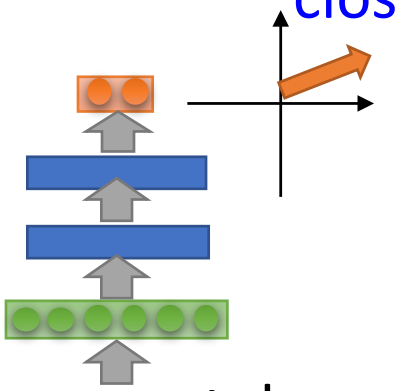
……

Training:



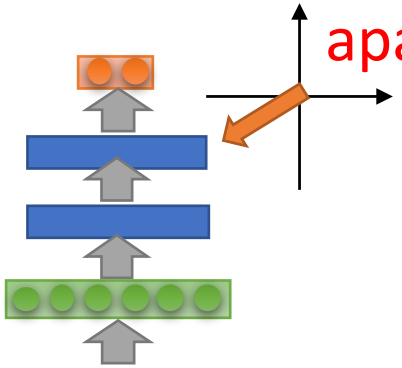
query q_1

close

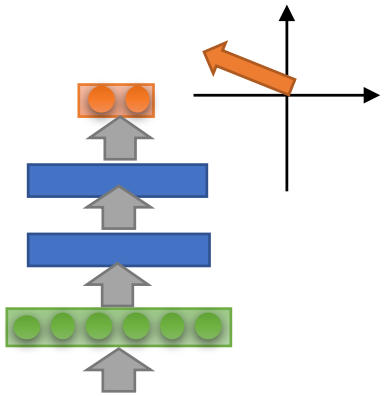


document d_1 +

Far apart

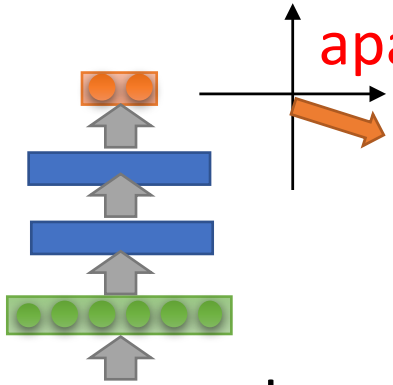


document d_2 -



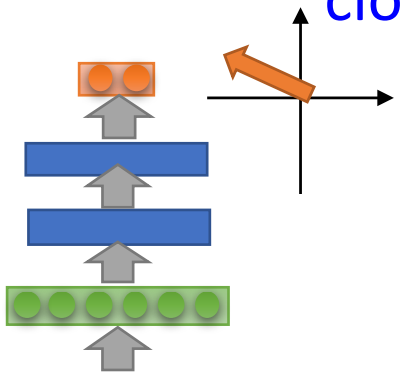
query q_2

Far apart



document d_3 -

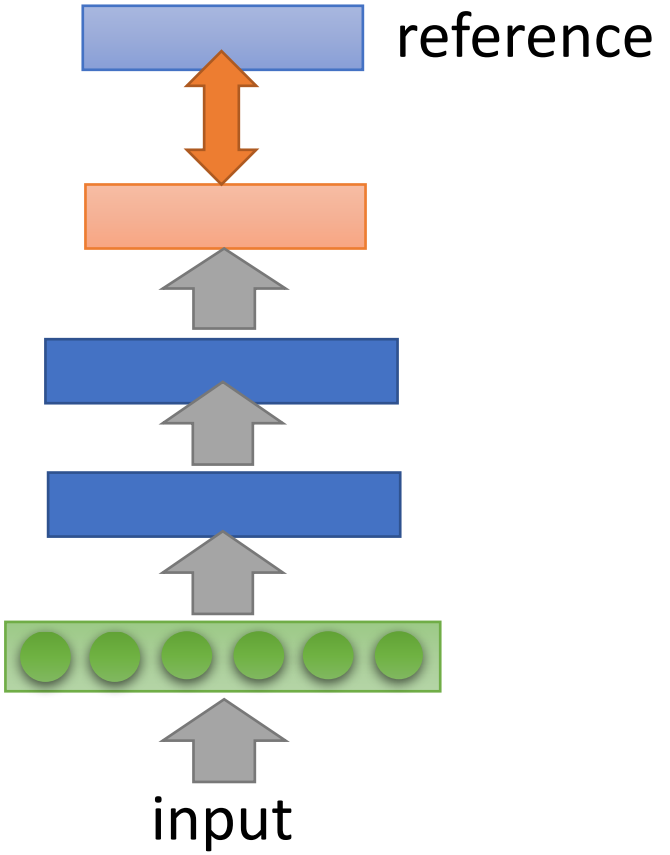
close



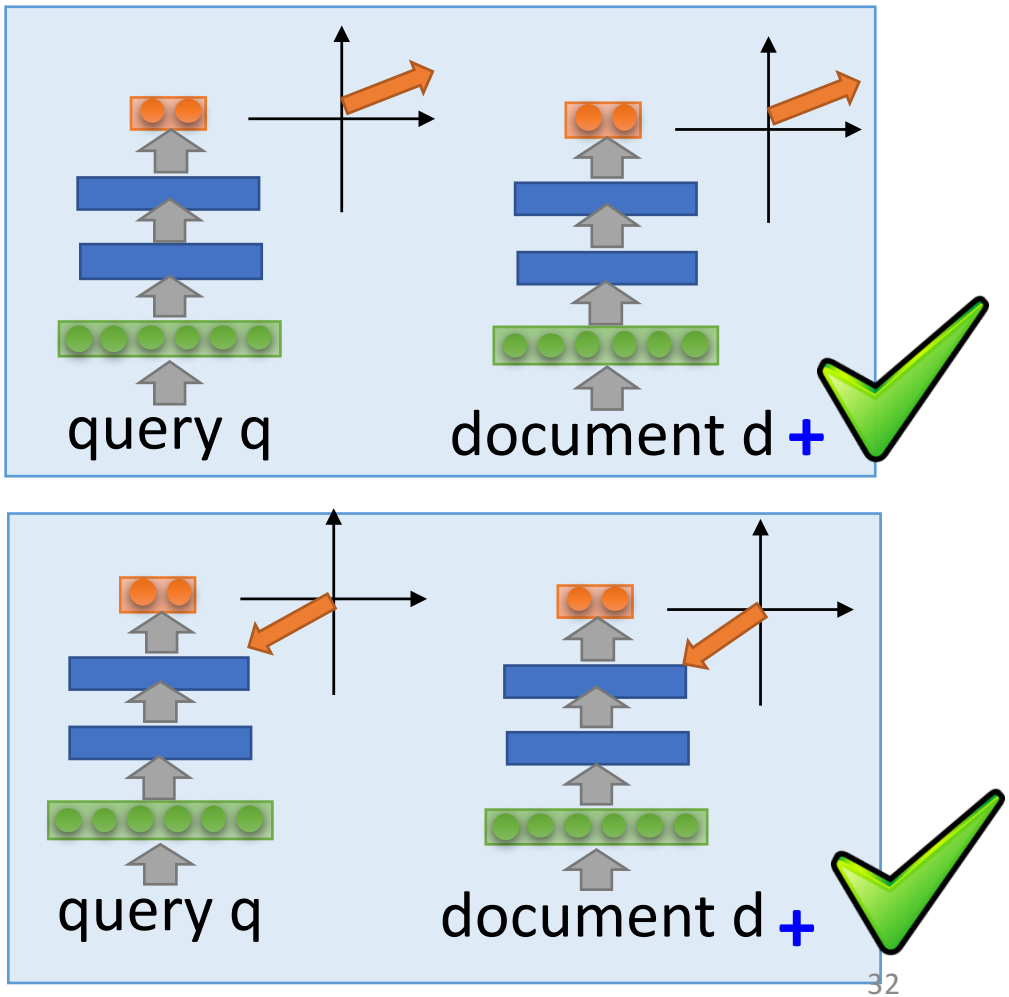
document d_4 +

DSSM v.s. Typical DNN

Typical DNN



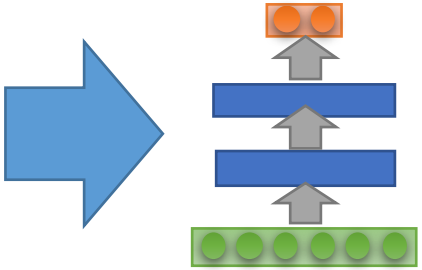
DSSM



Click-through data: $q_1 \longrightarrow d_1 : +\quad d_2 : -$

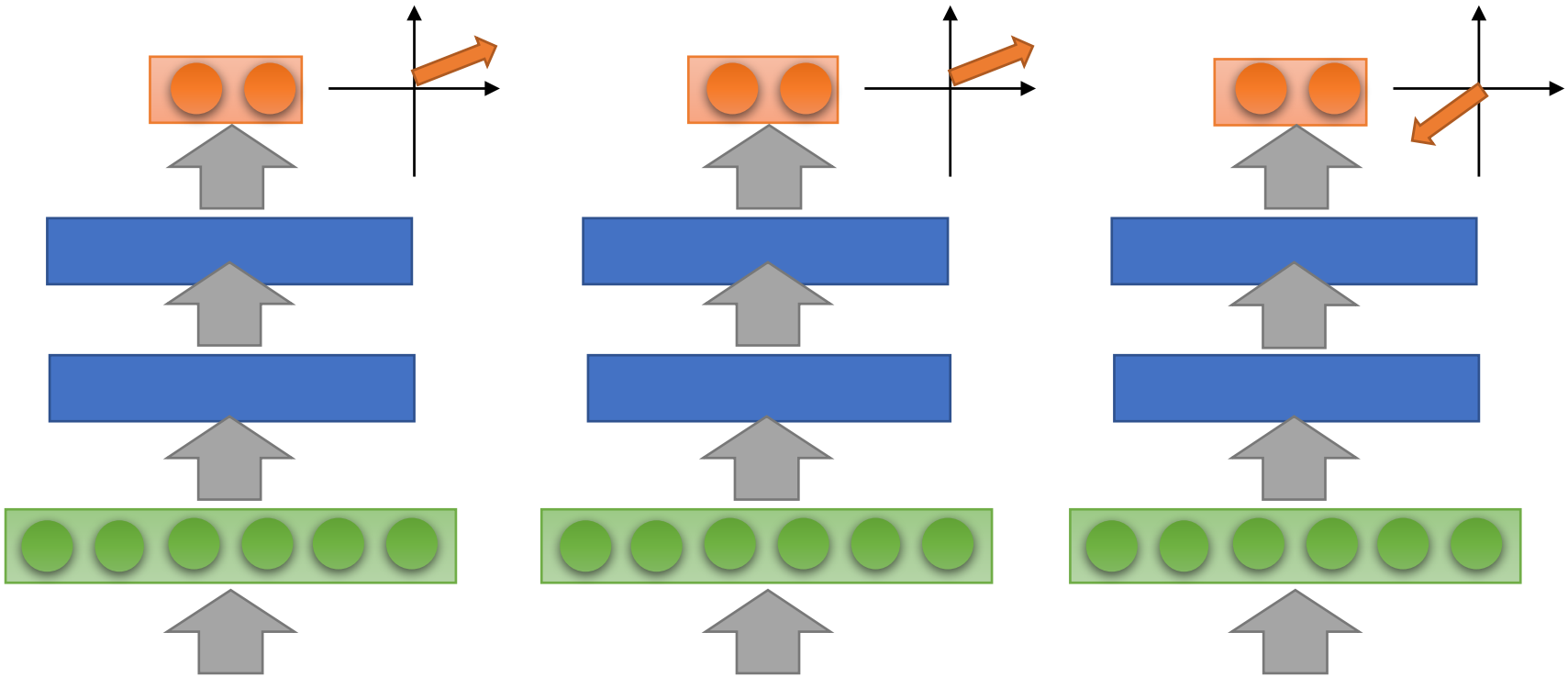
$q_2 \longrightarrow d_3 : -\quad d_4 : +$

……



- How to do retrieval?

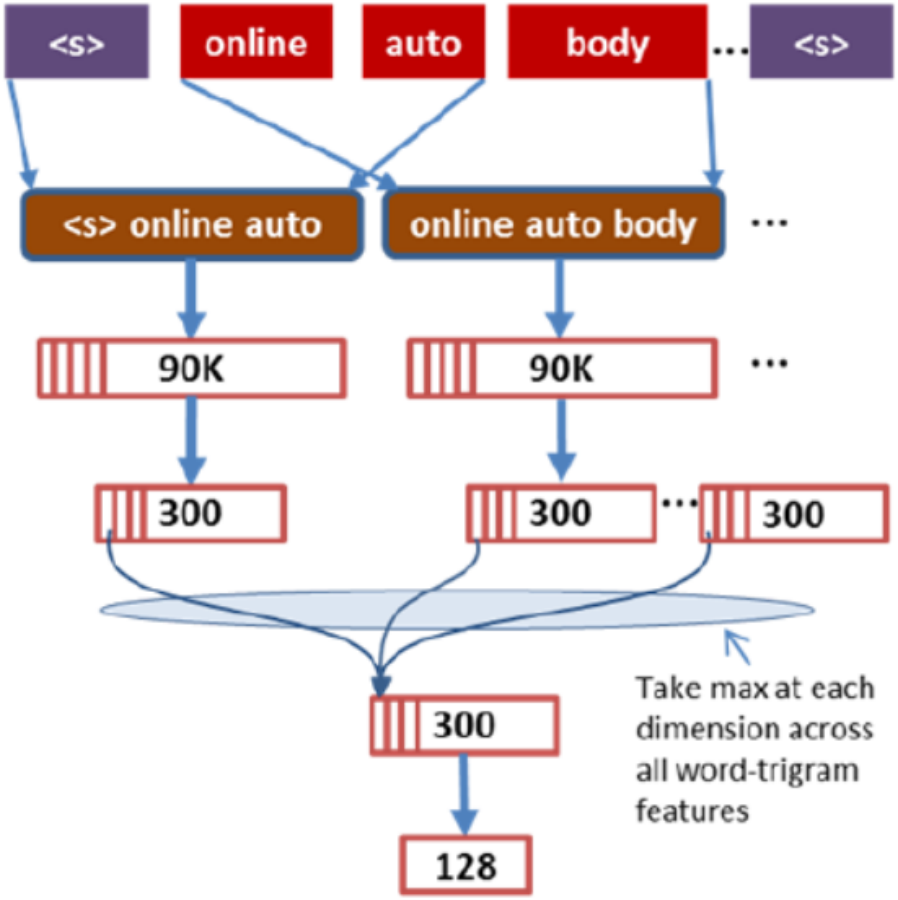
Retrieved



New Query q’ Document d_1 Document d_2

Reference

- Huang, Po-Sen, et al. "Learning deep structured semantic models for web search using clickthrough data." ACM, 2013.
- Shen, Yelong, et al. "A latent semantic model with convolutional-pooling structure for information retrieval." ACM, 2014.

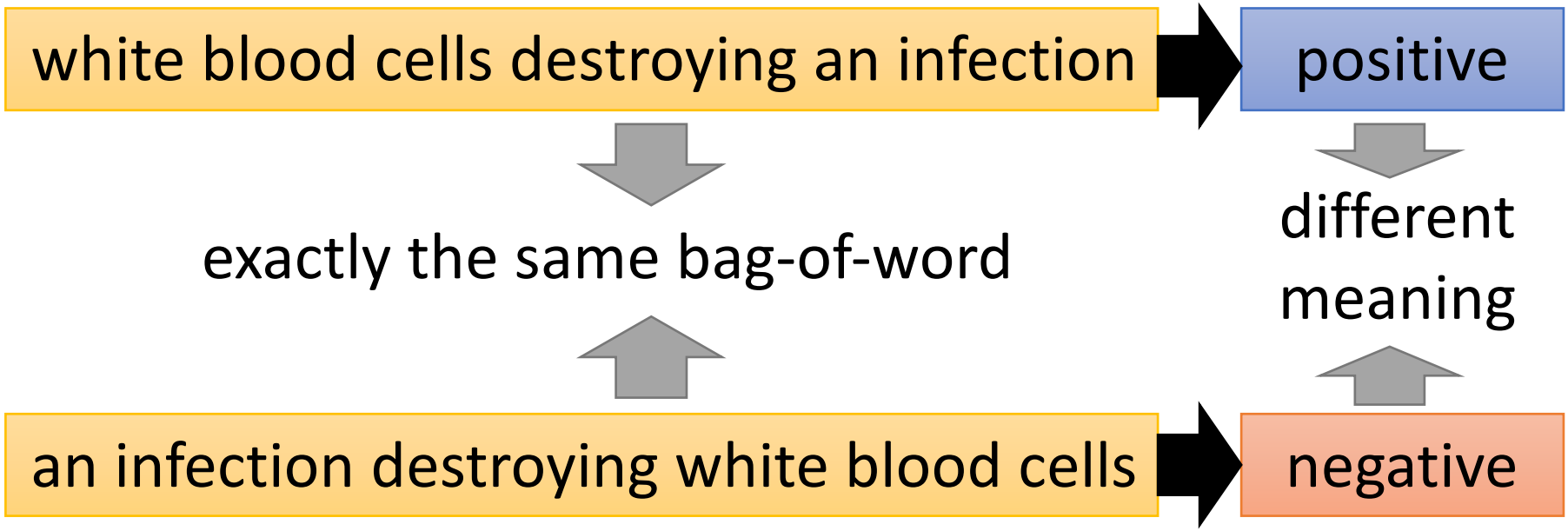


Outline

Deep Structured Semantic Model (DSSM)	• Application: Information Retrieval (IR)
Recursive Neural Network	• Application: Sentiment Analysis, Sentence Relatedness
Unsupervised	• Paragraph Vector • Sequence-to-sequence auto-encoder

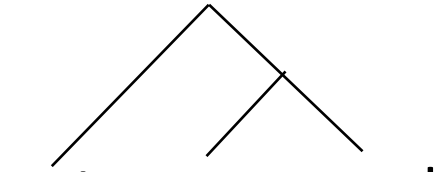
Recursive Deep Model

- To understand the meaning of a word sequence, the order of the words can not be ignored.



Recursive Deep Model

syntactic structure

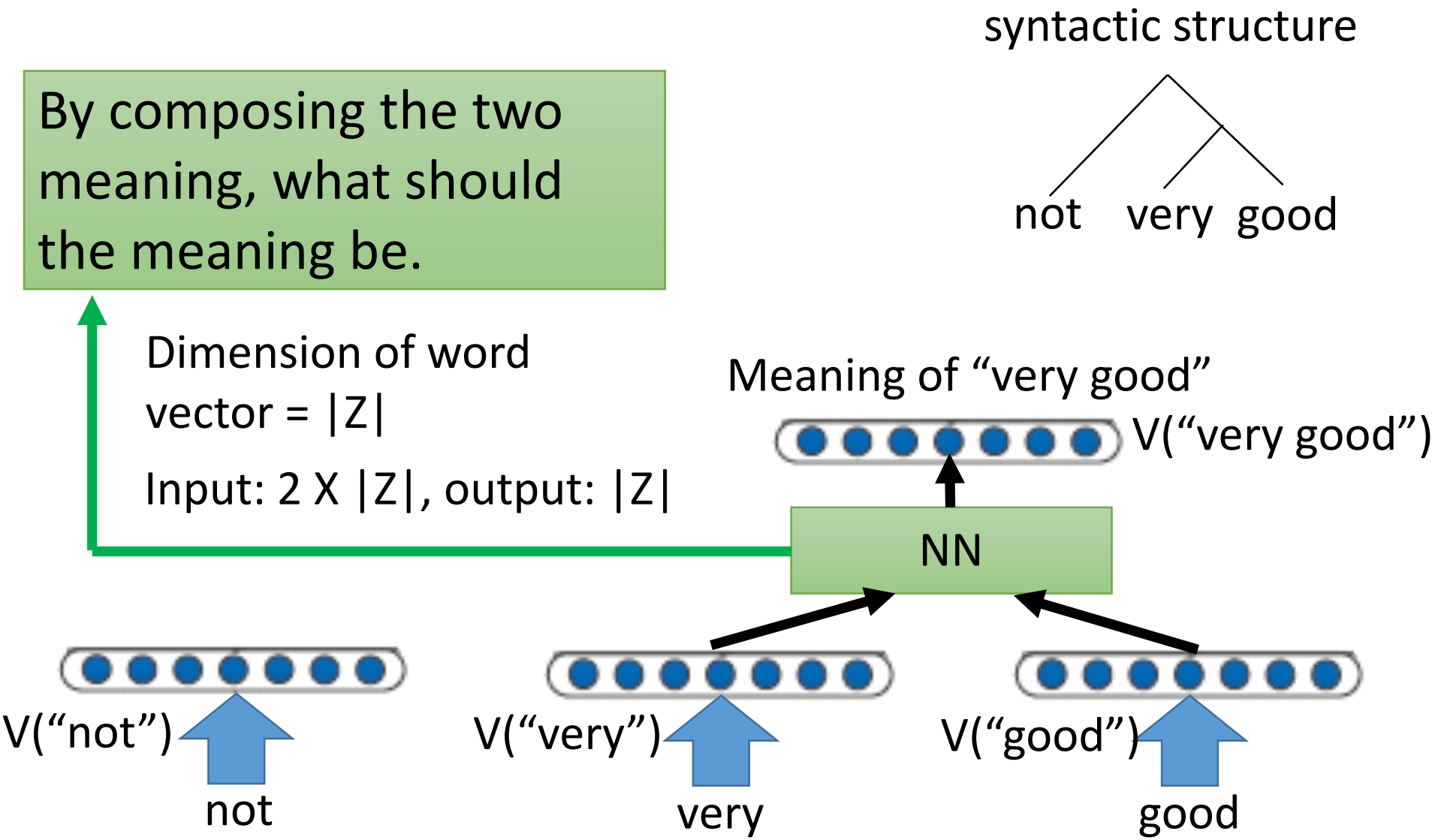


How to do it is out of the scope

word sequence:

not very good

Recursive Deep Model



Recursive Deep Model

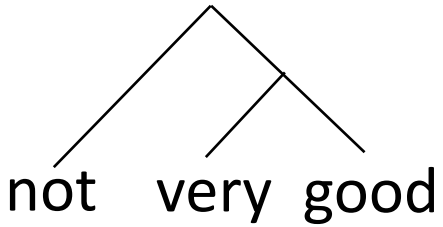
$V(w_A\, w_B) \neq V(w_A) + V(w_B)$

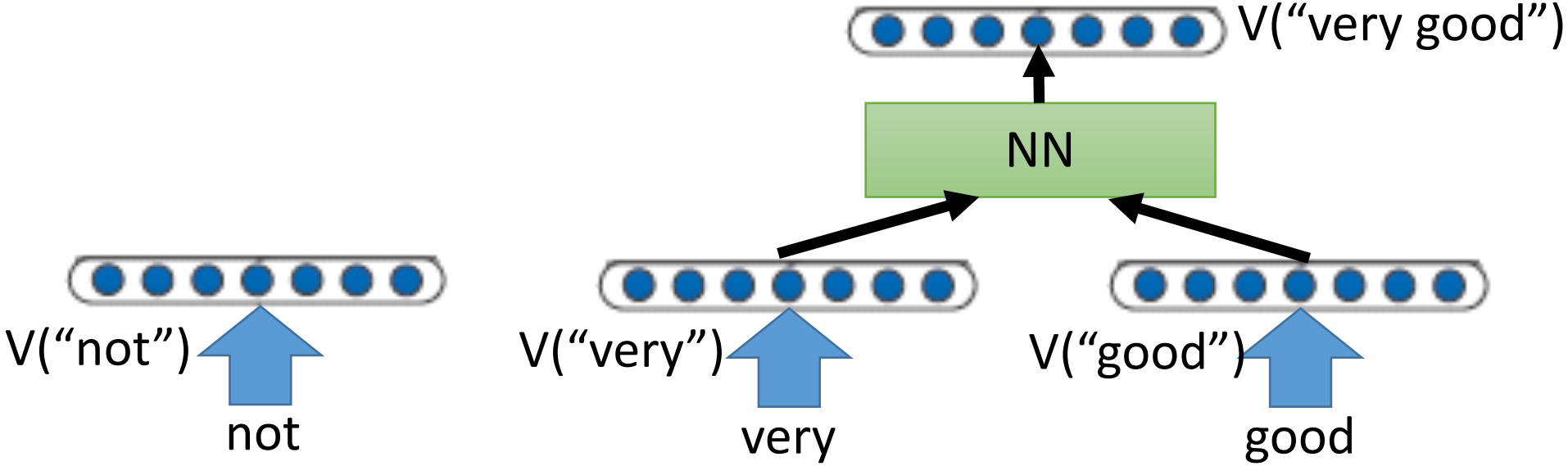
"not": neutral

"good": positive

"not good": negative

syntactic structure





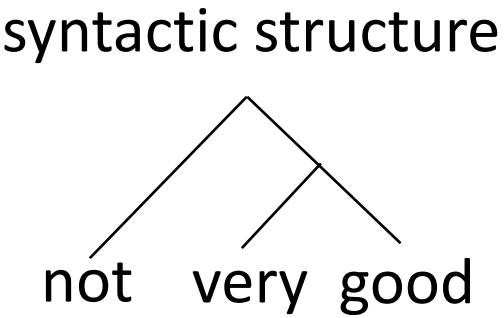
Recursive Deep Model

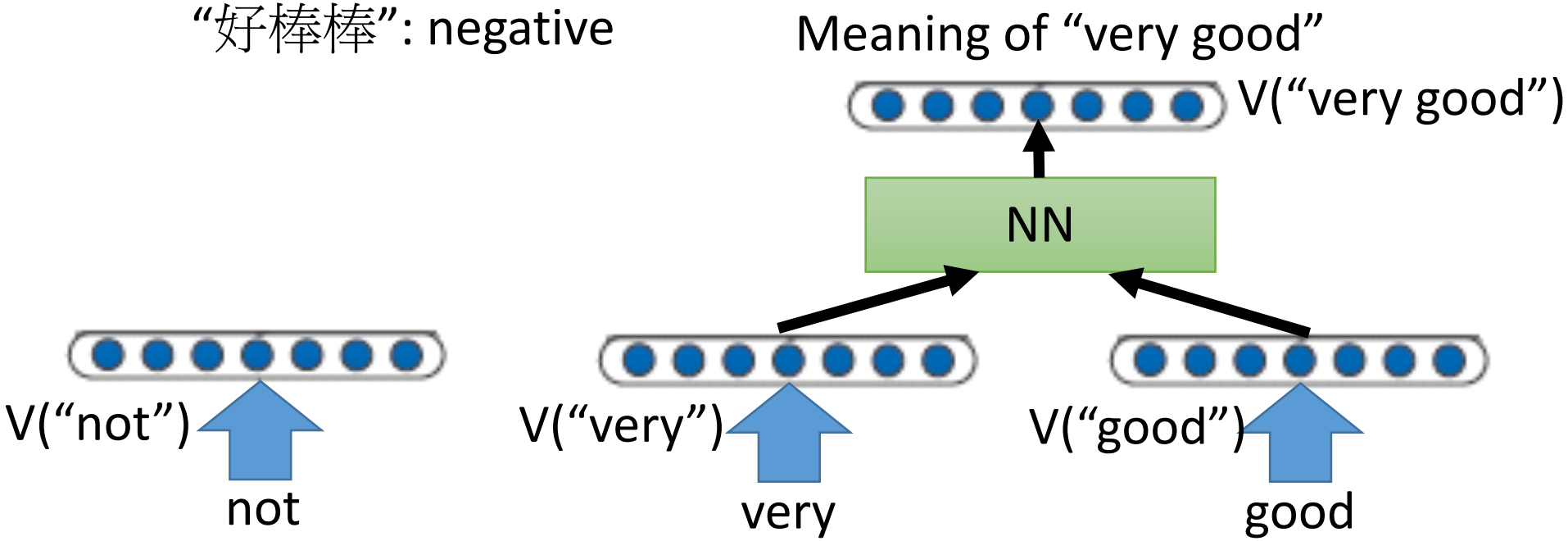
$V(w_A\, w_B) \neq V(w_A) + V(w_B)$

“棒”: positive

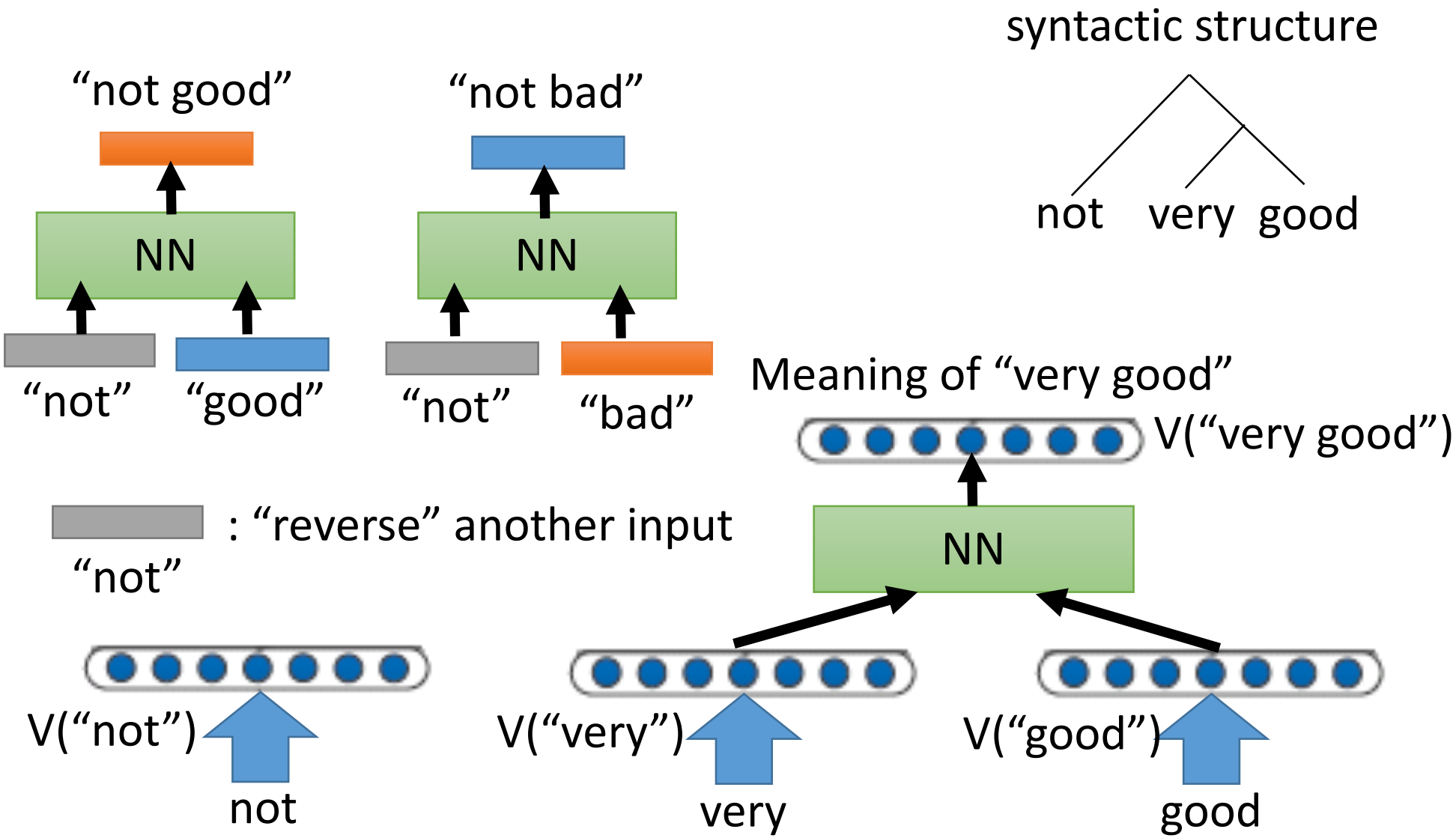
“好棒”: positive

“好棒棒”: negative

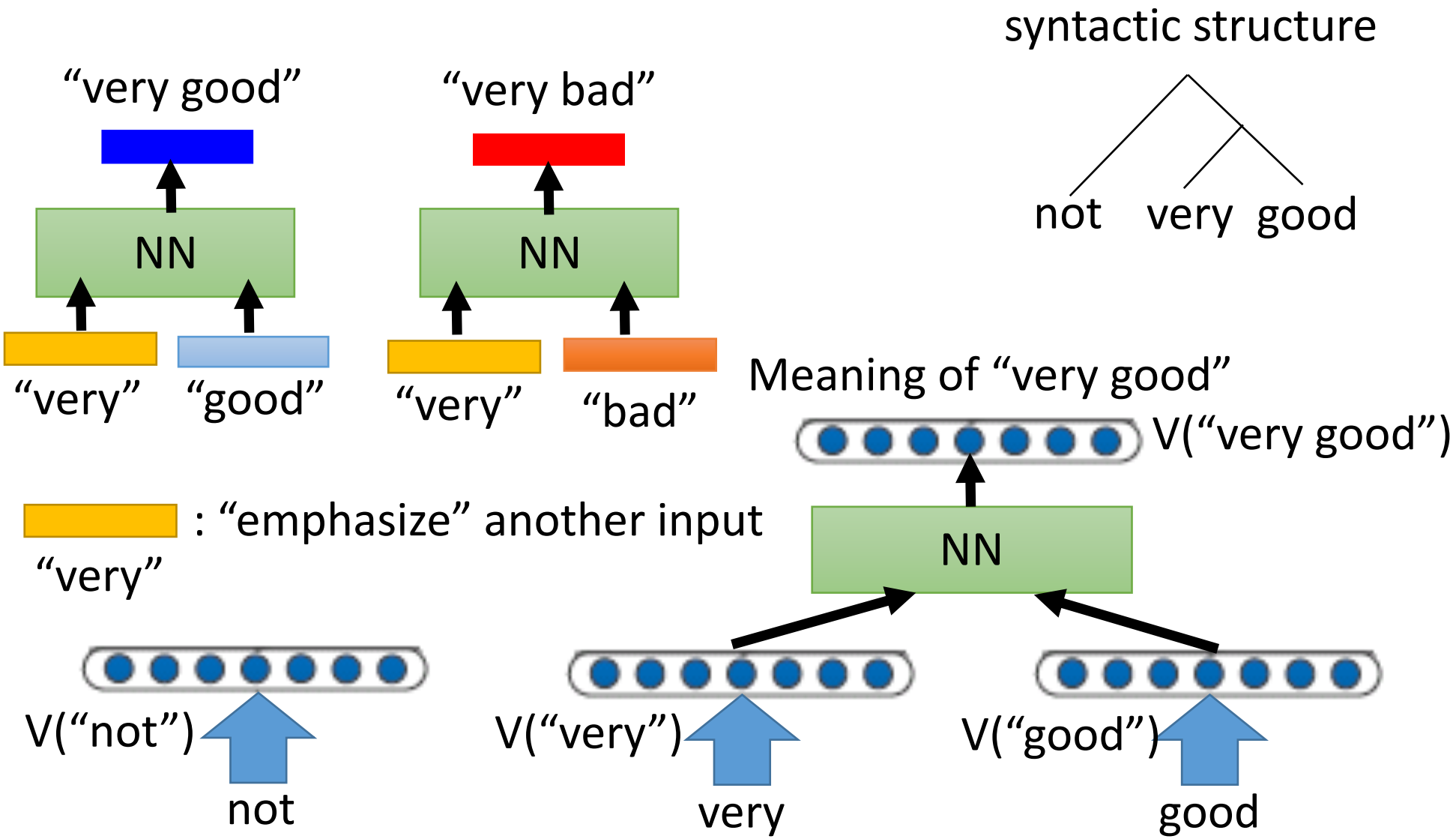




Recursive Deep Model

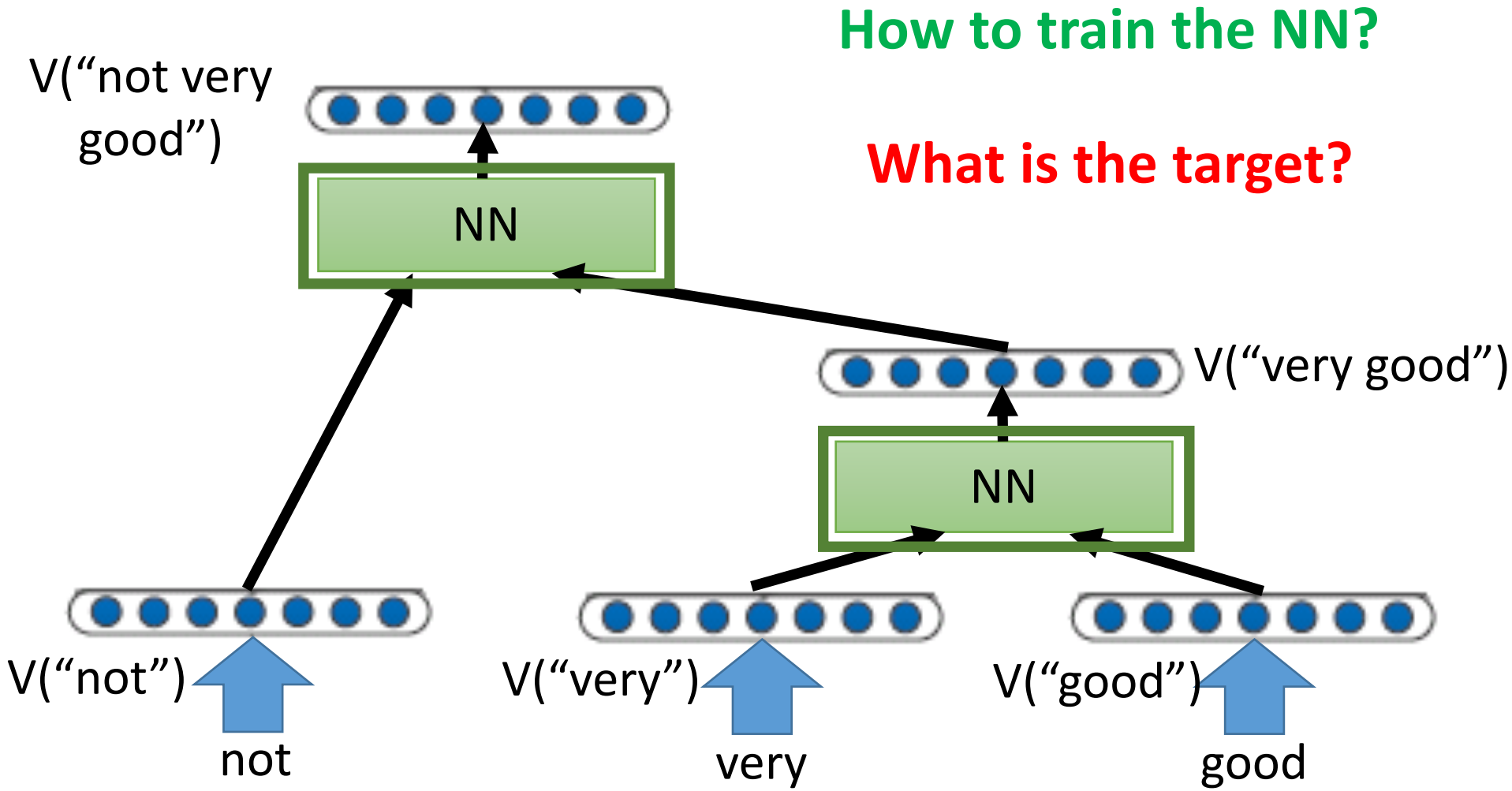


Recursive Deep Model



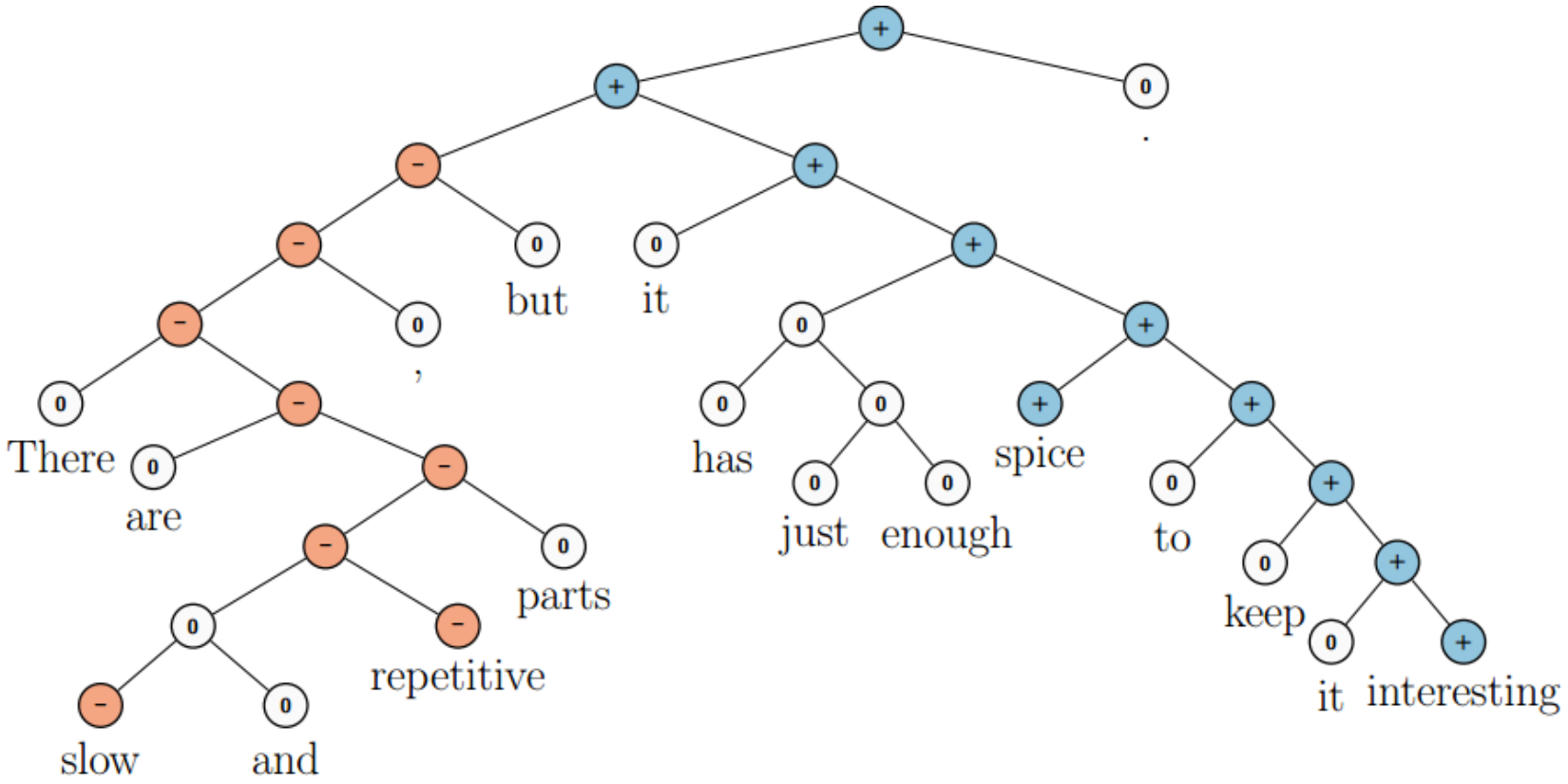
The word order is considered.

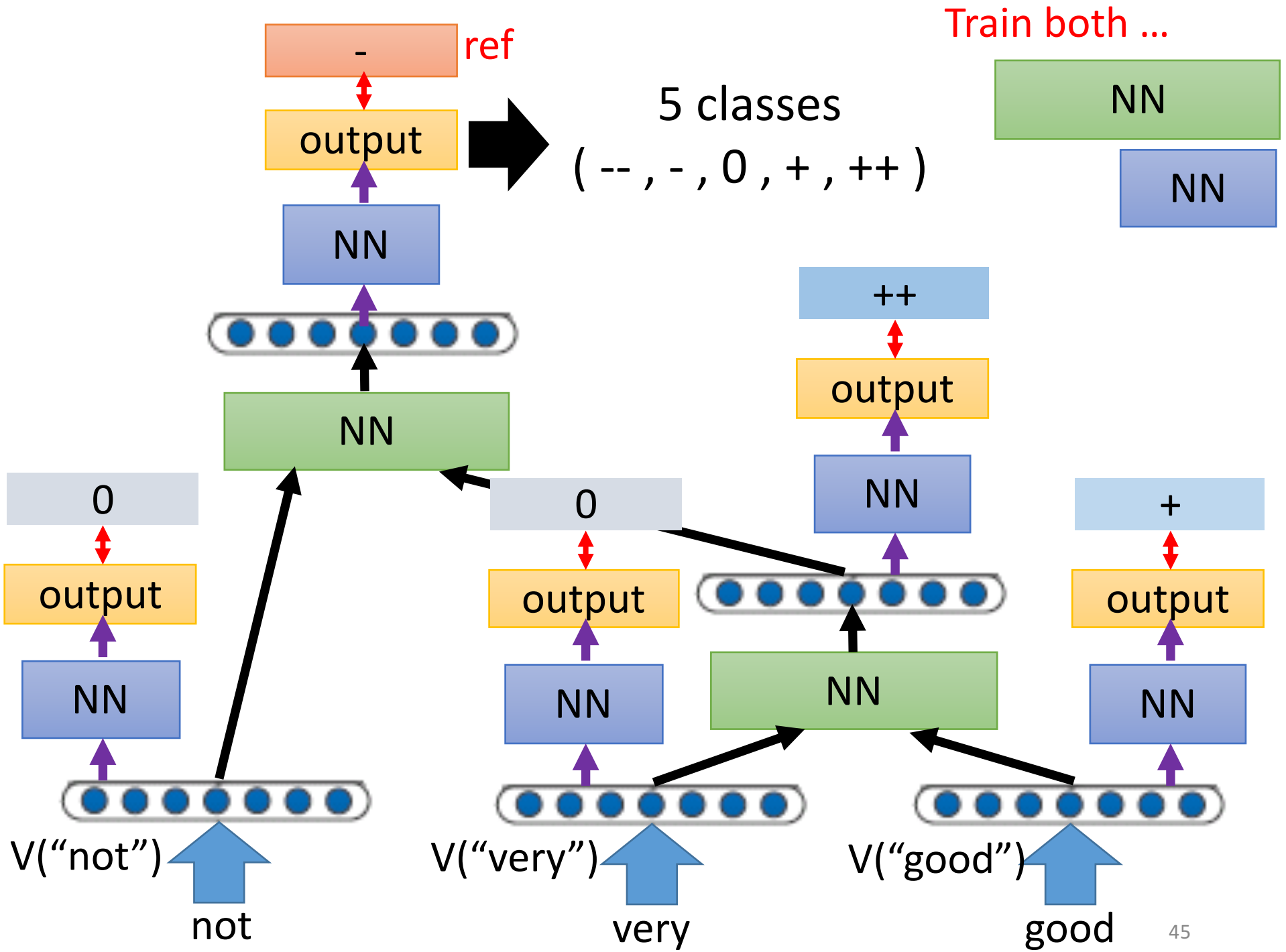
The representation of the sequence will change if the order of the words are changed



Need a Training Target ……

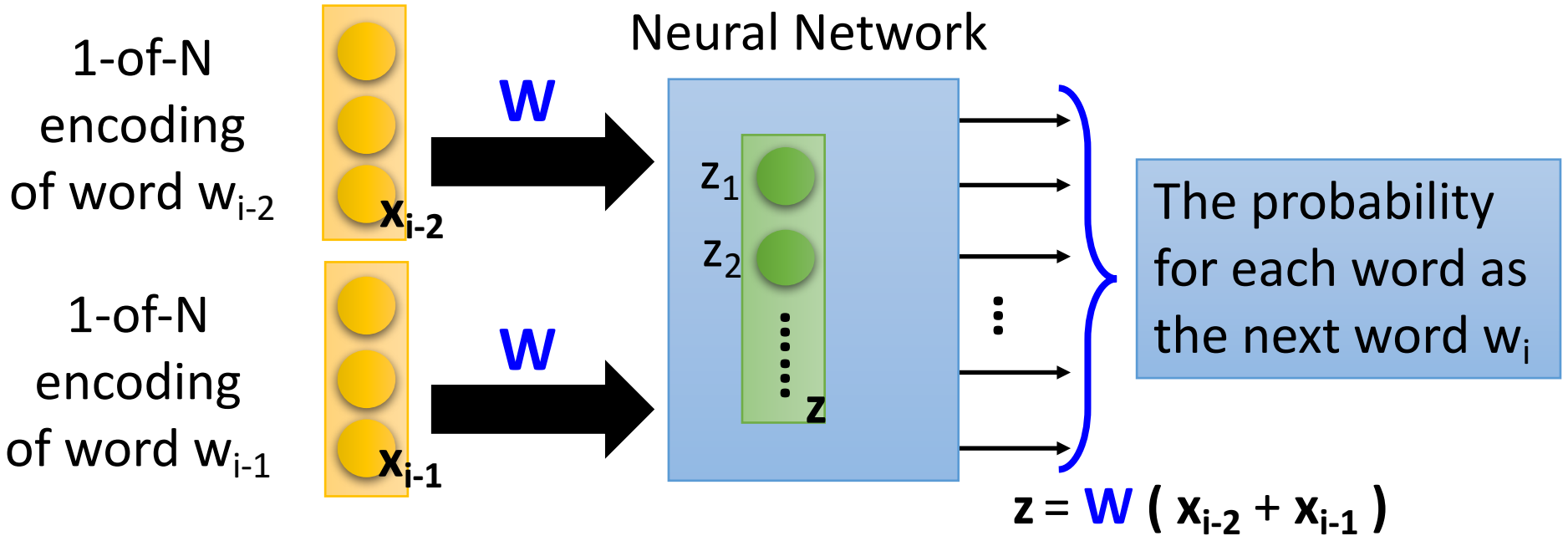
5-class sentiment classification (-- , - , 0 , + , ++)




Train both …
NN
NN
-
ref
output
5 classes
(-- , - , 0 , + , ++)
NN
NN
++
output
NN
0
output
NN
V(“not”)
not
0
output
NN
NN
V(“very”)
very
+
output
NN
V(“good”)
good

Outline

Deep Structured Semantic Model (DSSM)	• Application: Information Retrieval (IR)
Recursive Neural Network	• Application: Sentiment Analysis, Sentence Relatedness
Unsupervised	• Paragraph Vector • Sequence-to-sequence auto-encoder



Paragraph d_1: (The paragraph is from "The lord of the ring")

…… 魔君 名叫 索倫 (Sauron) ……

w_{i-2} w_{i-1} w_i

$z = W (x_{i-2} + x_{i-1})$

Paragraph d_2: (The paragraph is from "仙五")

…… 魔君 名叫 姜世離 ……

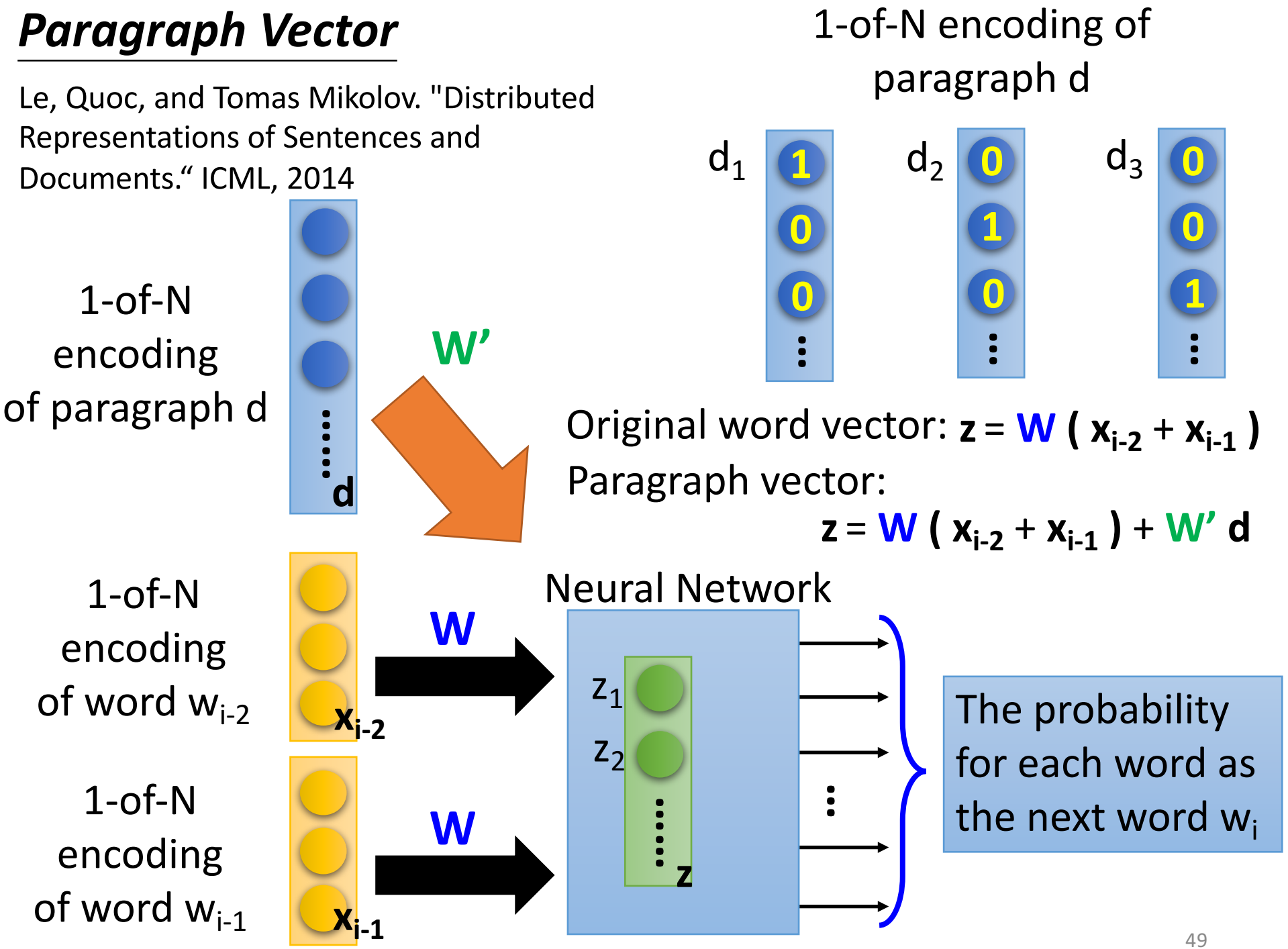
w_{i-2} w_{i-1} w_i

$z = W (x_{i-2} + x_{i-1})$

the same → Same output

Paragraph Vector

Le, Quoc, and Tomas Mikolov. "Distributed Representations of Sentences and Documents." ICML, 2014



Paragraph Vector

Le, Quoc, and Tomas Mikolov. "Distributed Representations of Sentences and Documents." ICML, 2014

Original word vector:

$$z = W (x_{i-2} + x_{i-1})$$

Paragraph vector:

$$z = W (x_{i-2} + x_{i-1}) + W' d$$

Then error of the prediction can be explained by the meaning of the paragraphs.

Paragraph d_1: (The paragraph is related to "The lord of the ring")

…… 魔君 名叫 索倫 (Sauron) ……

w_{i-2} w_{i-1} w_i

$z = W (x_{i-2} + x_{i-1}) + W' d_1$

different

Paragraph d_2: (The document is related to "仙五")

…… 魔君 名叫 姜世離 ……

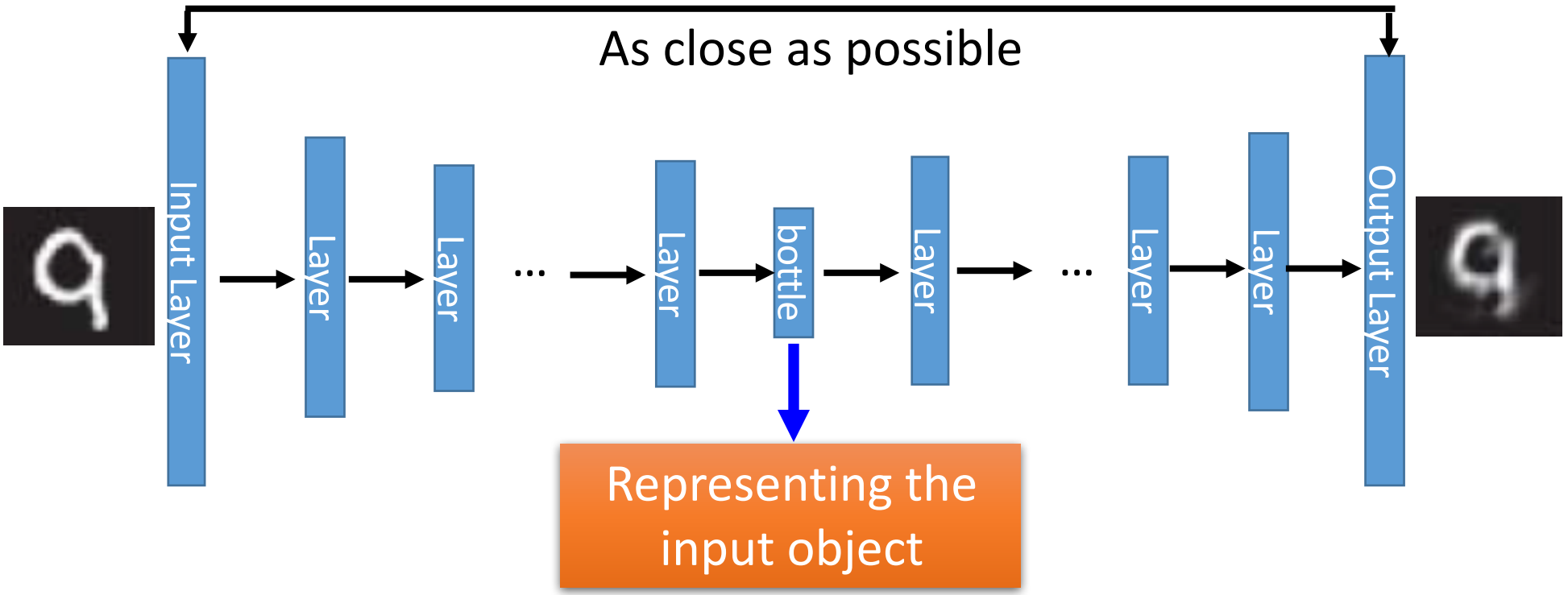
w_{i-2} w_{i-1} w_i

$z = W (x_{i-2} + x_{i-1}) + W' d_2$

Paragraph vector of d:V(d) = $W' d$ ➡ Meaning of the paragraph

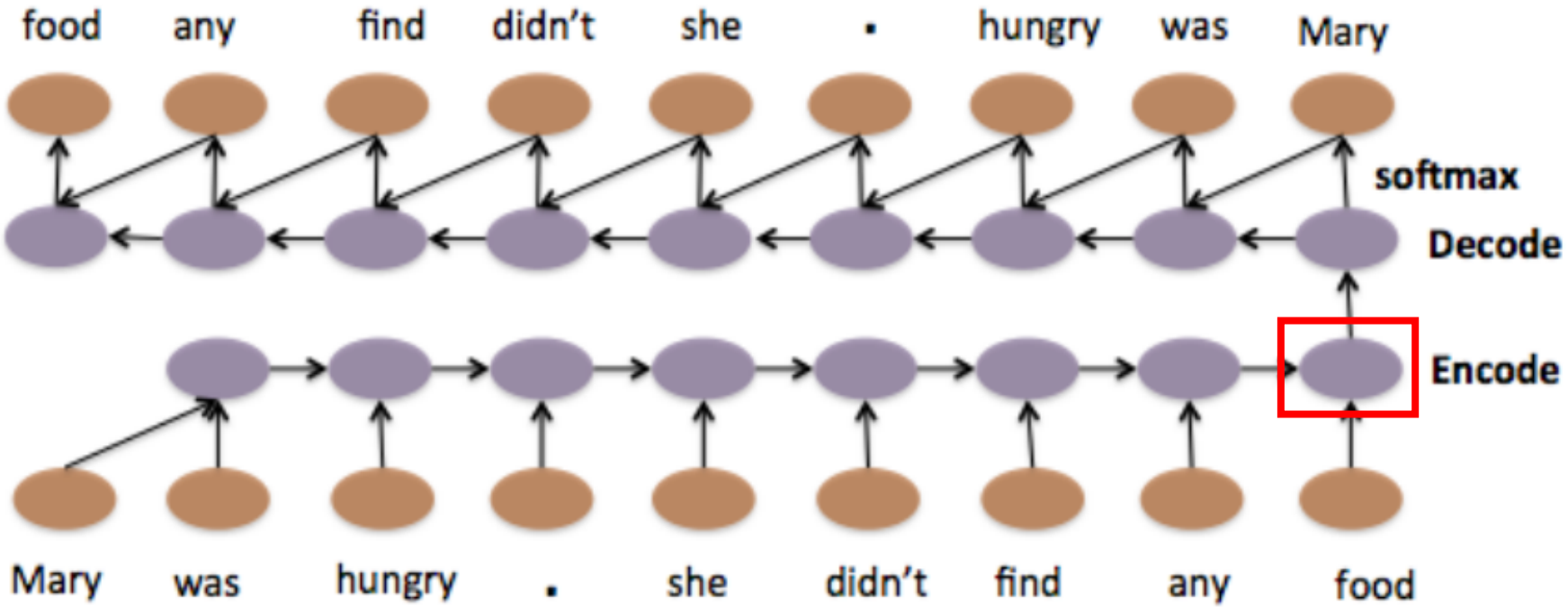
Sequence-to-sequence Auto-encoder

- Original Auto-encoder



Reference: Hinton, Geoffrey E., and Ruslan R. Salakhutdinov. "Reducing the dimensionality of data with neural networks." *Science* 313.5786 (2006): 504-507

Sequence-to-sequence Auto-encoder



Li, Jiwei, Minh-Thang Luong, and Dan Jurafsky. "A hierarchical neural autoencoder for paragraphs and documents." *arXiv preprint arXiv:1506.01057*(2015).

Summary

Deep Structured Semantic Model (DSSM)	• Application: Information Retrieval (IR)
Recursive Neural Network	• Application: Sentiment Analysis, Sentence Relatedness
Unsupervised	• Paragraph Vector • Sequence-to-sequence auto-encoder

Tools

- 中文斷詞
  - Jieba
- 英文stopwords
  - NLTK stopwords
- Easiest way to use word2vec is via the Gensim libarary for Python (tends to be slowish, even though it tries to use C optimizations like Cython, NumPy)

https://radimrehurek.com/gensim/models/word2vec.html

FastText

- Library for fast text representation and classification
  - Courtesy of Facebook Research
- Recent state-of-the-art English word vectors.
- Word vectors for 157 languages trained on Wikipedia and Crawl.
- Models for language identification and various supervised tasks.

- https://github.com/facebookresearch/fastText?utm_source=mybridge&utm_medium=blog&utm_campaign=read_more

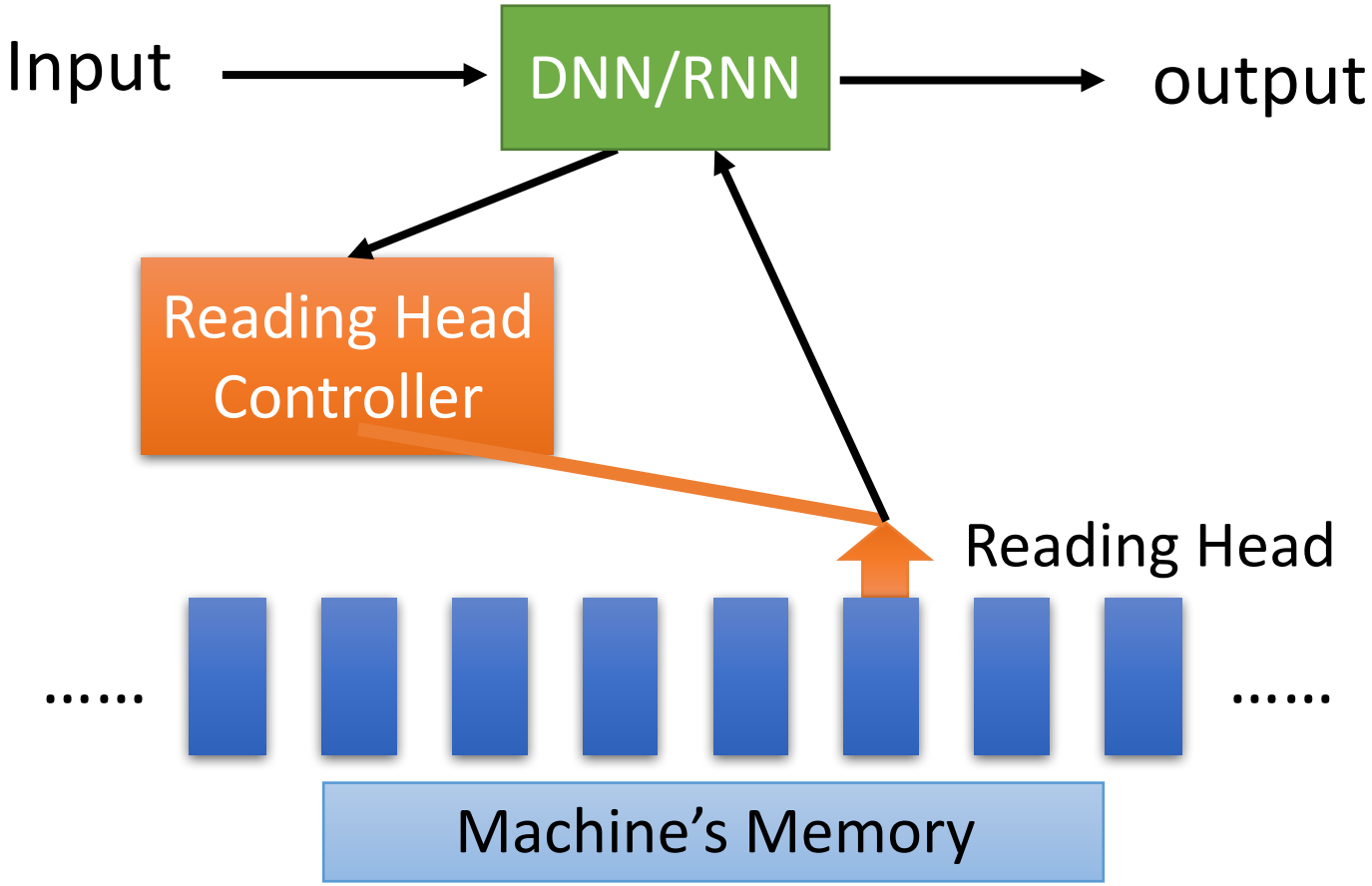
Fairseq

- sequence-to-sequence learning toolkit for Torch from Facebook AI Research
- English to French, English to German and English to Romanian translation

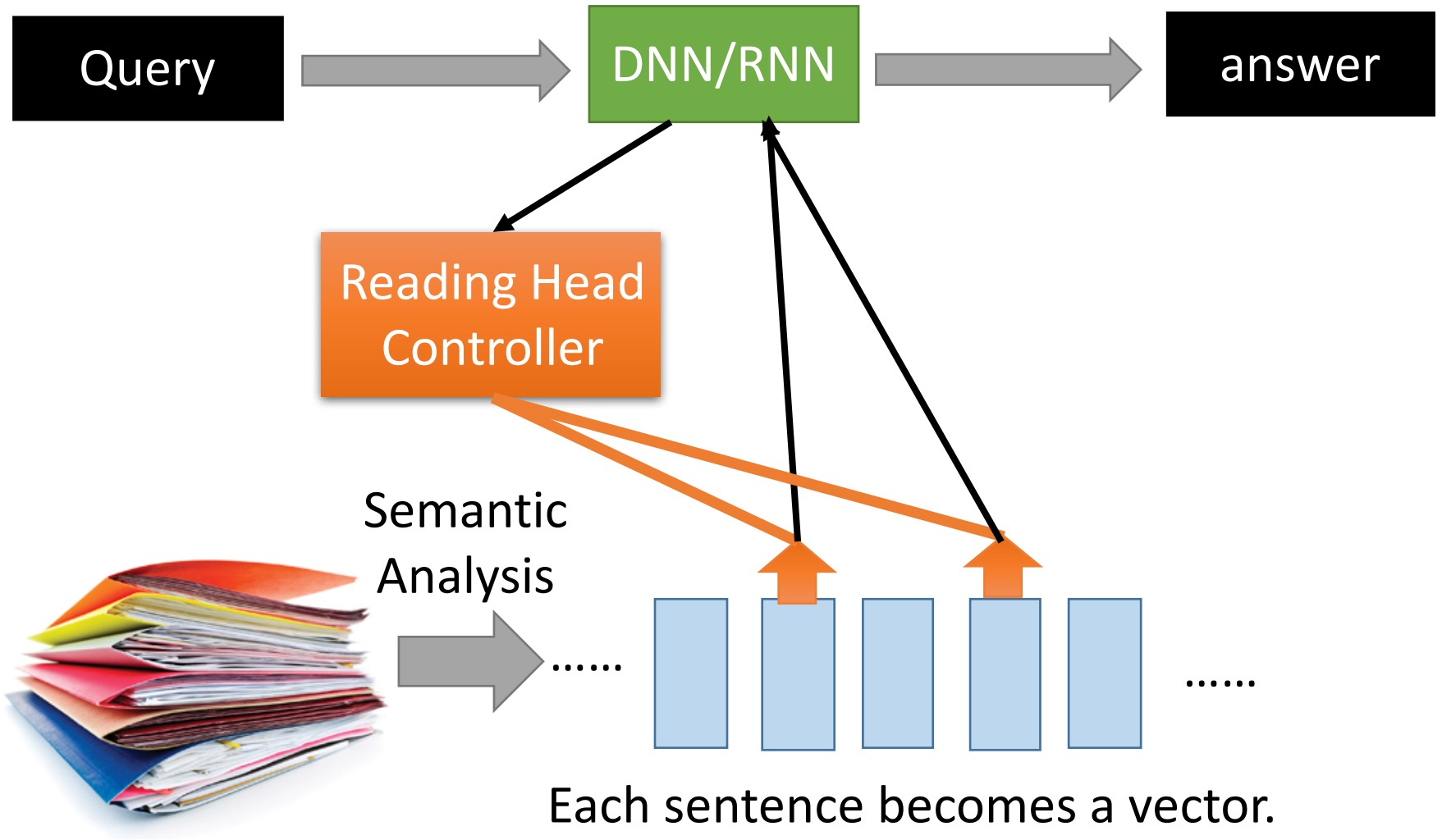
https://github.com/facebookresearch/fairseq?utm_source=mybridge&utm_medium=blog

Attention-based Model

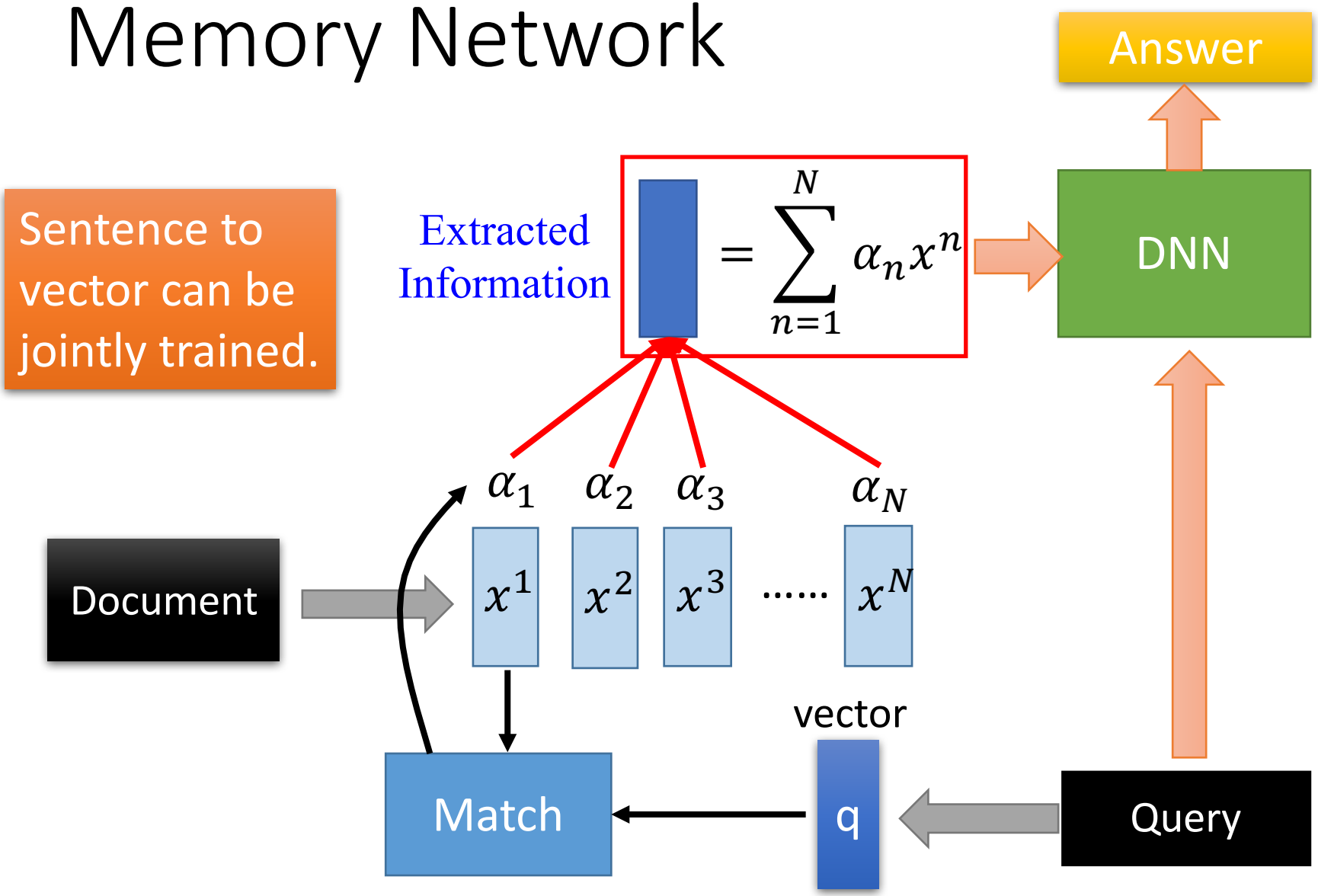
External Memory



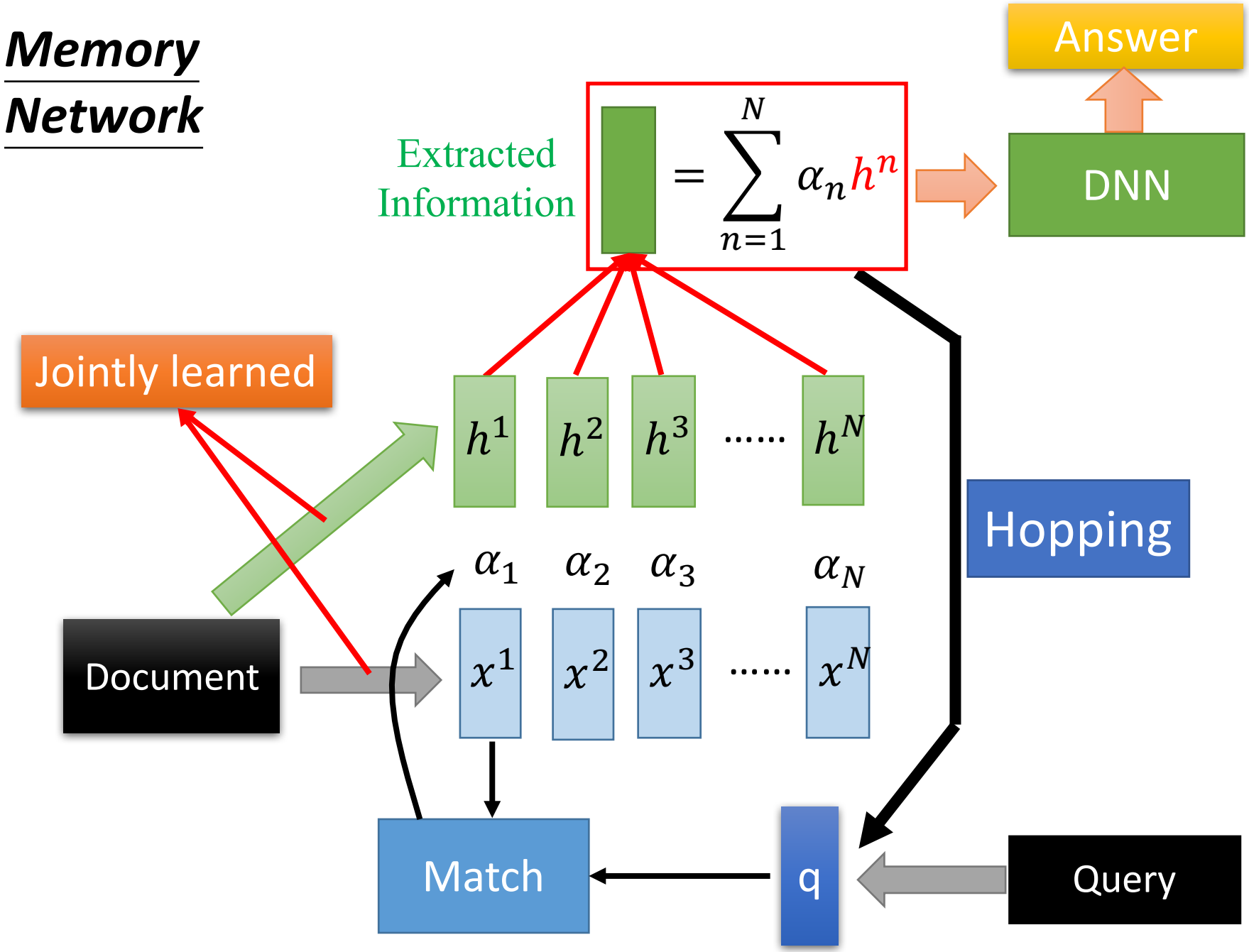
Reading Comprehension

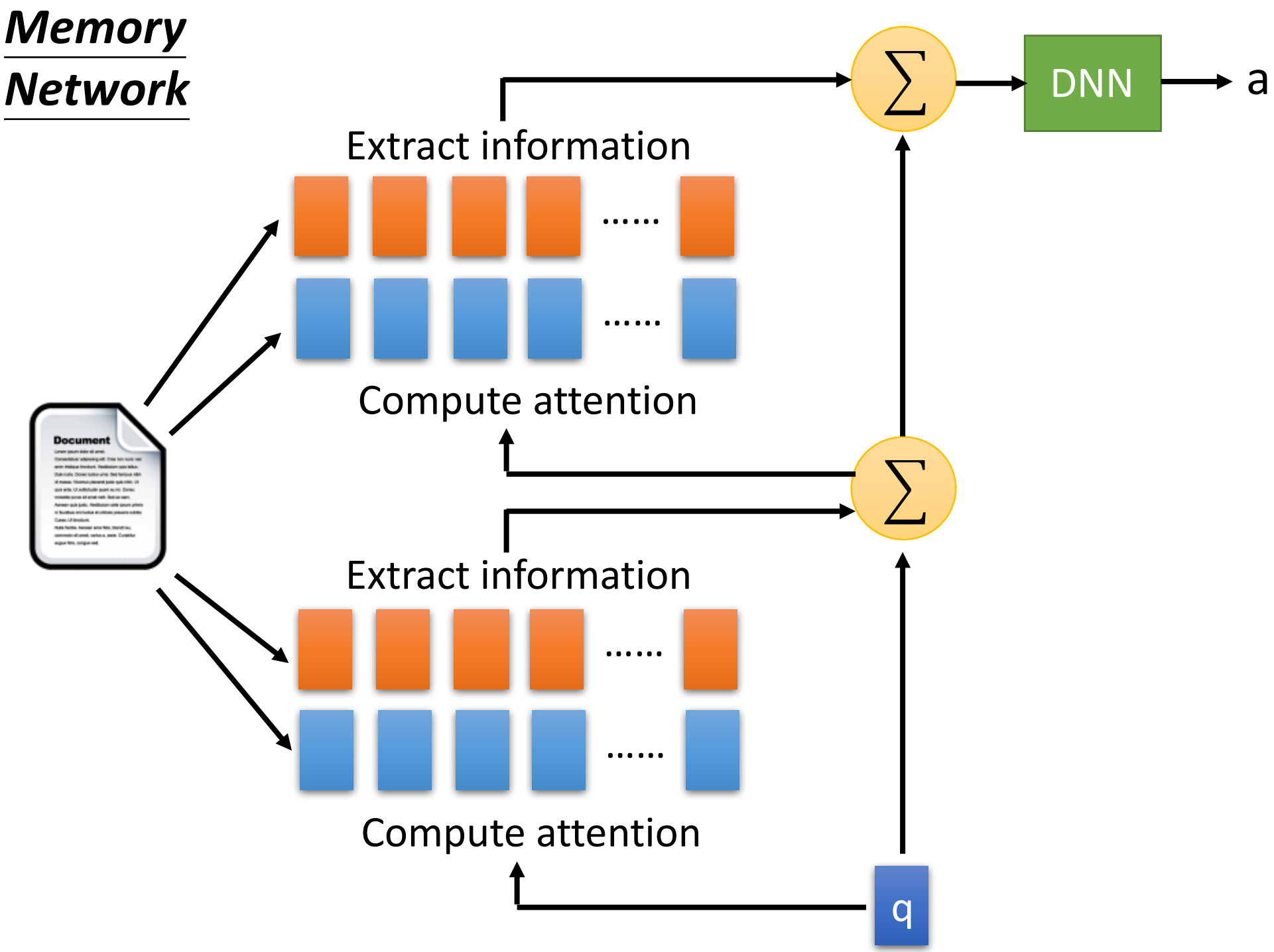


Memory Network



Sainbayar Sukhbaatar, Arthur Szlam, Jason Weston, Rob Fergus, "End-To-End Memory Networks", NIPS, 2015


Memory Network
Answer
DNN
Extracted Information
$= \sum_{n=1}^{N} \alpha_n h^n$
Jointly learned
h^1 h^2 h^3 ..... h^N
Hopping
α_1 α_2 α_3 α_N
Document
x^1 x^2 x^3 ..... x^N
Match
q
Query


Memory Network
Extract information
Compute attention
Extract information
Compute attention
Document
DNN
a
q

Multiple-hop

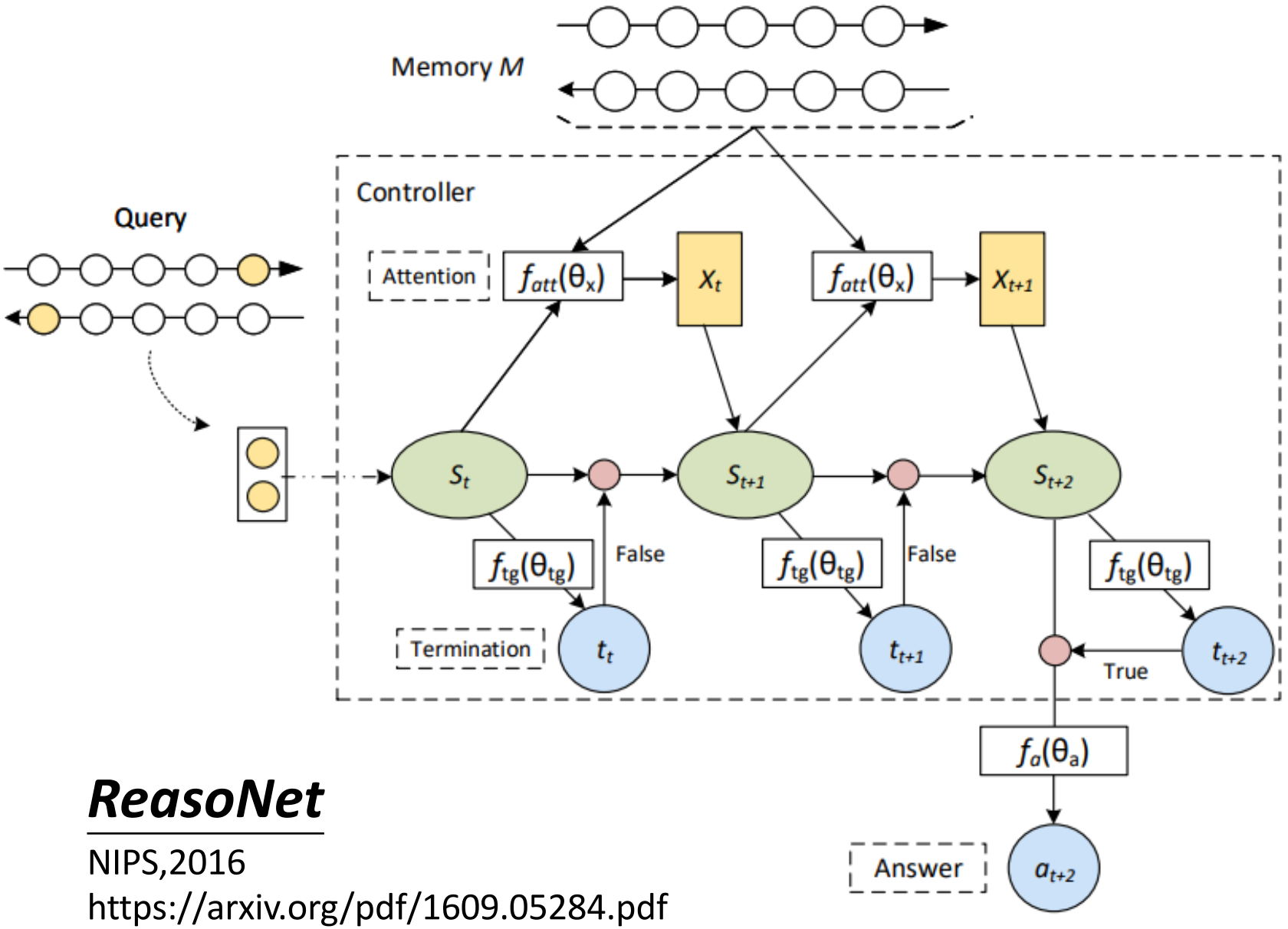
- End-To-End Memory Networks. S. Sukhbaatar, A. Szlam, J. Weston, R. Fergus. NIPS, 2015.

The position of reading head:

Story (16: basic induction)	Support	Hop 1	Hop 2	Hop 3
Brian is a frog.	yes	0.00	0.98	0.00
Lily is gray.		0.07	0.00	0.00
Brian is yellow.	yes	0.07	0.00	1.00
Julius is green.		0.06	0.00	0.00
Greg is a frog.	yes	0.76	0.02	0.00
What color is Greg? Answer: yellow	**Prediction: yellow**			

Keras has example:
https://github.com/fchollet/keras/blob/master/examples/babi_memnn.py

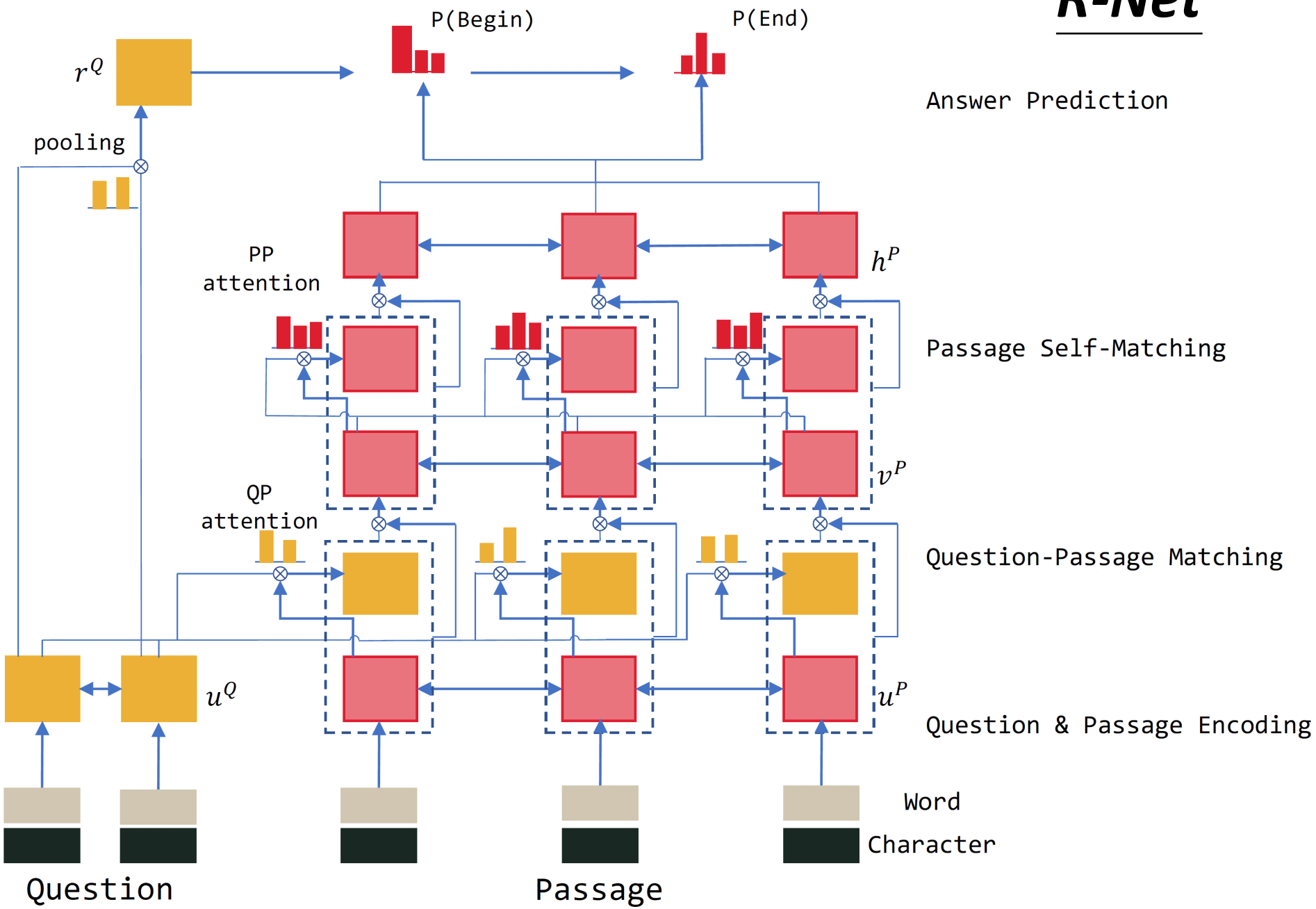
Multiple-hop

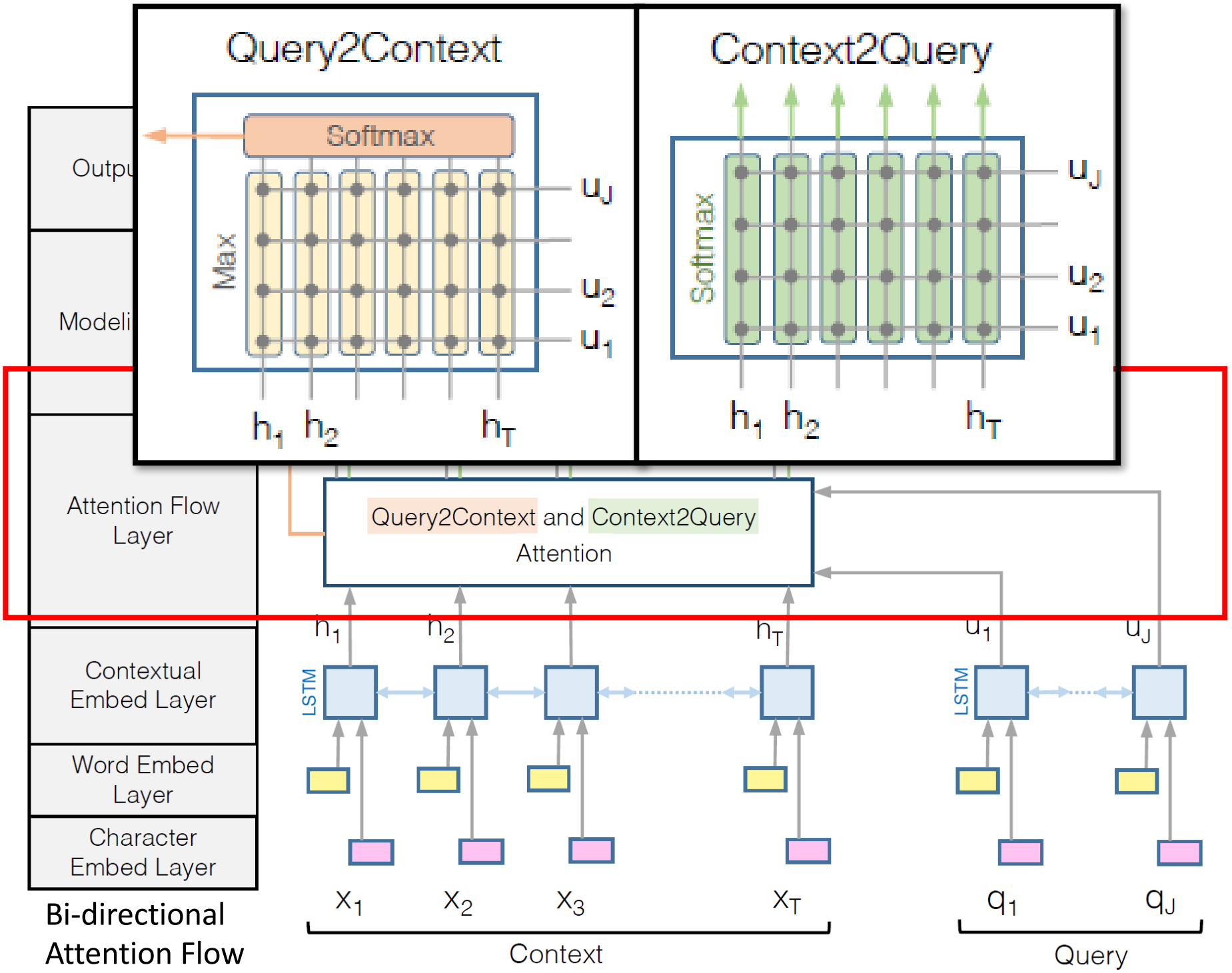


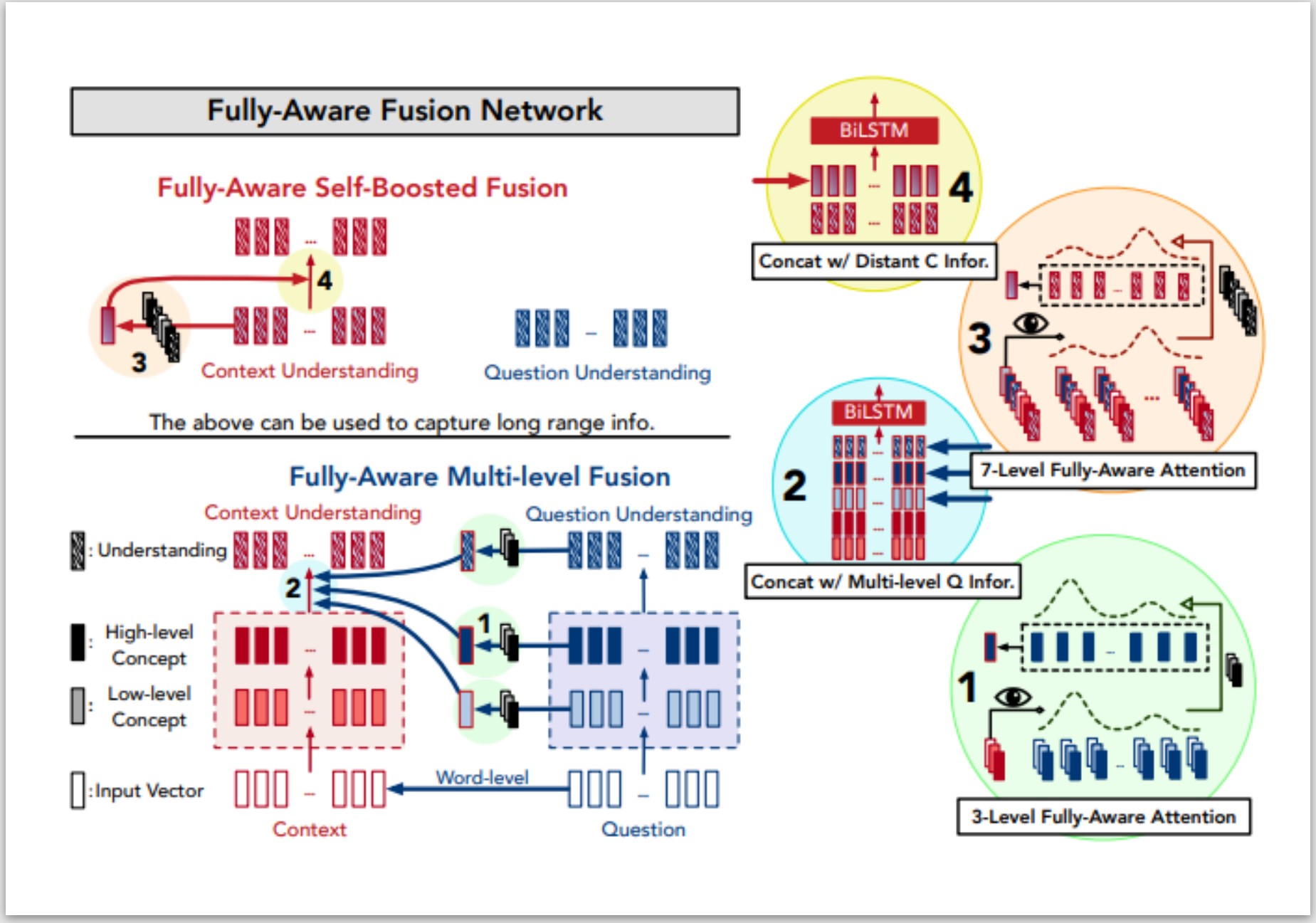
ReasoNet

NIPS,2016

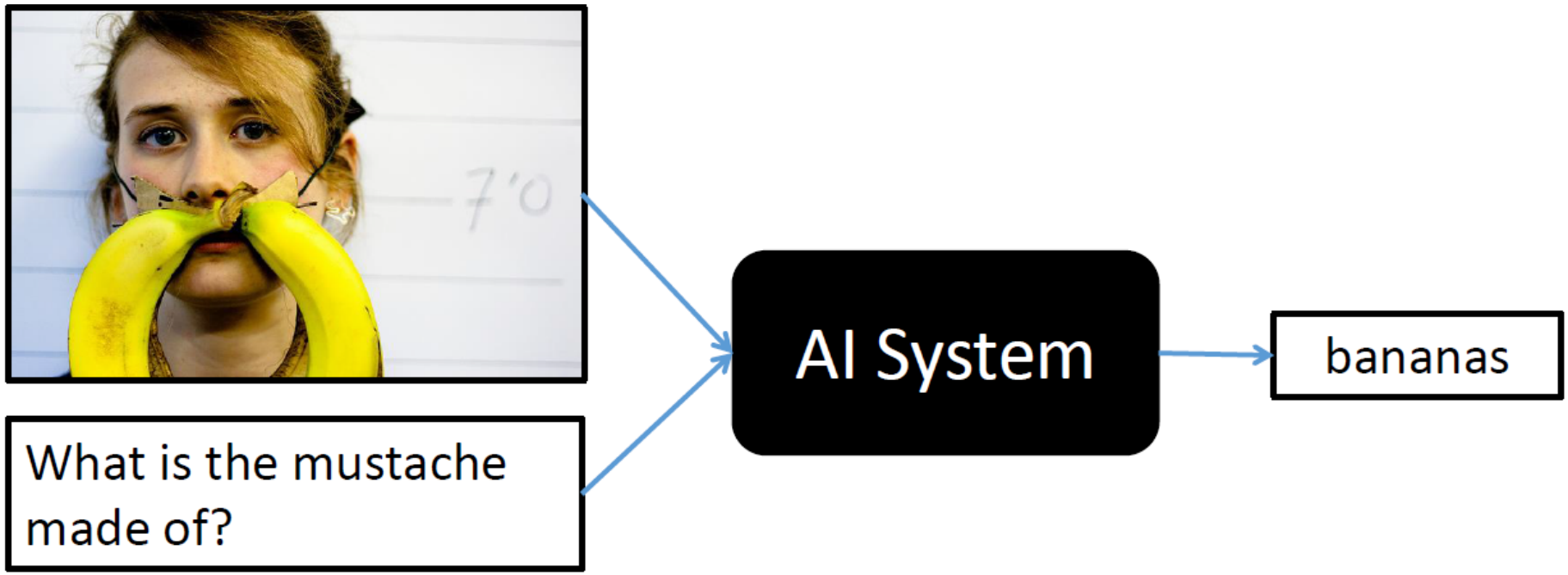
https://arxiv.org/pdf/1609.05284.pdf


R-Net
P(Begin)
P(End)
r^Q
pooling
Answer Prediction
PP
attention
h^P
Passage Self-Matching
v^P
QP
attention
Question-Passage Matching
u^Q
u^P
Question & Passage Encoding
Word
Character
Question
Passage


Query2Context
Softmax
Max
u_J
u_2
u_1
h_1 h_2 h_T
Context2Query
Softmax
u_J
u_2
u_1
h_1 h_2 h_T
Outpu
Modeli
Attention Flow Layer
Query2Context and Context2Query Attention
Contextual Embed Layer
Word Embed Layer
Character Embed Layer
LSTM
LSTM
h_1 h_2 h_T
u_1 u_J
x_1 x_2 x_3 x_T
q_1 q_J
Context
Query
Bi-directional Attention Flow

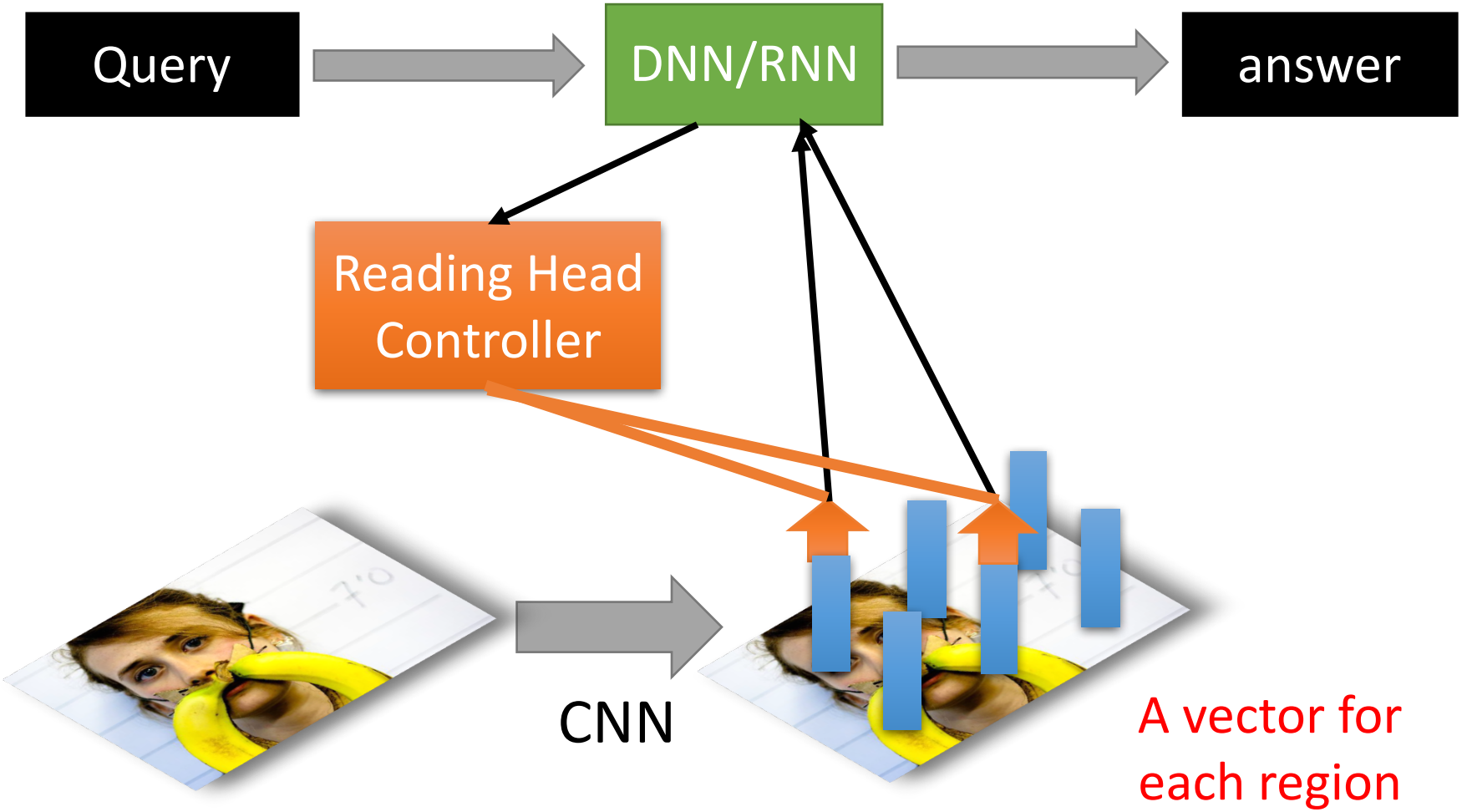

Fully-Aware Fusion Network
Fully-Aware Self-Boosted Fusion
4
3
Context Understanding
Question Understanding
The above can be used to capture long range info.
Fully-Aware Multi-level Fusion
Context Understanding
Question Understanding
: Understanding
2
1
: High-level Concept
: Low-level Concept
:Input Vector
Word-level
Context
Question
BiLSTM
4
Concat w/ Distant C Infor.
3
7-Level Fully-Aware Attention
BiLSTM
2
Concat w/ Multi-level Q Infor.
1
3-Level Fully-Aware Attention

Visual Question Answering



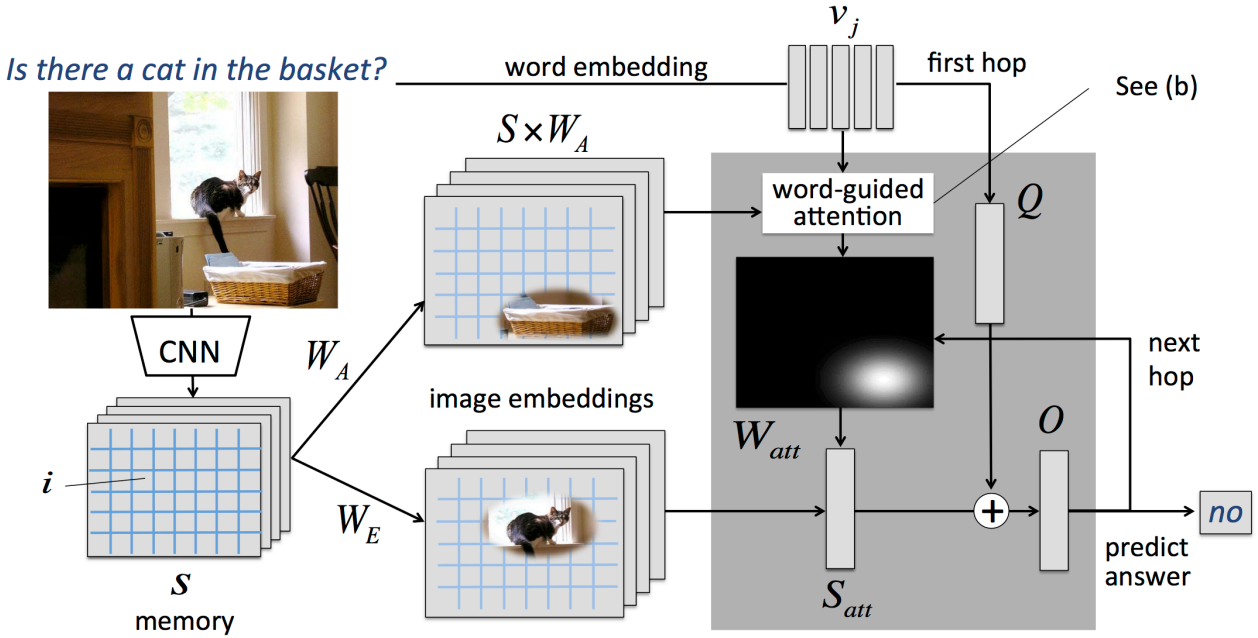
source: http://visualqa.org/

Visual Question Answering

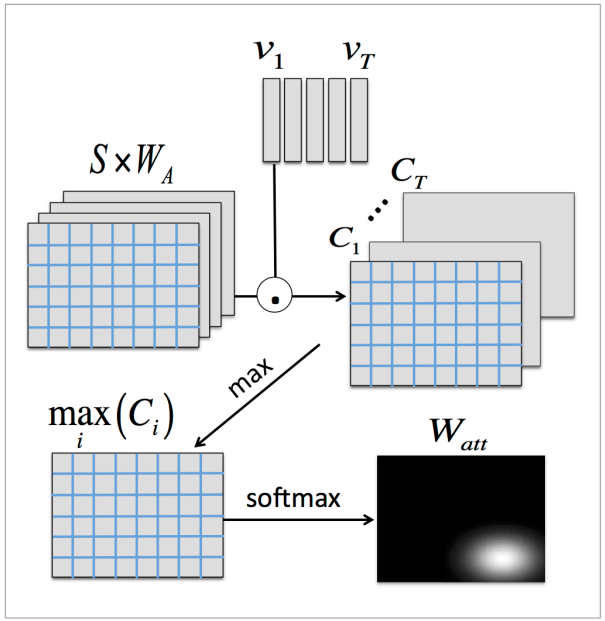


Visual Question Answering

- Huijuan Xu, Kate Saenko. Ask, Attend and Answer: Exploring Question-Guided Spatial Attention for Visual Question Answering. arXiv Pre-Print, 2015



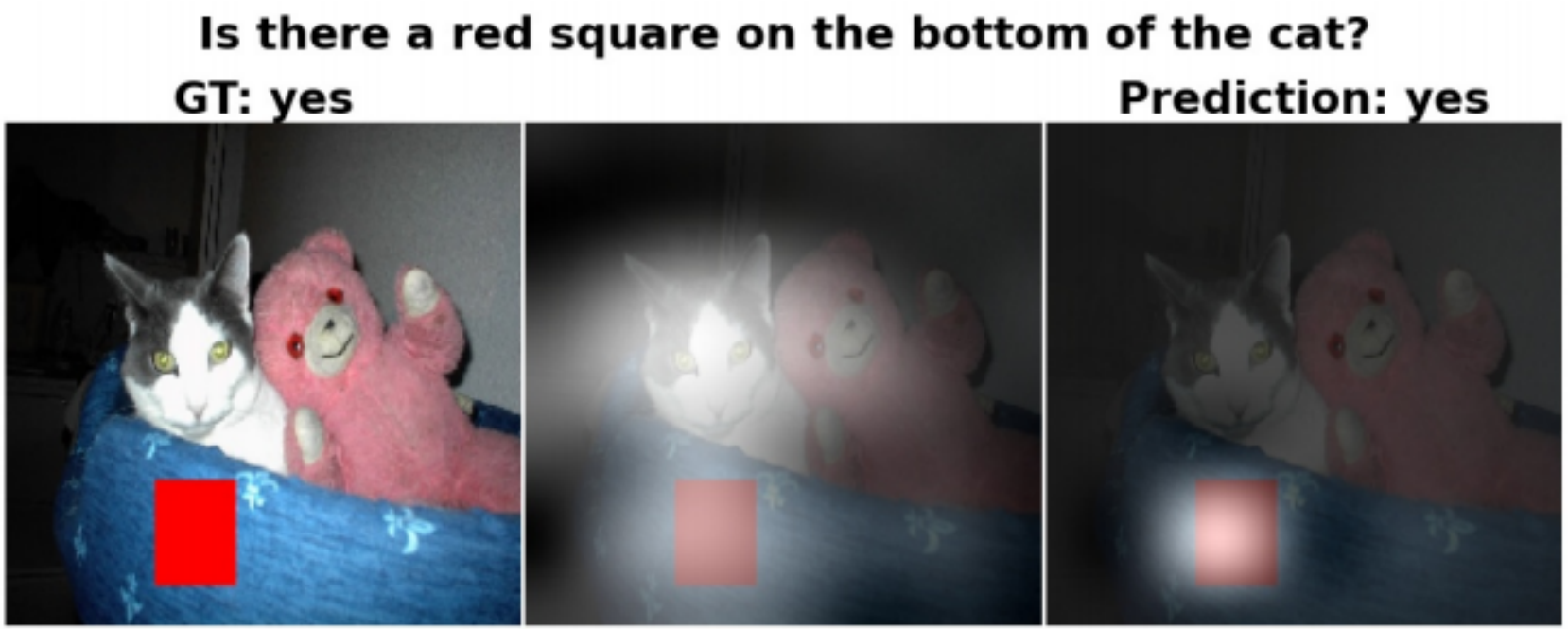
(a) Overview



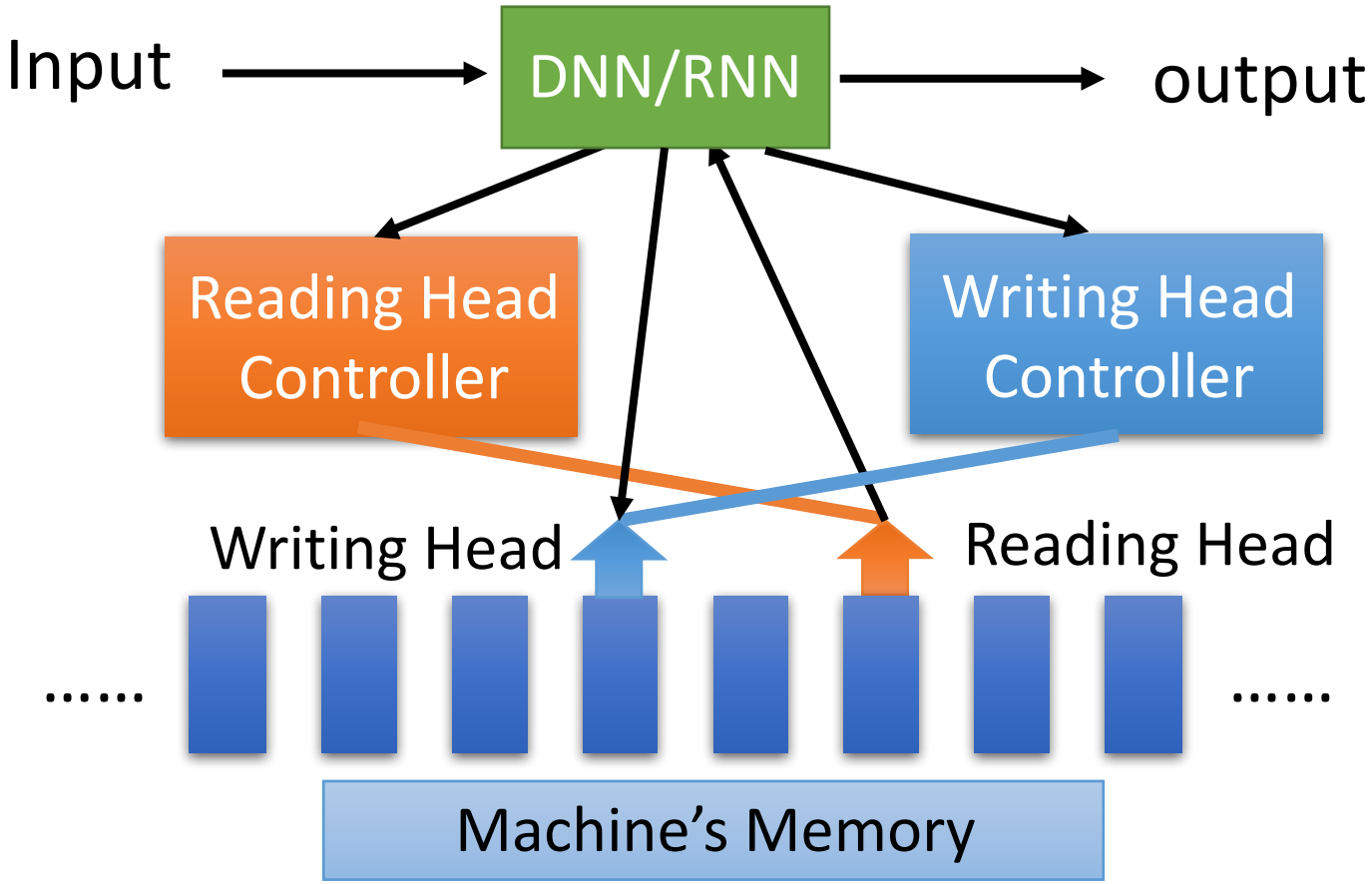
(b) Word-guided attention

Visual Question Answering

- Huijuan Xu, Kate Saenko. Ask, Attend and Answer: Exploring Question-Guided Spatial Attention for Visual Question Answering. ECCV, 2016.

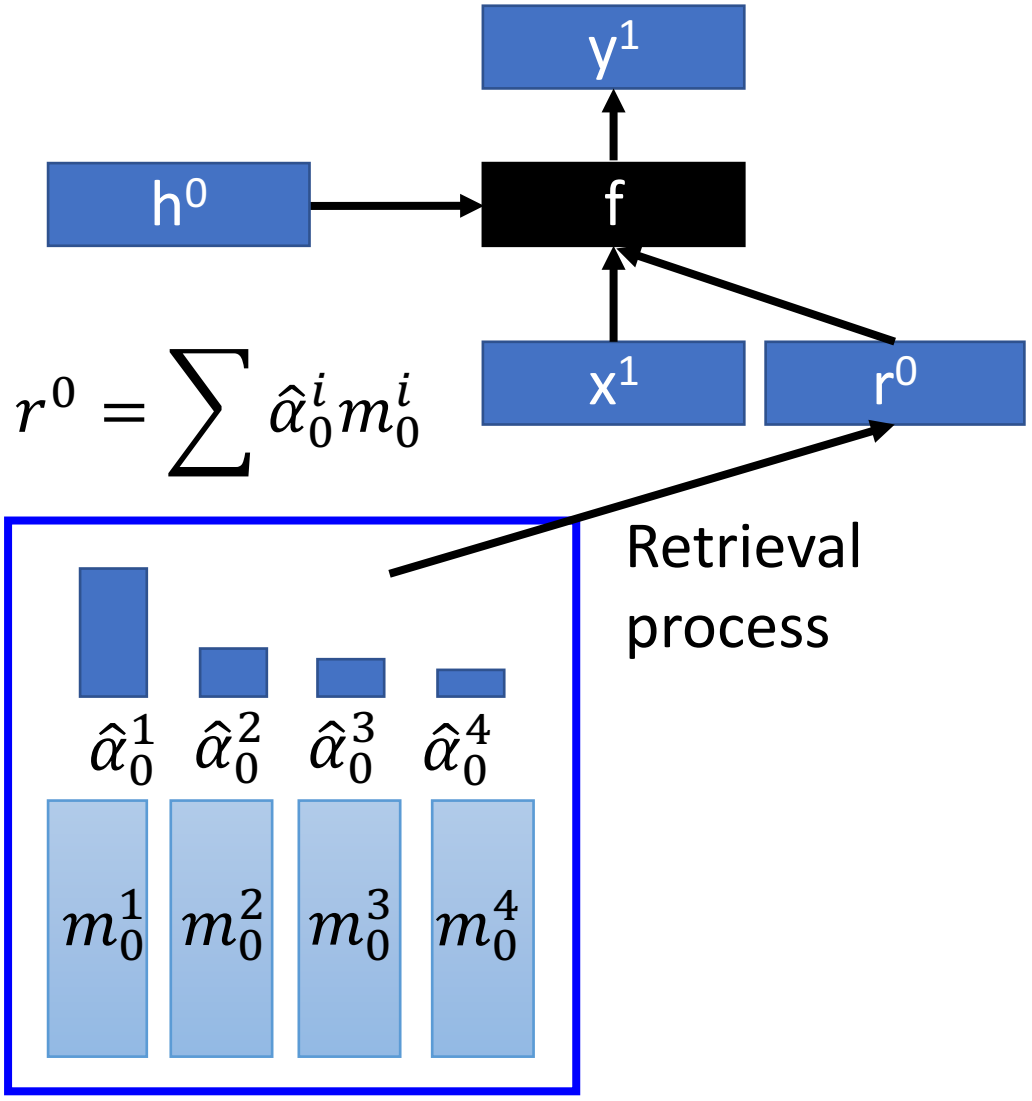


External Memory v2

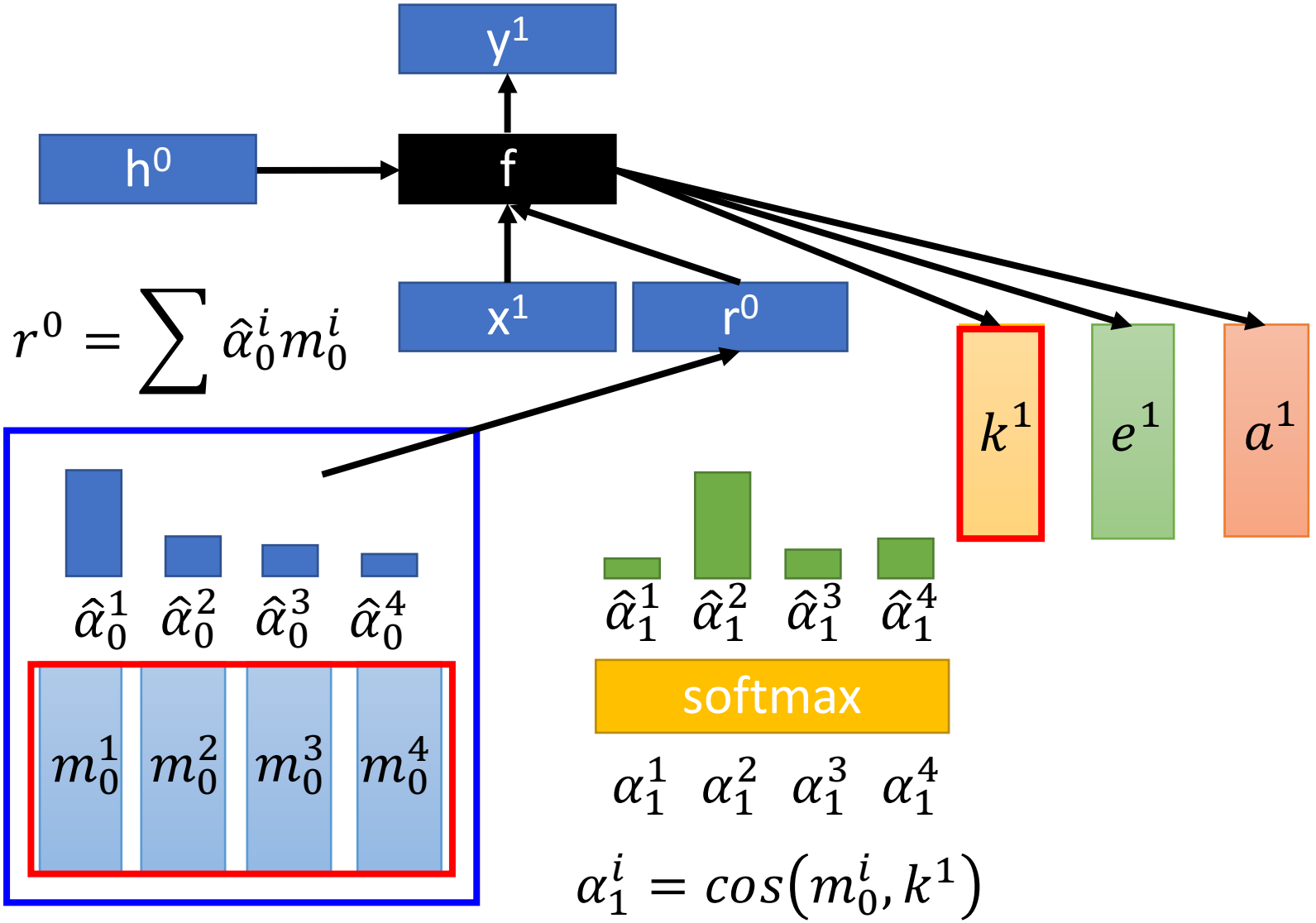


Neural Turing Machine

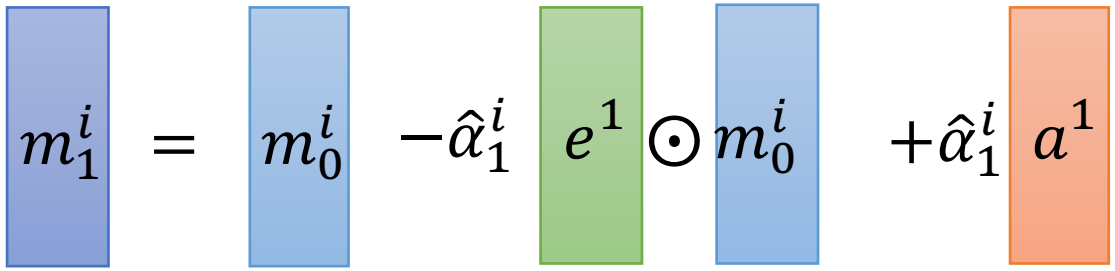
Neural Turing Machine



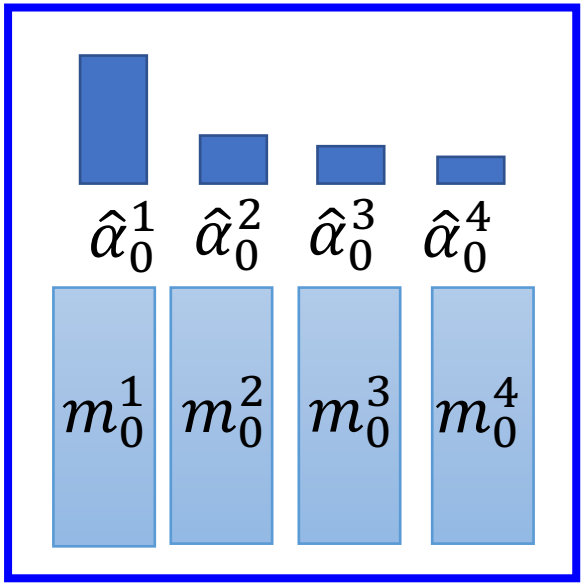
Neural Turing Machine

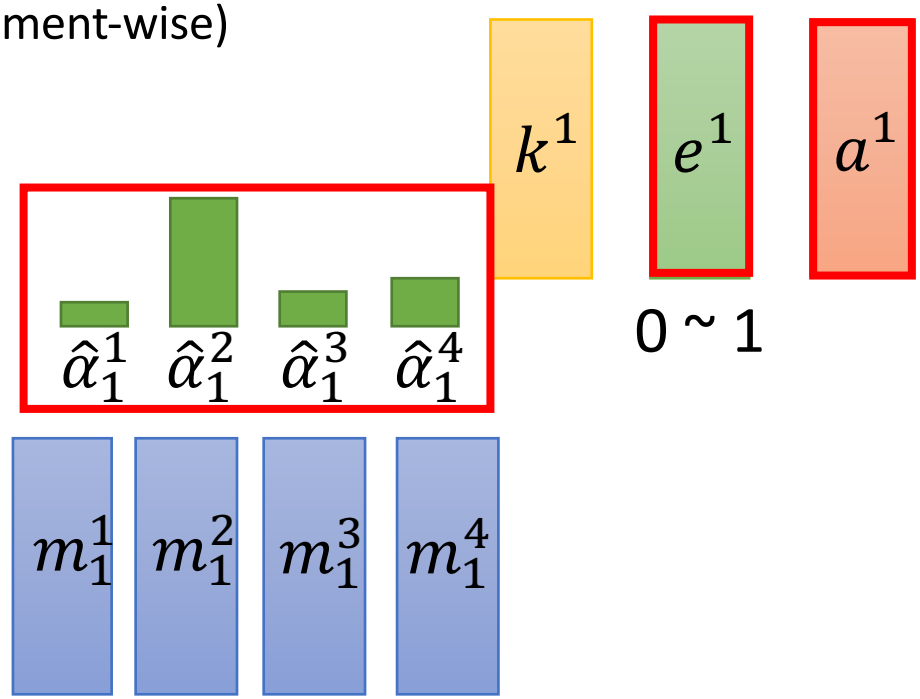


Neural Turing Machine

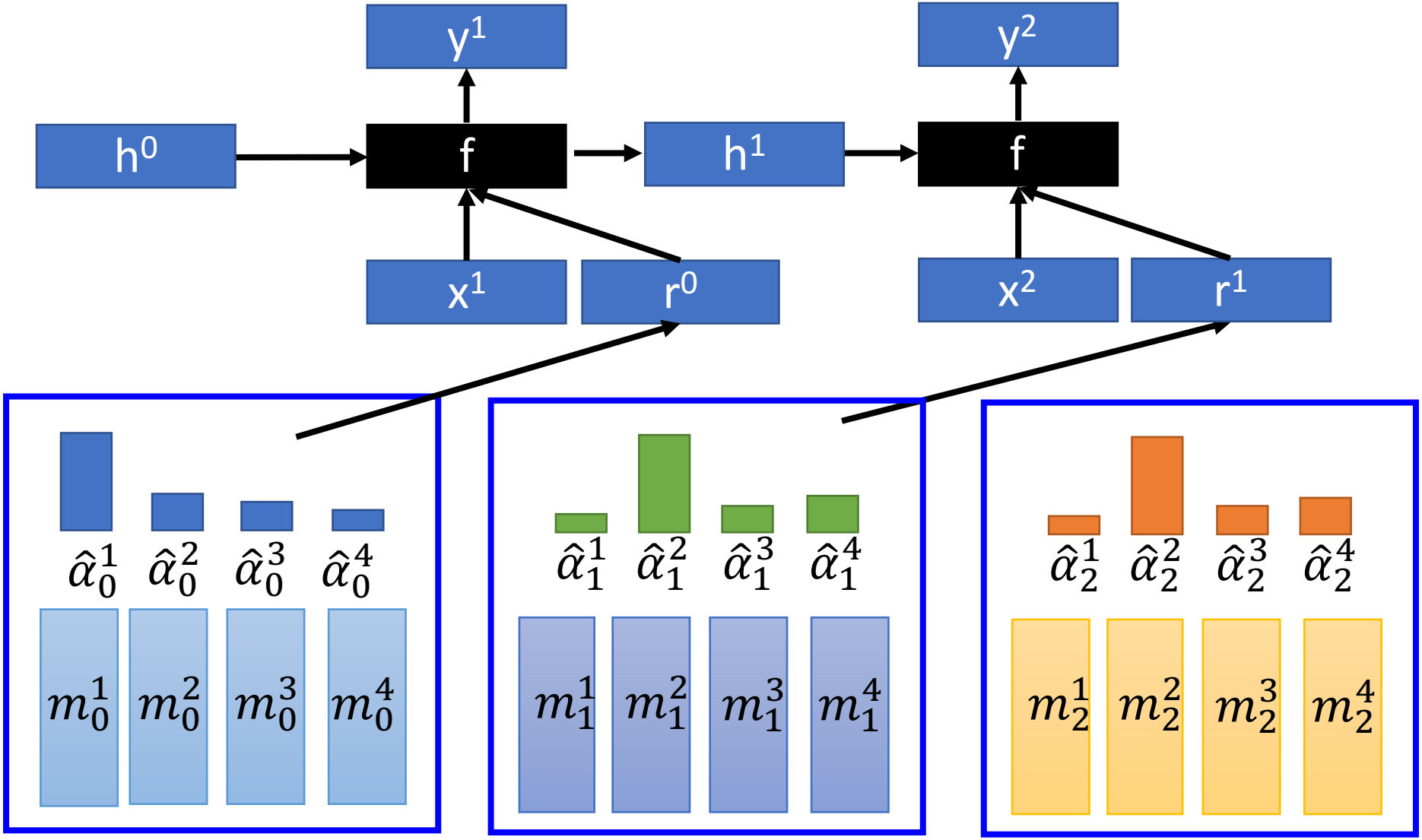


(element-wise)

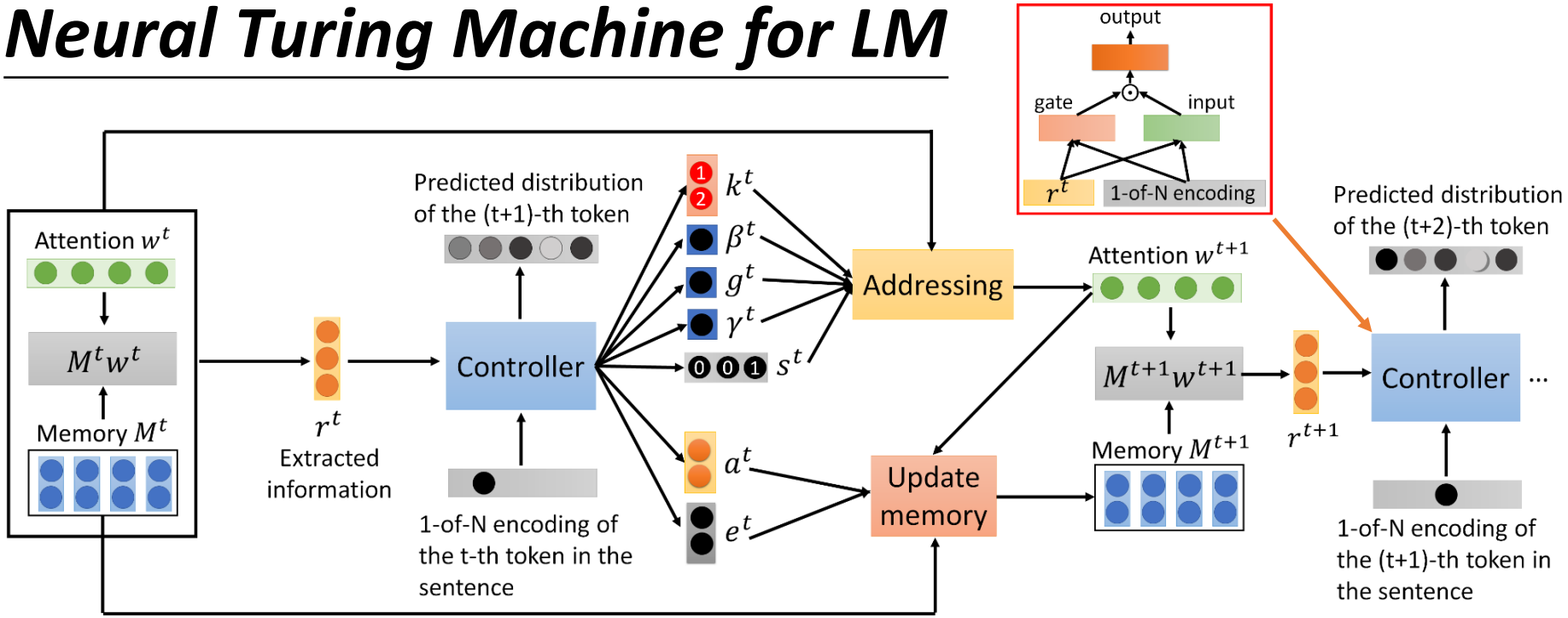


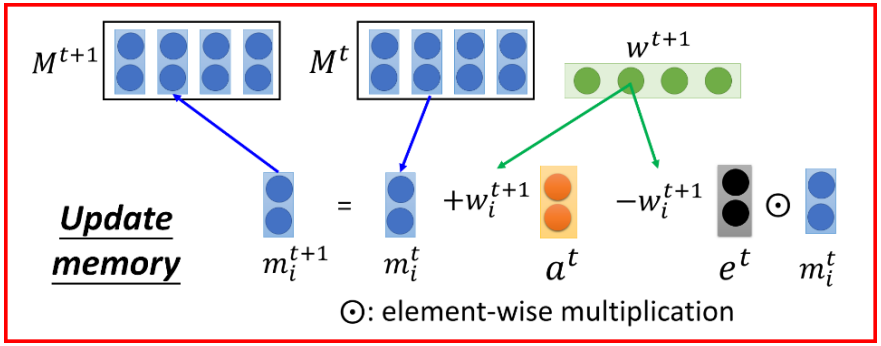


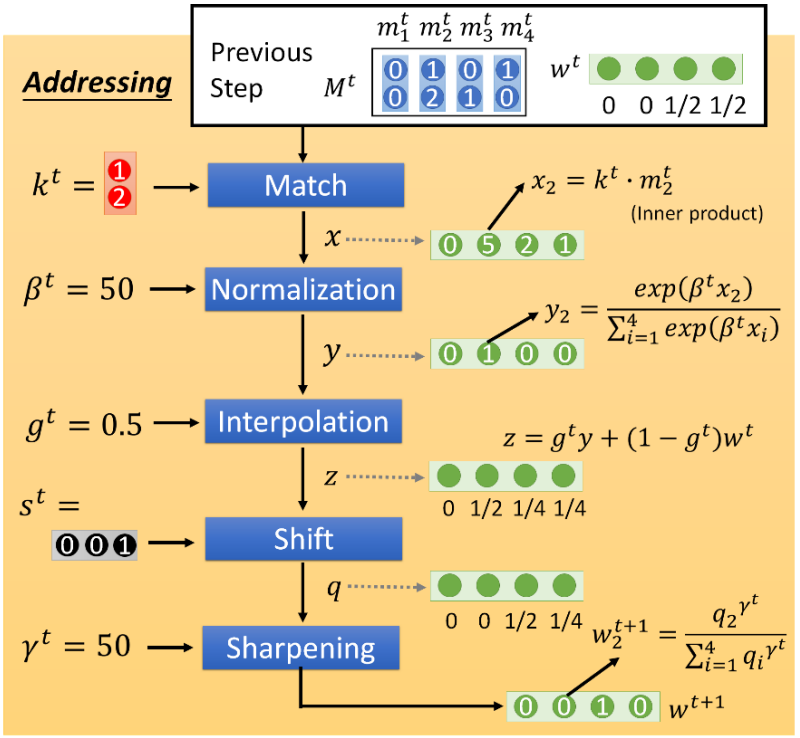
Neural Turing Machine



Neural Turing Machine for LM



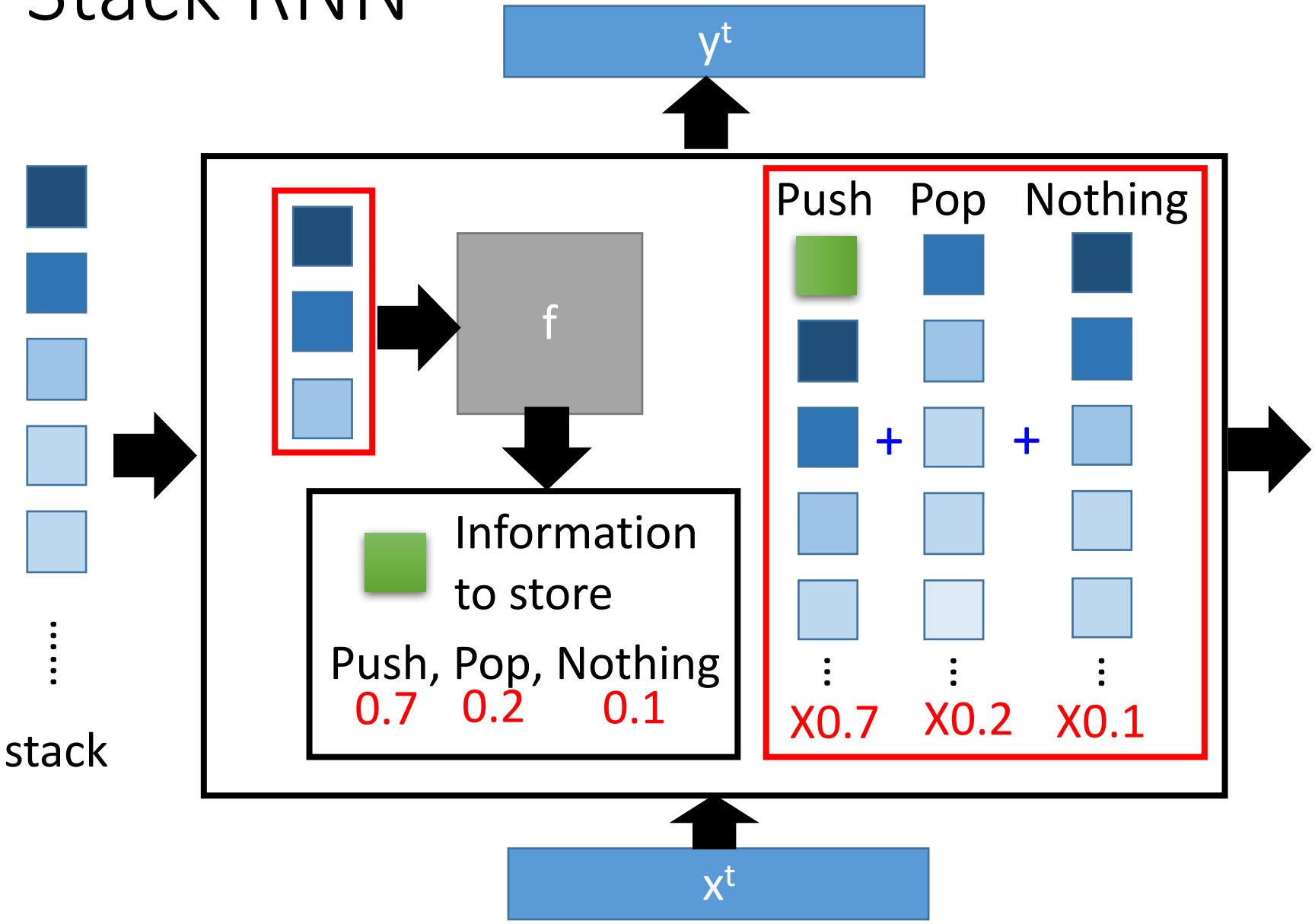




Wei-Jen Ko, Bo-Hsiang Tseng, Hung-yi Lee, "Recurrent Neural Network based Language Modeling with Controllable External Memory", ICASSP, 2017

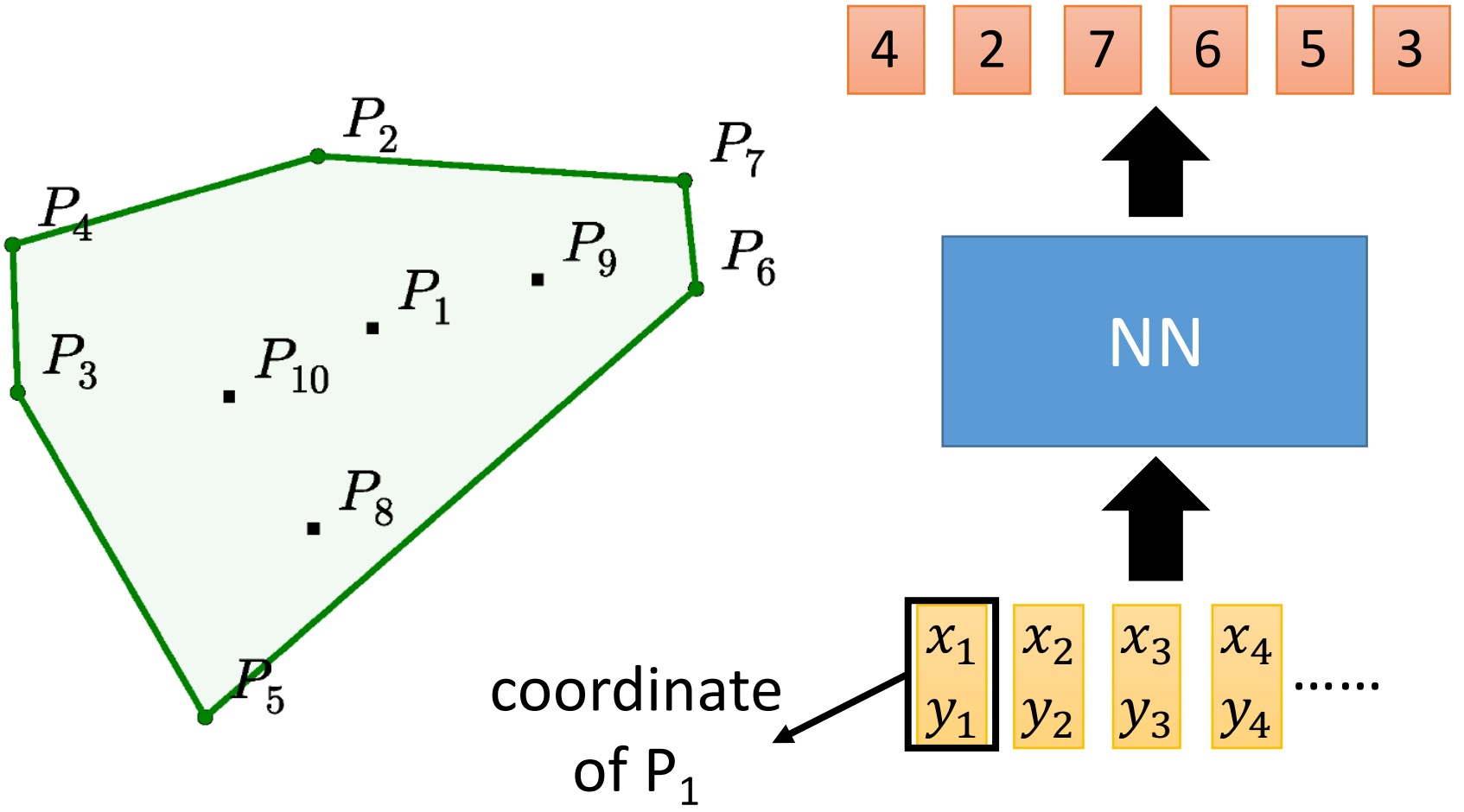
Armand Joulin, Tomas Mikolov, Inferring Algorithmic Patterns with Stack-Augmented Recurrent Nets, arXiv Pre-Print, 2015

Stack RNN



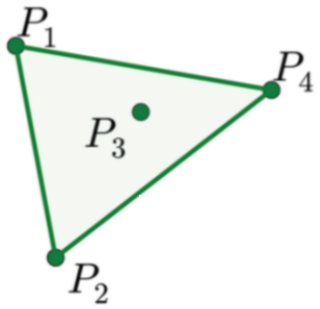
Pointer Network

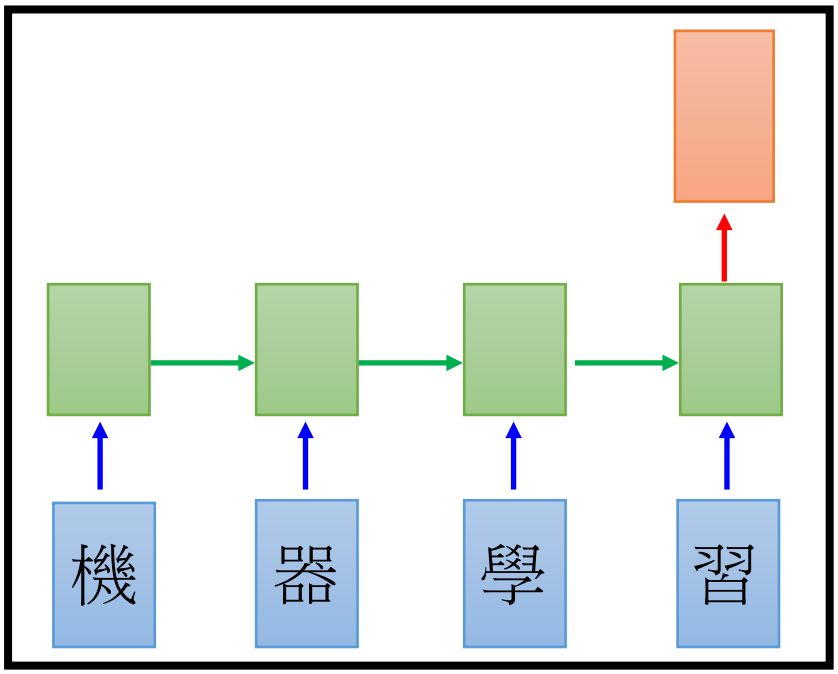
Pointer Network



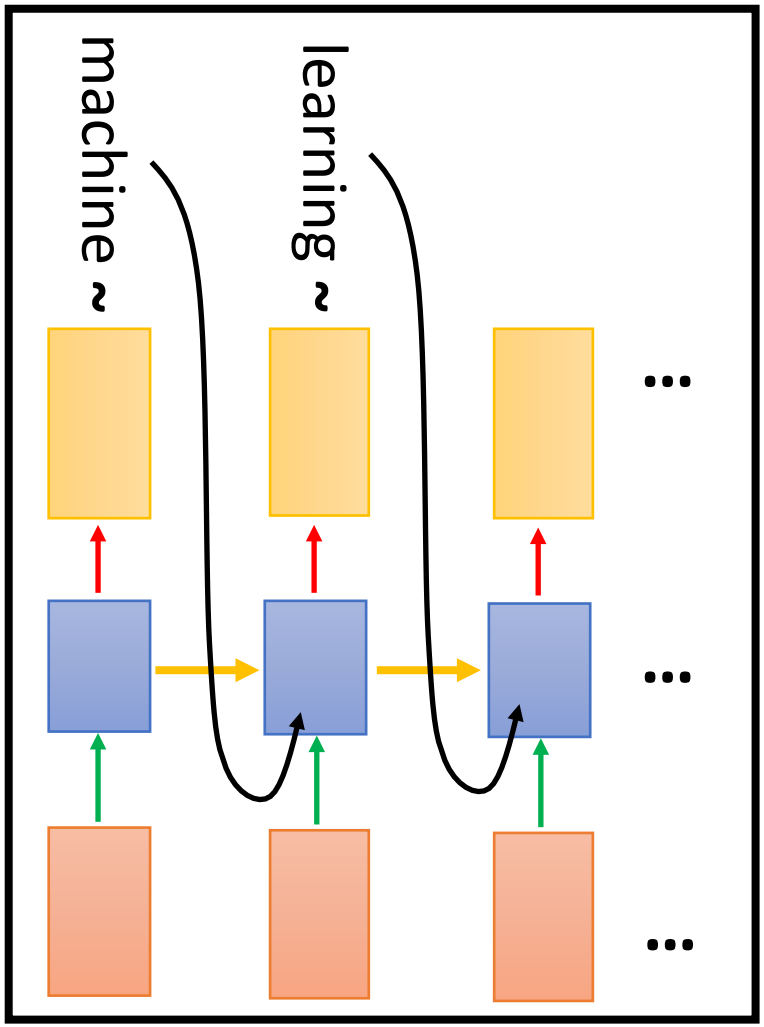
planar convex hulls, computing Delaunay triangulations, and the planar Travelling Salesman Problem

Sequence-to-sequence?





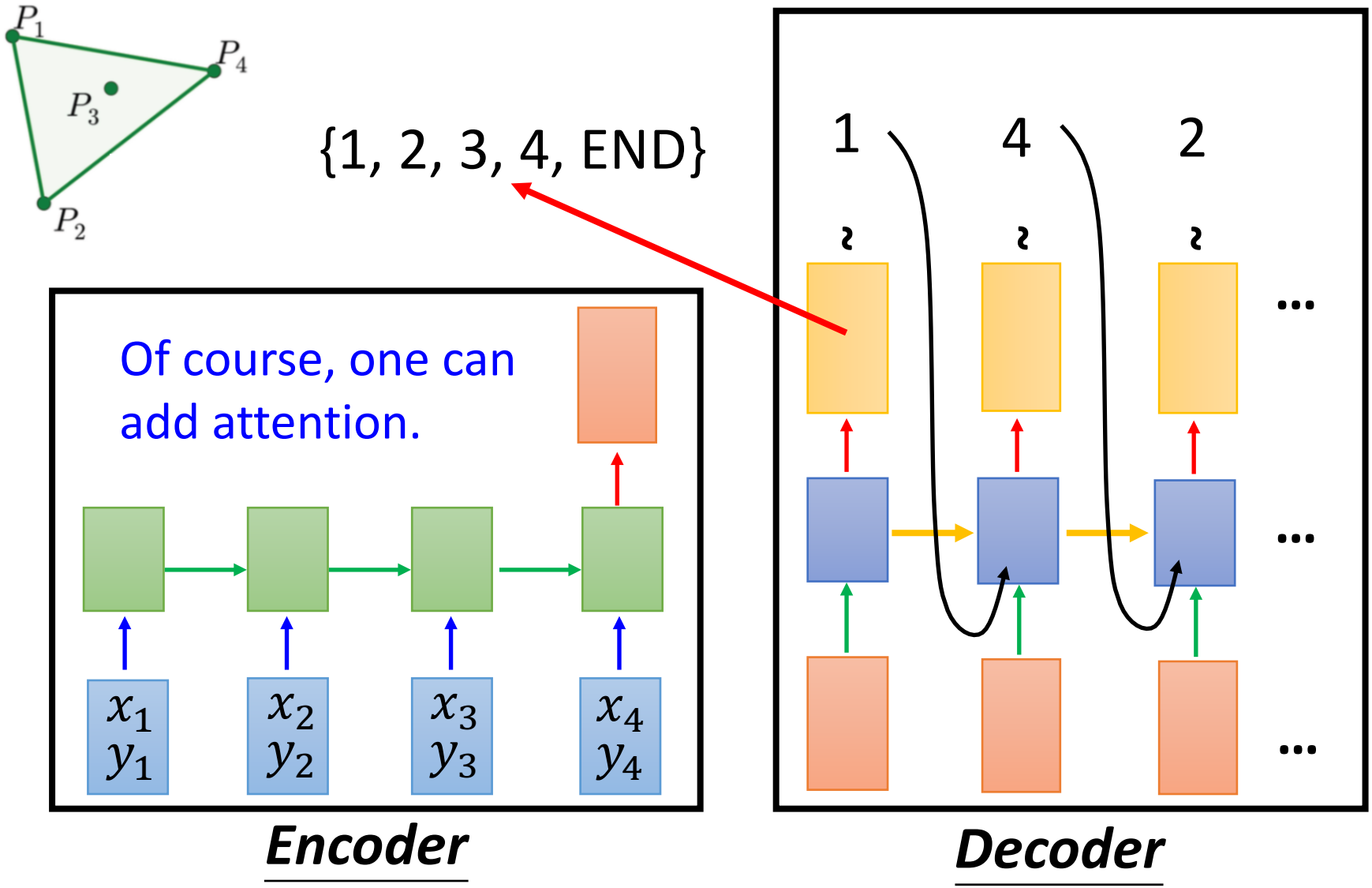
Encoder



Decoder

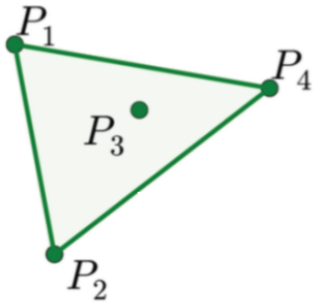
Problem?

Sequence-to-sequence?



Pointer Network

x_0 y_0 : END



Attention Weight

key

z^0

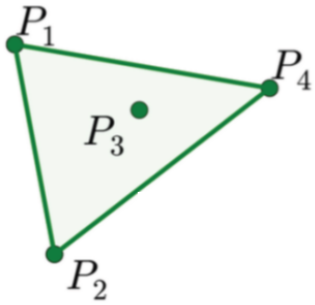
0.5

h^0 h^1 h^2 h^3 h^4

x_0 y_0 | x_1 y_1 | x_2 y_2 | x_3 y_3 | x_4 y_4

Pointer Network

x_0 y_0 : END

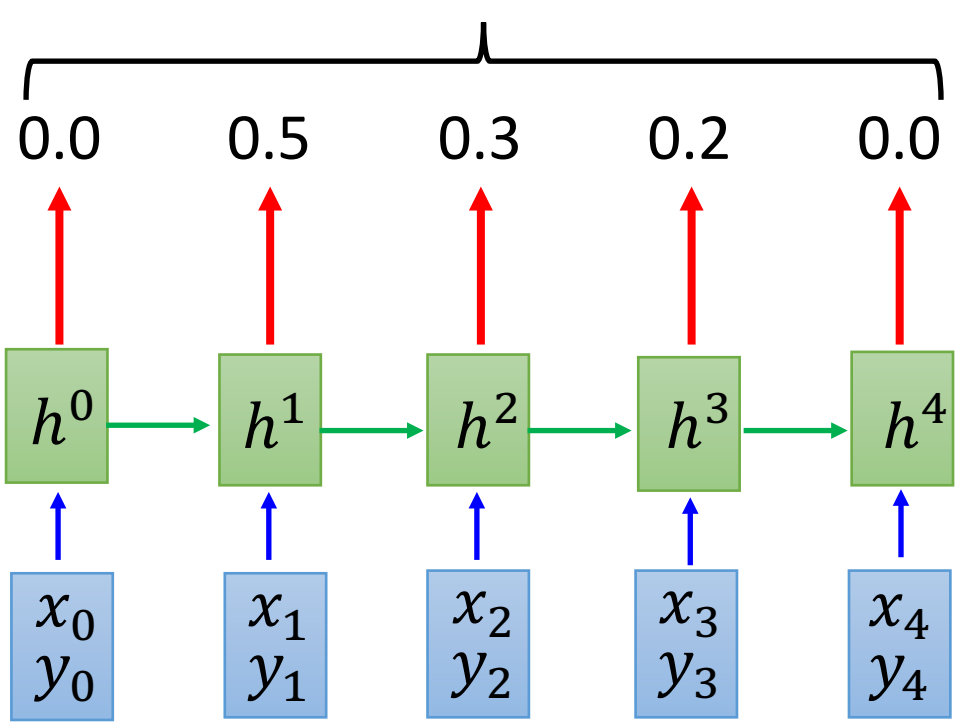


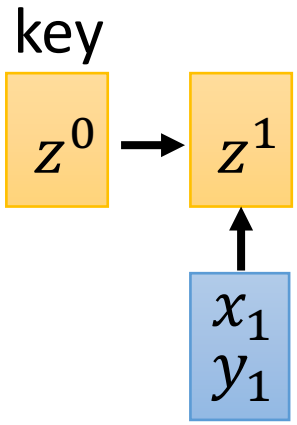
Output: 1

≈

argmax from this distribution

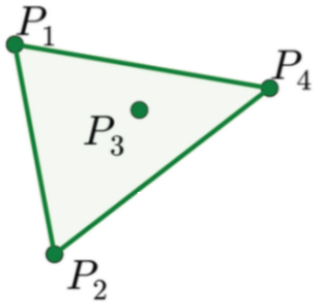
What decoder can output depends on the input.





Pointer Network

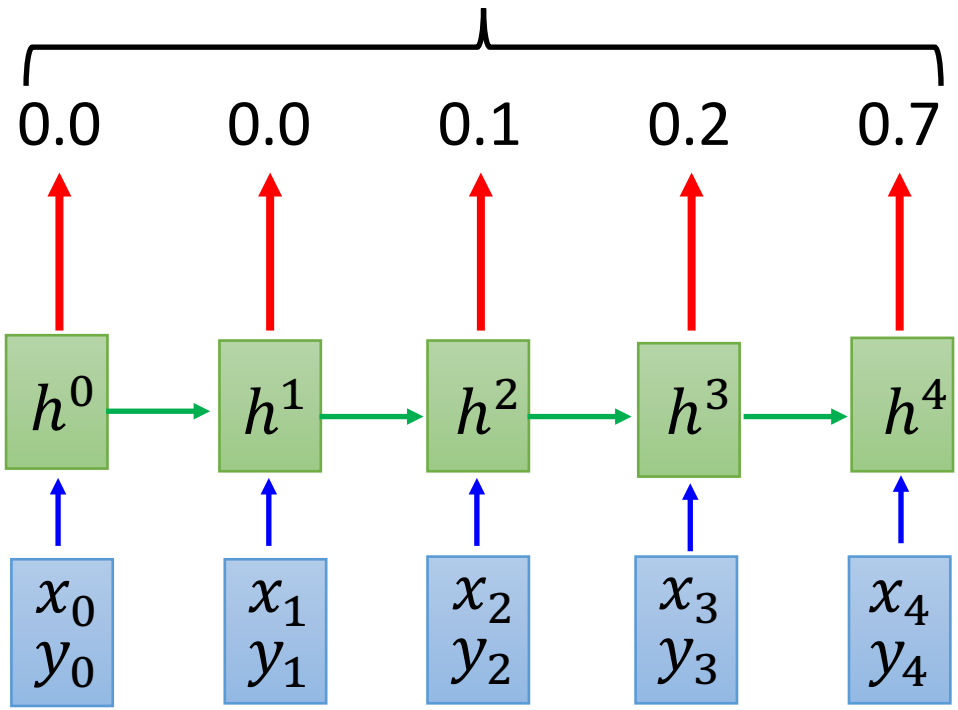
$\begin{matrix} x_0 \\ y_0 \end{matrix}$: END



Output: 4

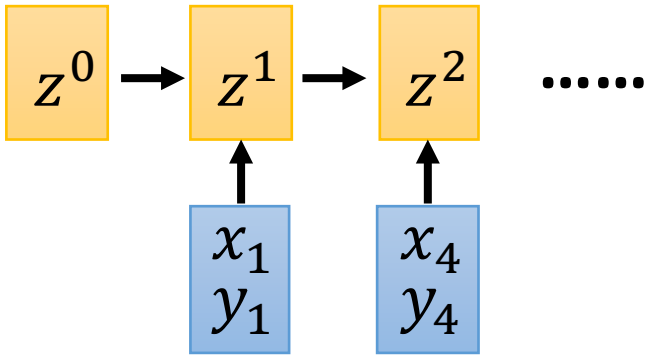
≈

argmax from this distribution



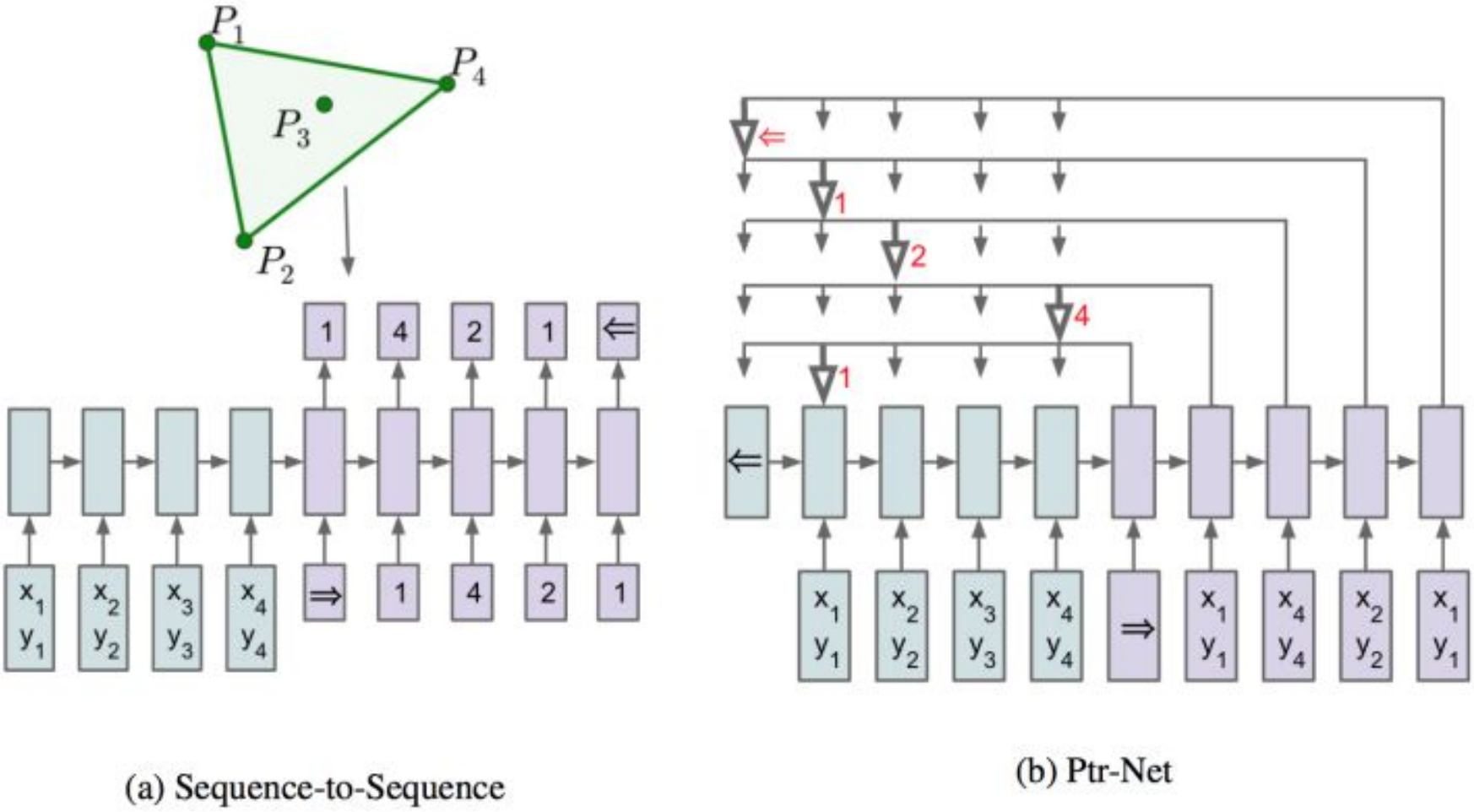
What decoder can output depends on the input.

key



The process stops when "END" has the largest attention weights.

Pointer Network

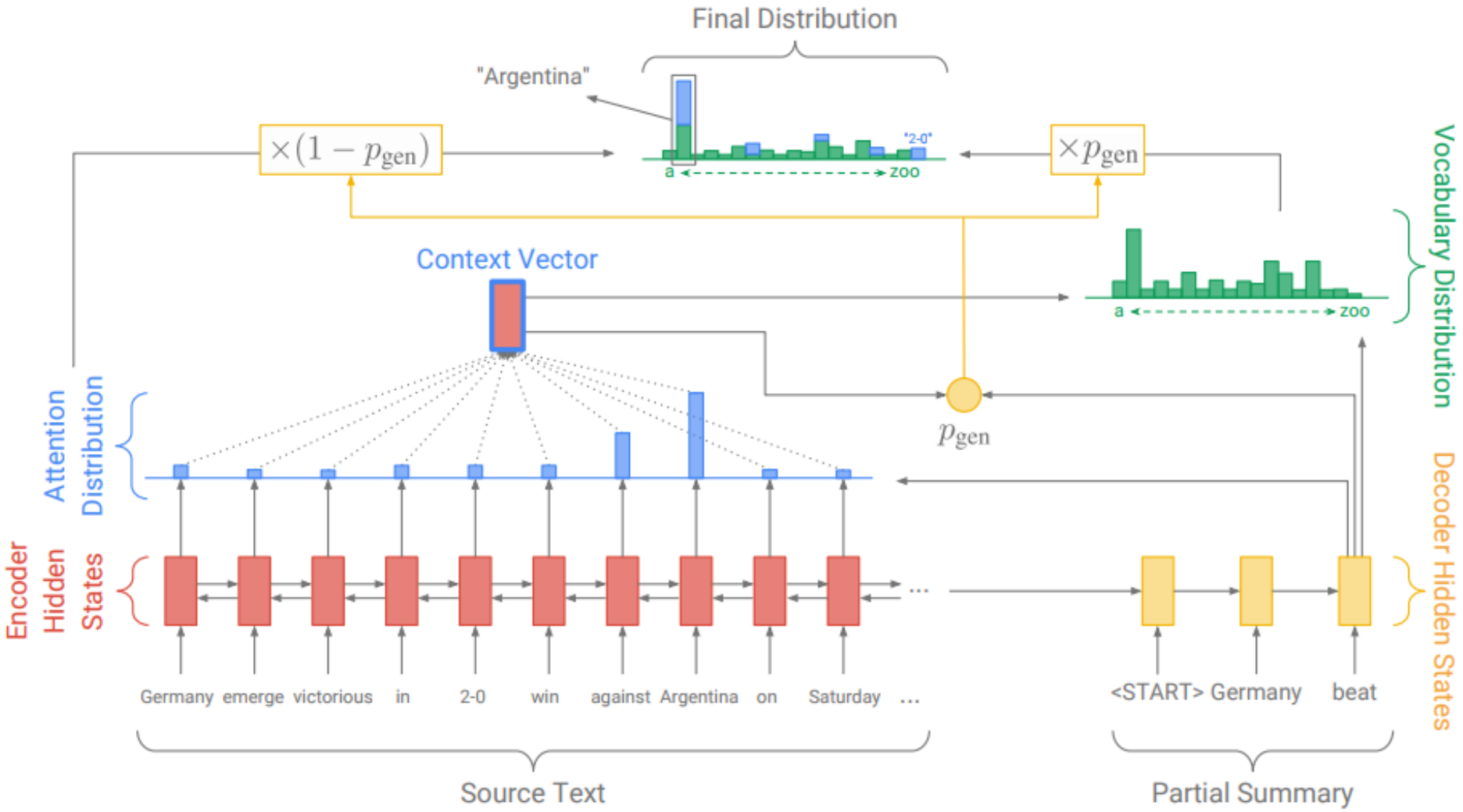


(a) Sequence-to-Sequence

(b) Ptr-Net

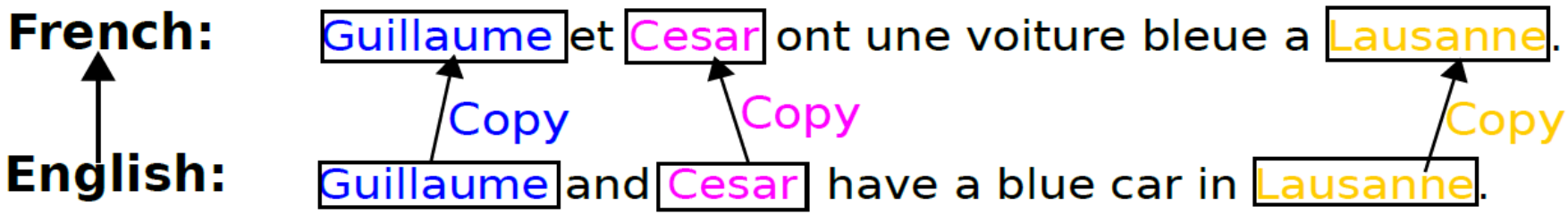
https://arxiv.org/abs/1704.04368

Applications - Summarization



More Applications

Machine Translation



Chat-bot

User: X寶你好，我是庫洛洛

Machine:庫洛洛你好，很高興認識你

More Applications

- Article summarization -《Neural Summarization by Extracting Sentences and Words》
- Information retrieval -《End-to-End Information Extraction without Token-Level Supervision》
- Sentence ordering =《End-to-End Neural Sentence Ordering Using Pointer Network》
- Auto programming for cards of board games
  《Latent predictor networks for code generation》