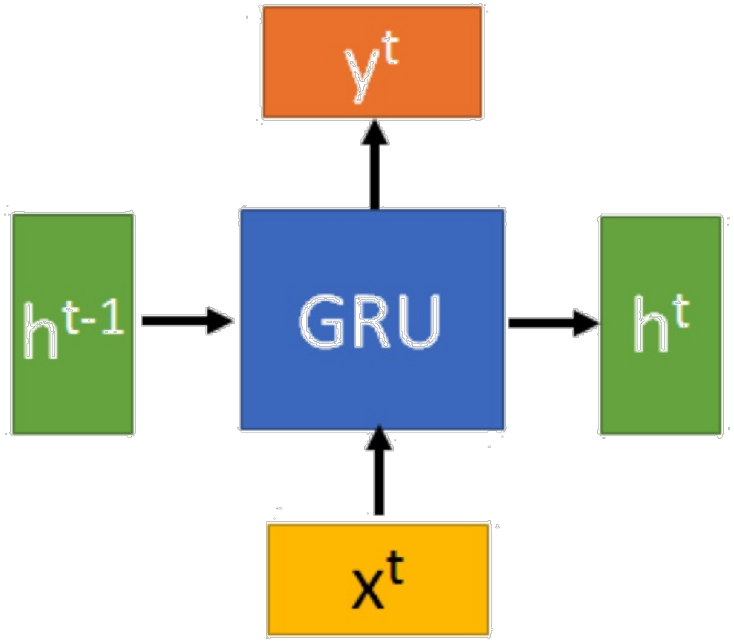
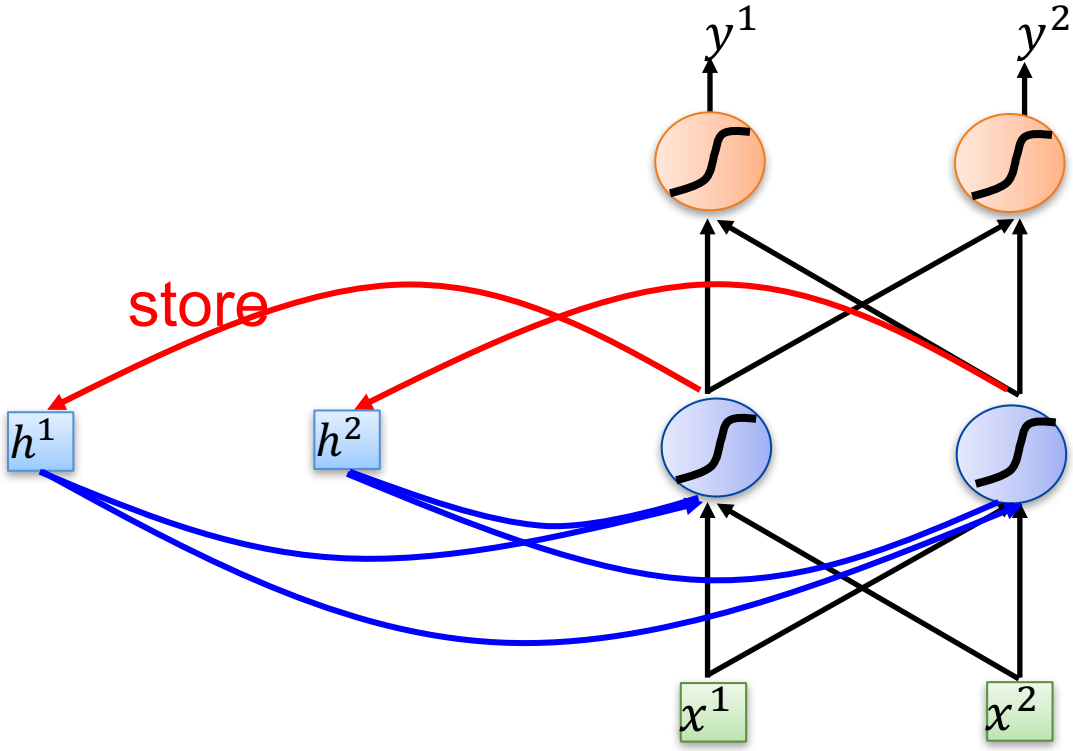
Gate Recurrent Unit

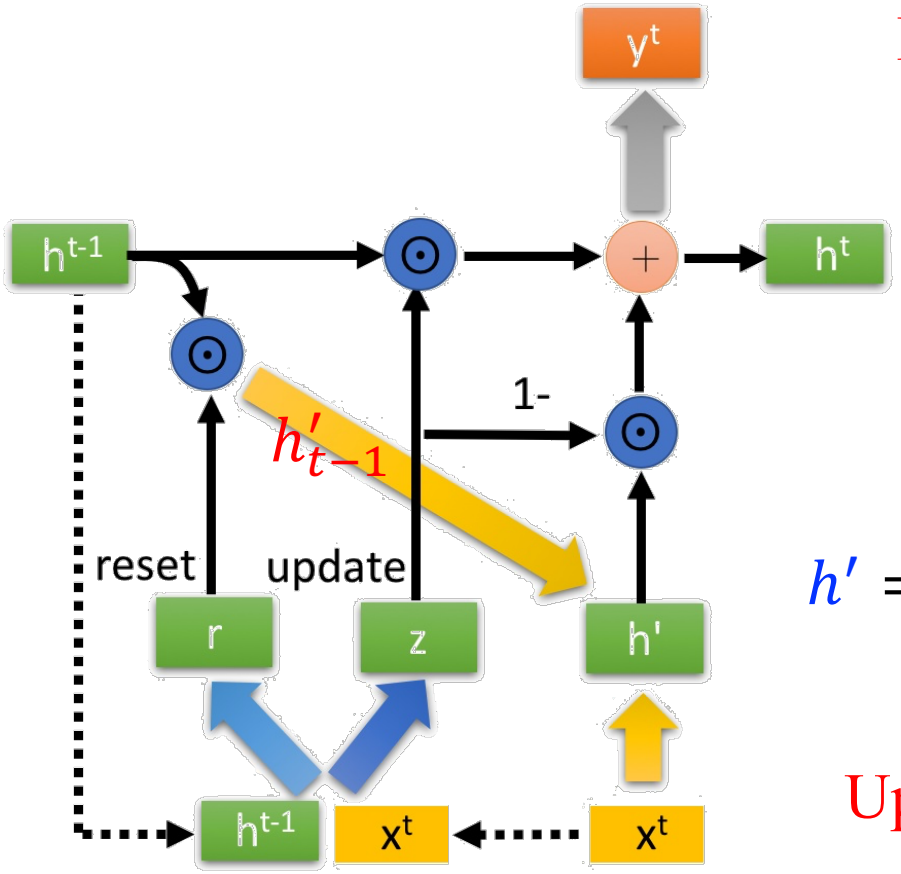
Gate Recurrent Unit





Vanilla RNN

Gate Recurrent Unit



Reset Gate

$$r_t = \sigma(w_{rh}h_{t-1} + w_{rx}x_t + b_r)$$

$$h'_{t-1} = r_t * h_{t-1}$$

$$h' = \tanh(w_{hh}h'_{t-1} + w_{hx}x_t + b_h)$$

Update Gate (forget gate + input gate)

$$z_t = \sigma(w_{zh}h_{t-1} + w_{zx}x_t + b_z)$$

$$h_t = z_t * h_{t-1} + (1 - z_t) * h'$$

Application

- Visual Question Answer (VQA)
- Reading Comprehension
- Sequence To Sequence (Seq2seq)
  - Language Translation
  - Chat Robot

VQA

What's the mustache made of ?

AI System

Banana

Image credit to http://www.visualqa.org/challenge.html

VQA

Who is wearing glasses?

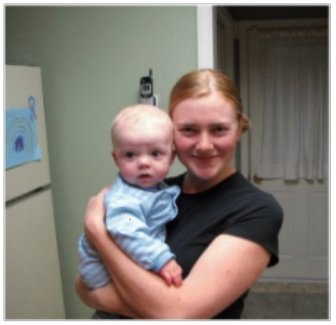
man

woman

Where is the child sitting?

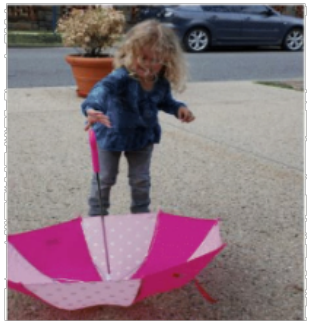
fridge

arms



Is the umbrella upside down?

yes



no

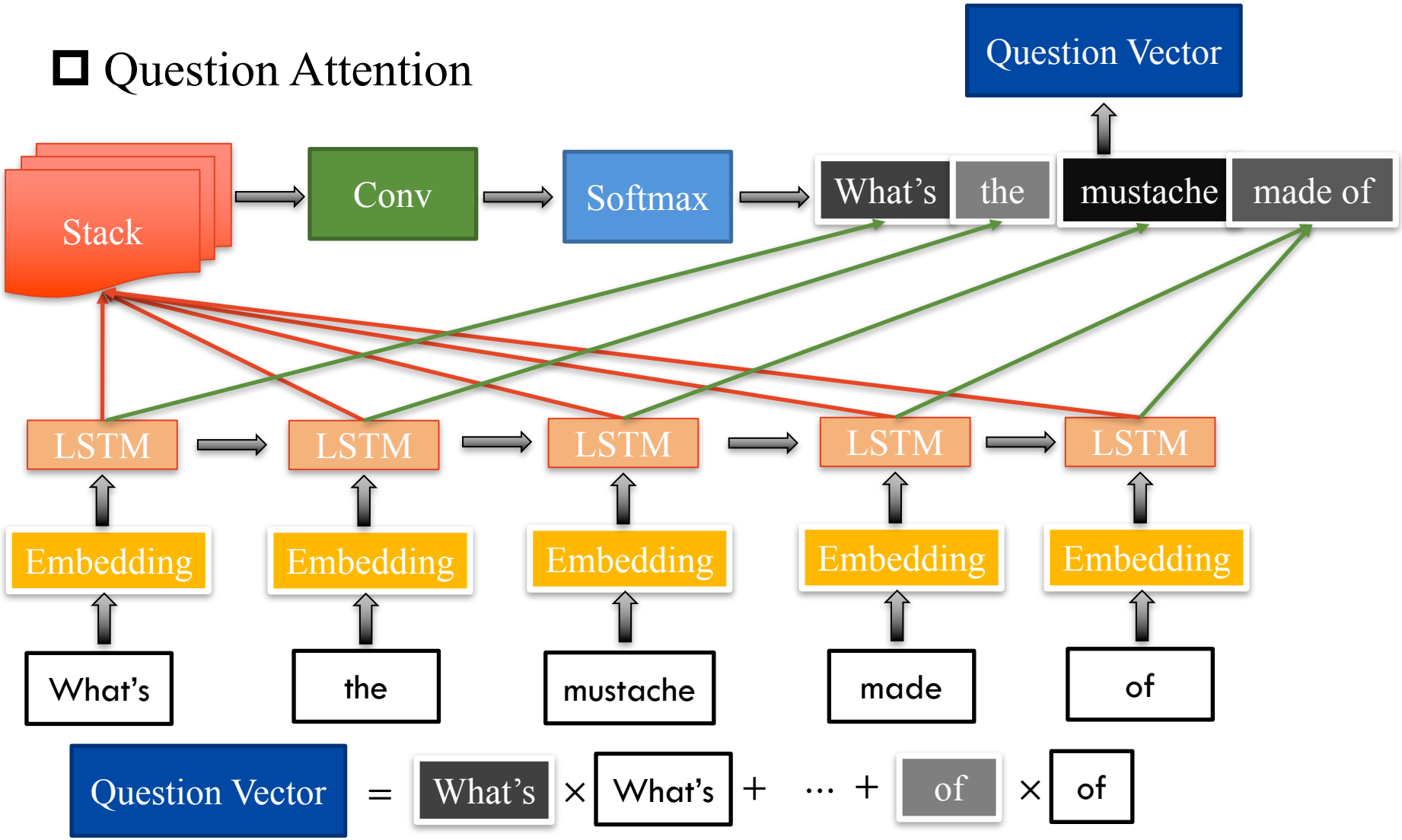
How many children are in the bed?

2

1

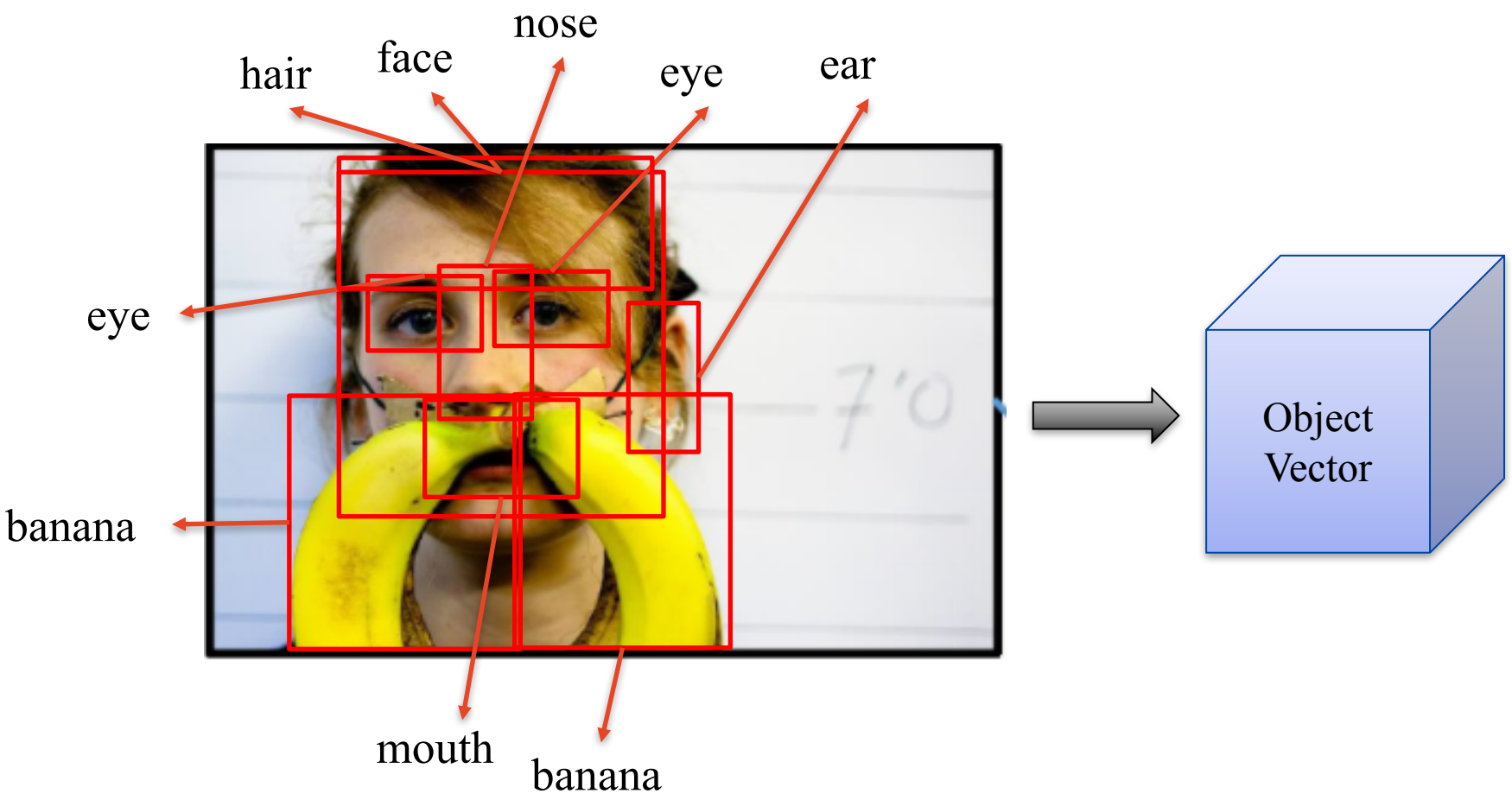
VQA

Question Attention



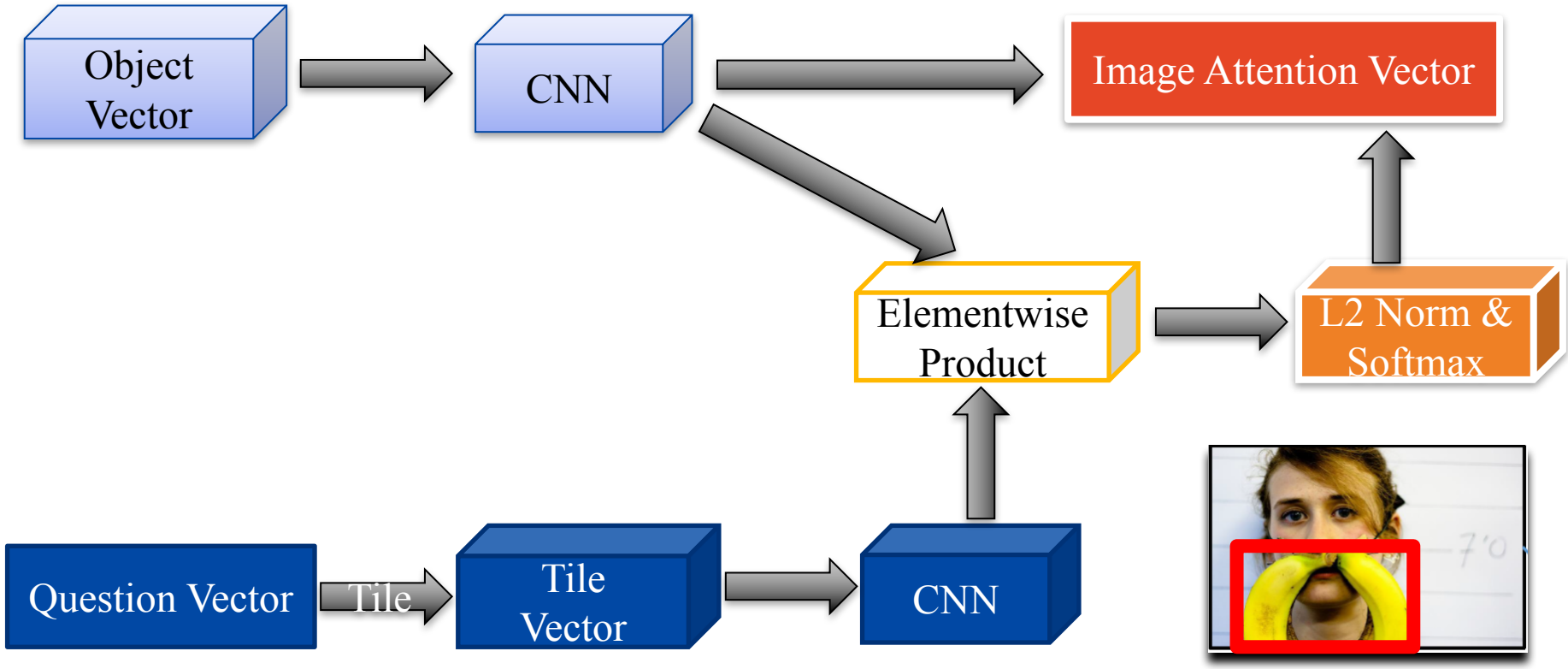
VQA

- Object Detection

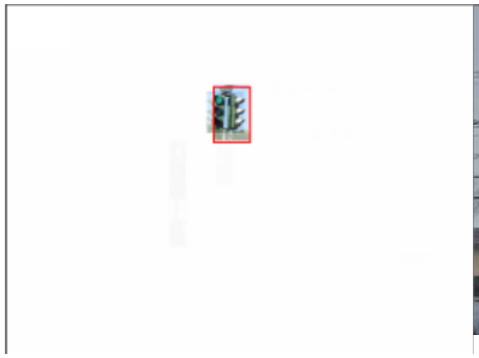


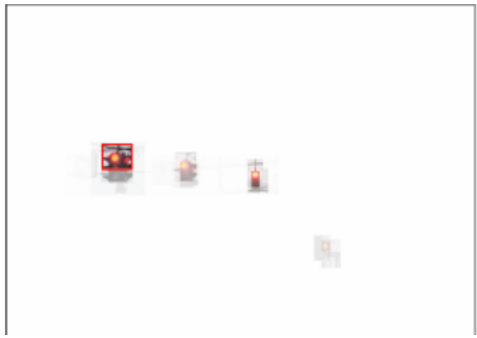
VQA

- Image Attention

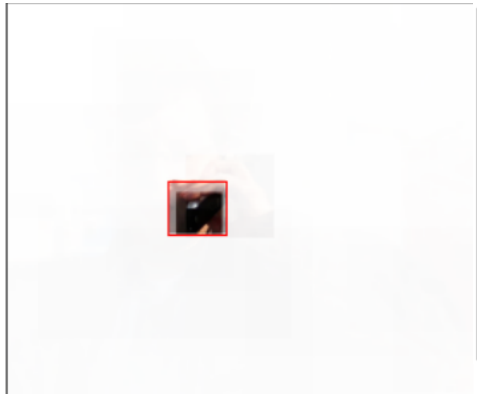


VQA



What color is illuminated on the traffic light?

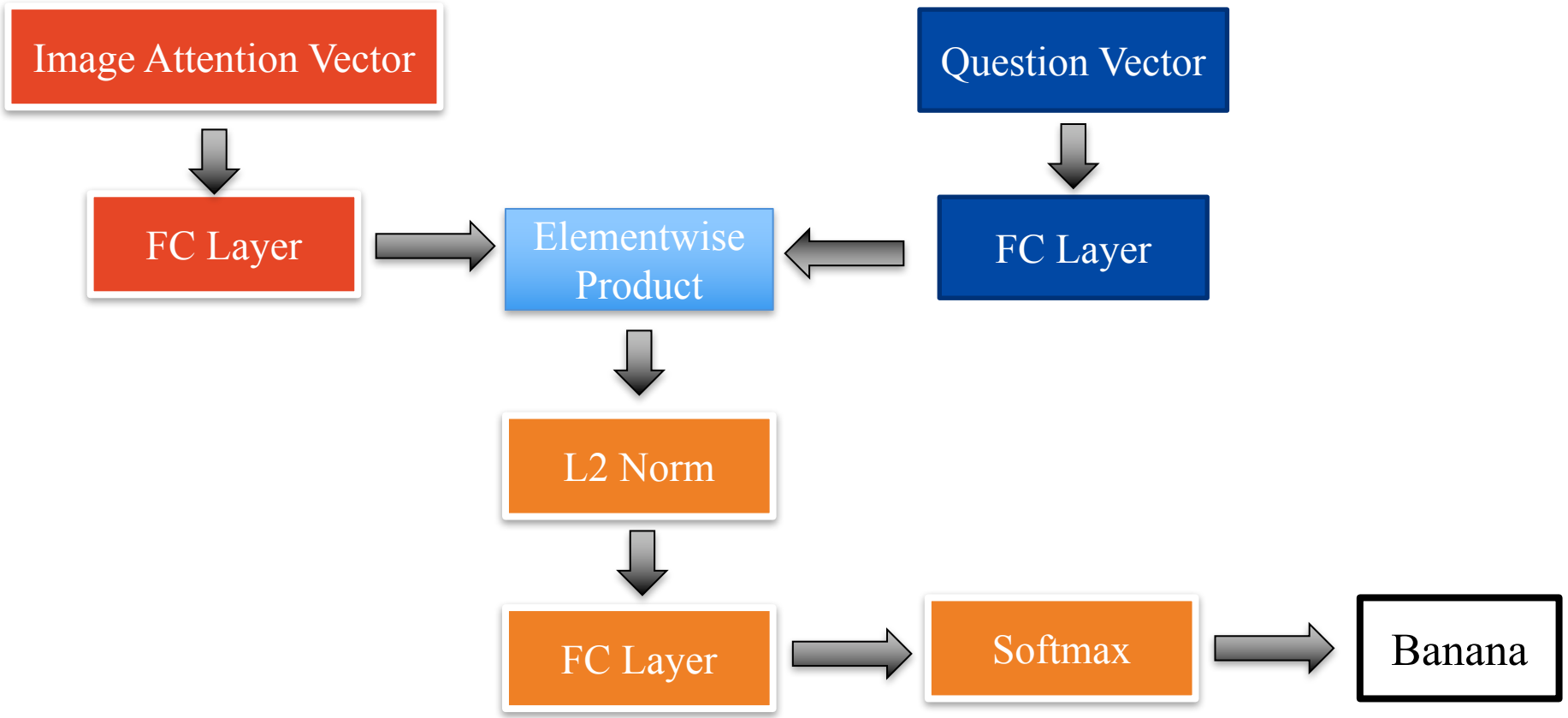



What is the man holding?

Image credit to Bottom-Up and Top-Down Attention for Image Captioning and Visual Question Answering

VQA

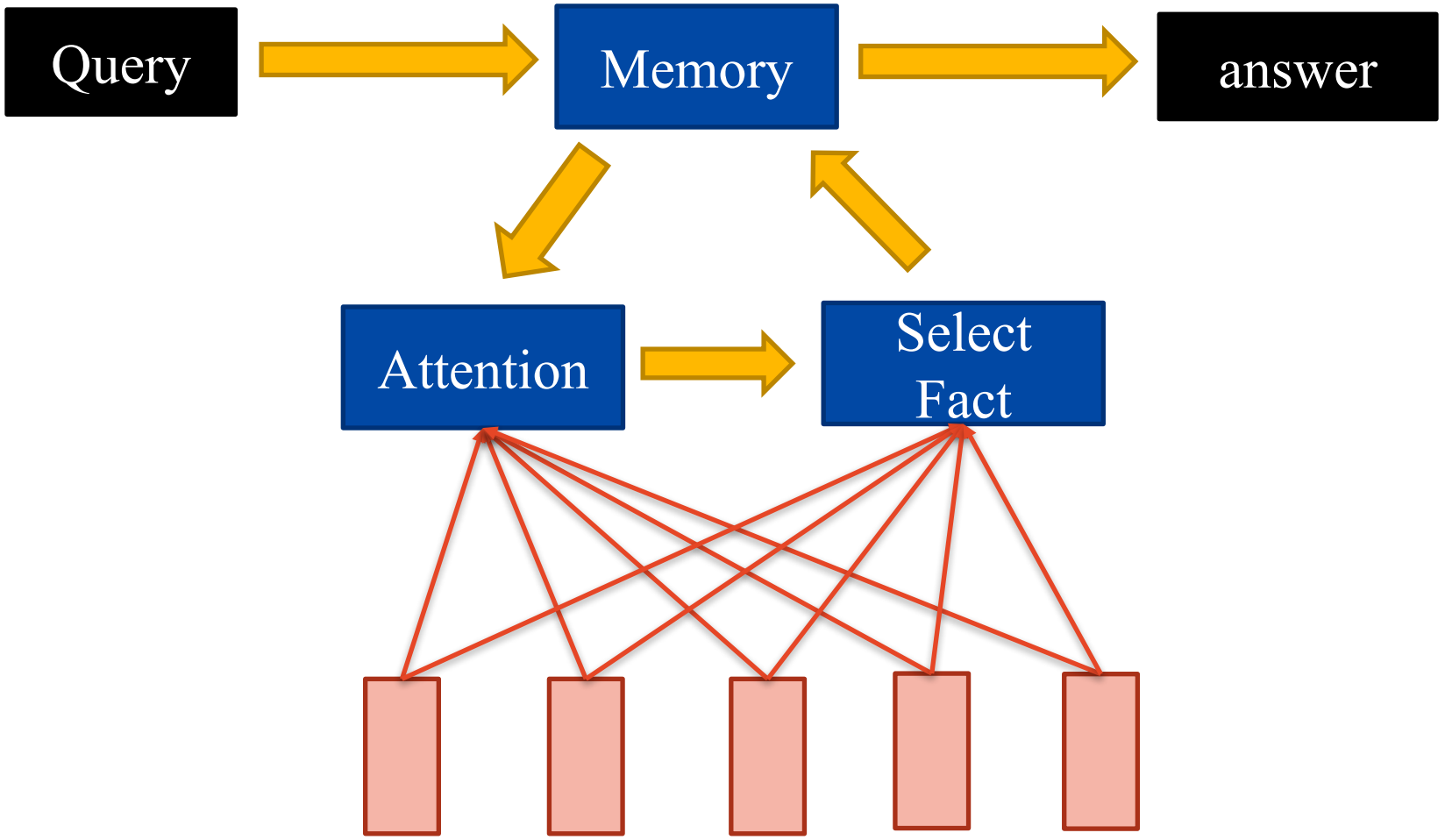
- Question & Image Fusion



Reading Comprehension

- Answering question based on given facts
- Example
  - A. Brian is a frog.
  - B. Lily is gray.
  - C. Brian is yellow
  - D. Julius is green.
  - E. Greg is a frog
  - ✓ What color is Greg?

Reading Comprehension



Each Fact becomes a vector by LSTM

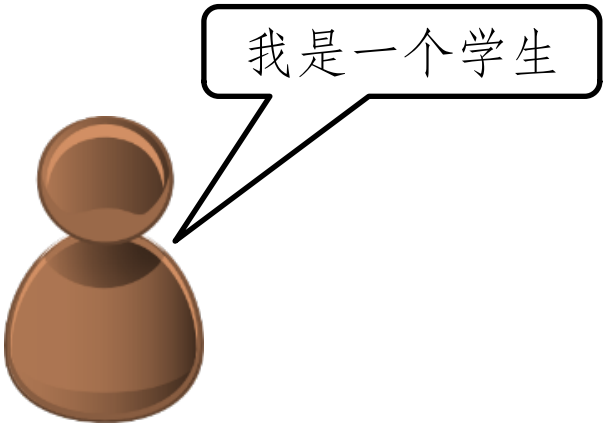
Reading Comprehension

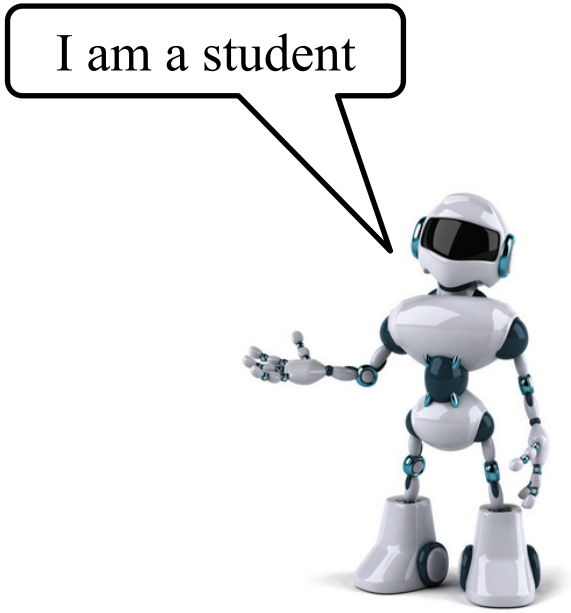
Story (16: basic induction)	Support	Hop 1	Hop 2	Hop 3
Brian is a frog.	yes	0.00	0.98	0.00
Lily is gray.		0.07	0.00	0.00
Brian is yellow.	yes	0.07	0.00	1.00
Julius is green.		0.06	0.00	0.00
Greg is a frog.	yes	0.76	0.02	0.00
What color is Greg? Answer: yellow Prediction: yellow				

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

Seq2seq

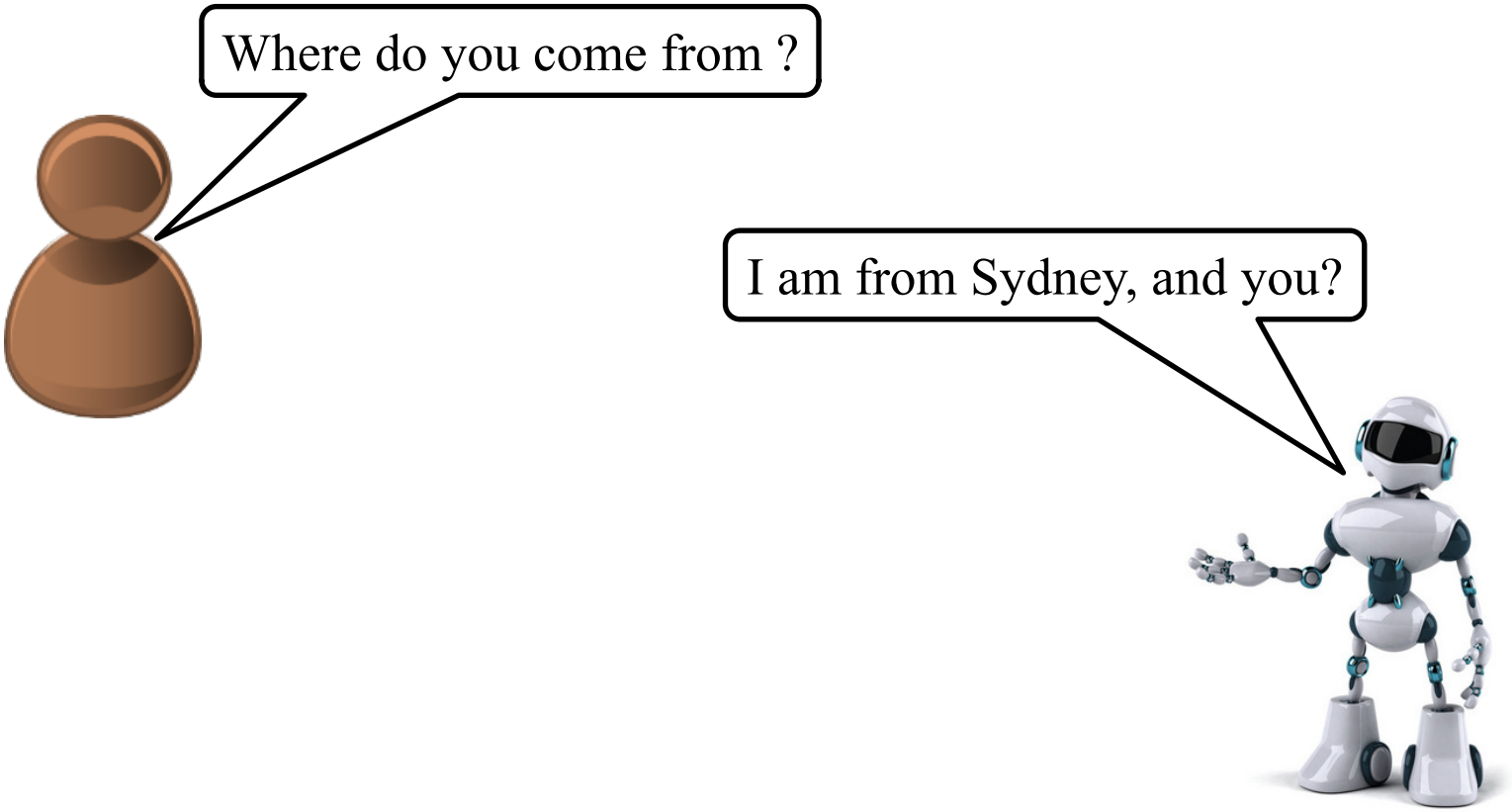
- Language Translation



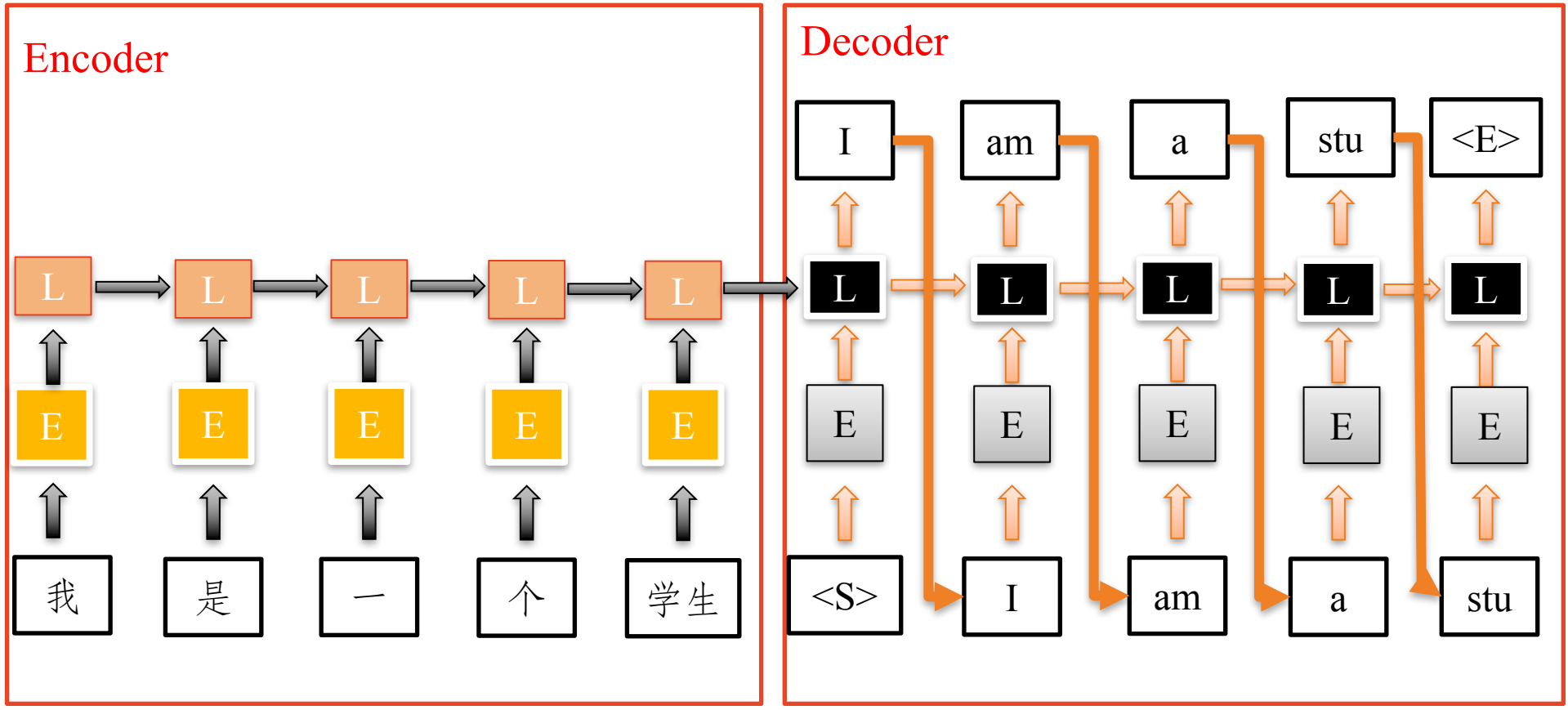


Seq2seq

- Chat Robot

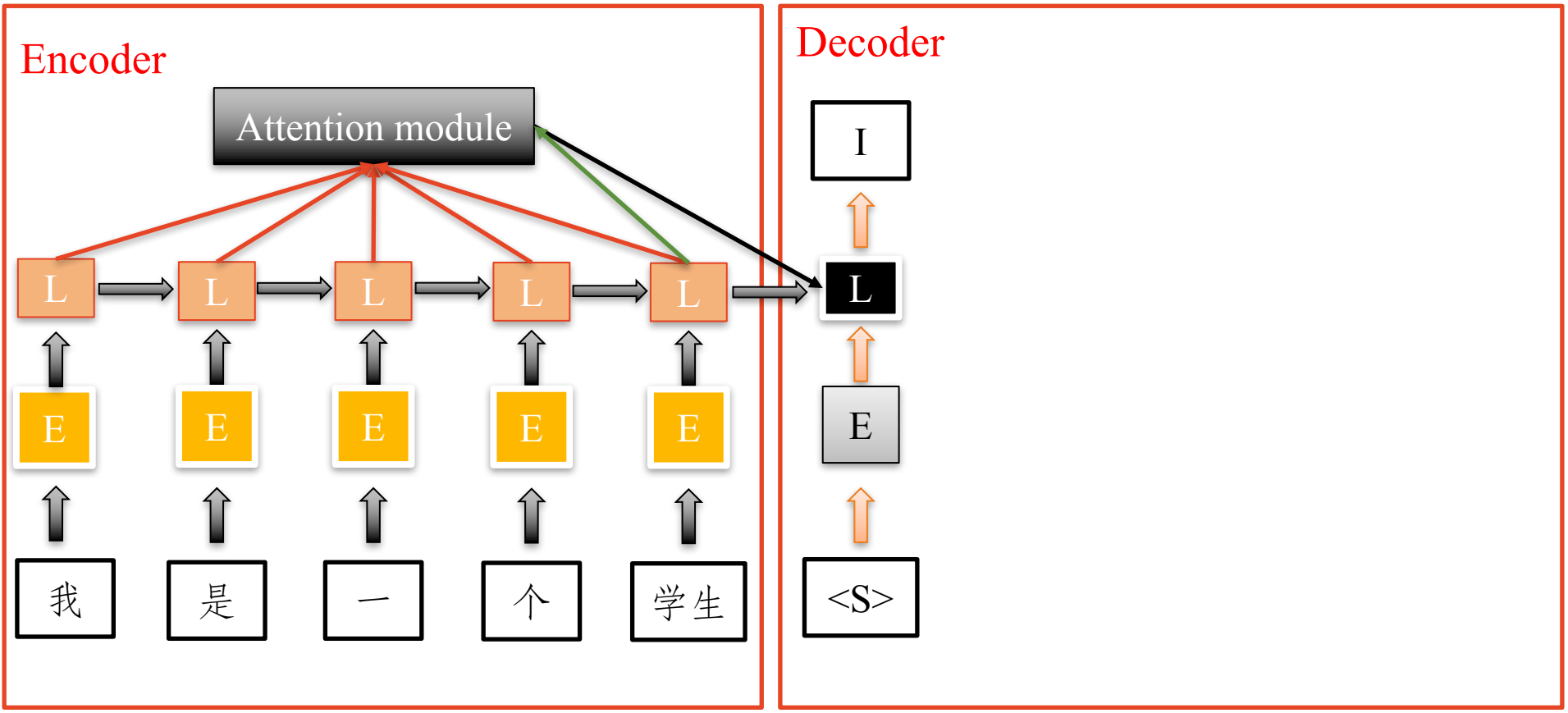


Seq2seq



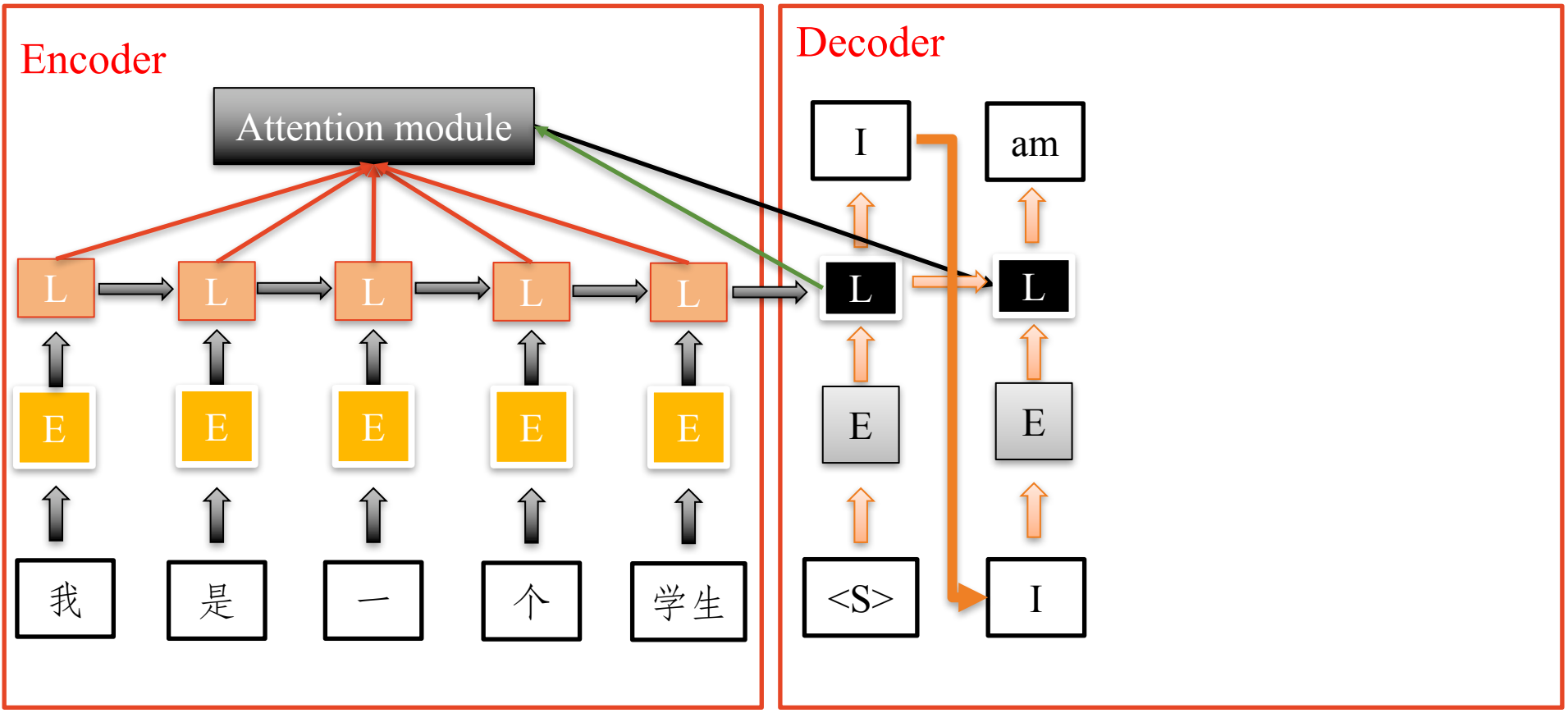
Seq2seq

- Attention



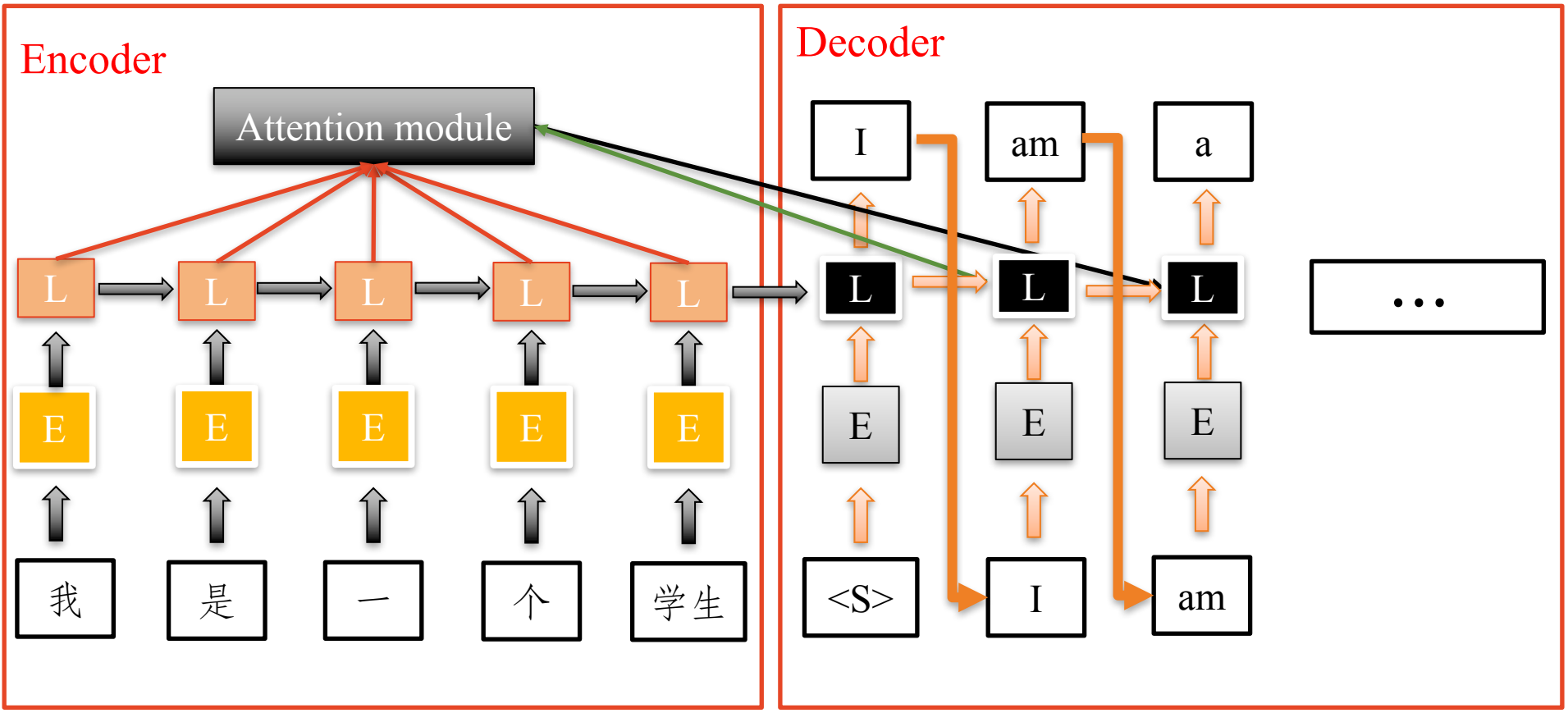
Seq2seq

- Attention



Seq2seq

- Attention



Seq2seq

- Two stage
  - Encode stage
  - Decode stage
- Attention
  - Weight for each encode symbol in decode step
  - Another input of LSTM represented as a_t
  - $f_t = \sigma(w_{fh}h_{t-1} + w_{fx}x_t + w_{fa}a_t + b_f)$
  - $i_t = \sigma(w_{ih}h_{t-1} + w_{ix}x_t + w_{ia}a_t + b_i)$
  - $o_t = \sigma(w_{oh}h_{t-1} + w_{ox}x_t + w_{oa}a_t + b_o)$
  - $\widetilde{c_t} = \tanh(w_{hh}h_{t-1} + w_{hx}x_t + w_{ha}a_t + b_h)$

Seq2seq

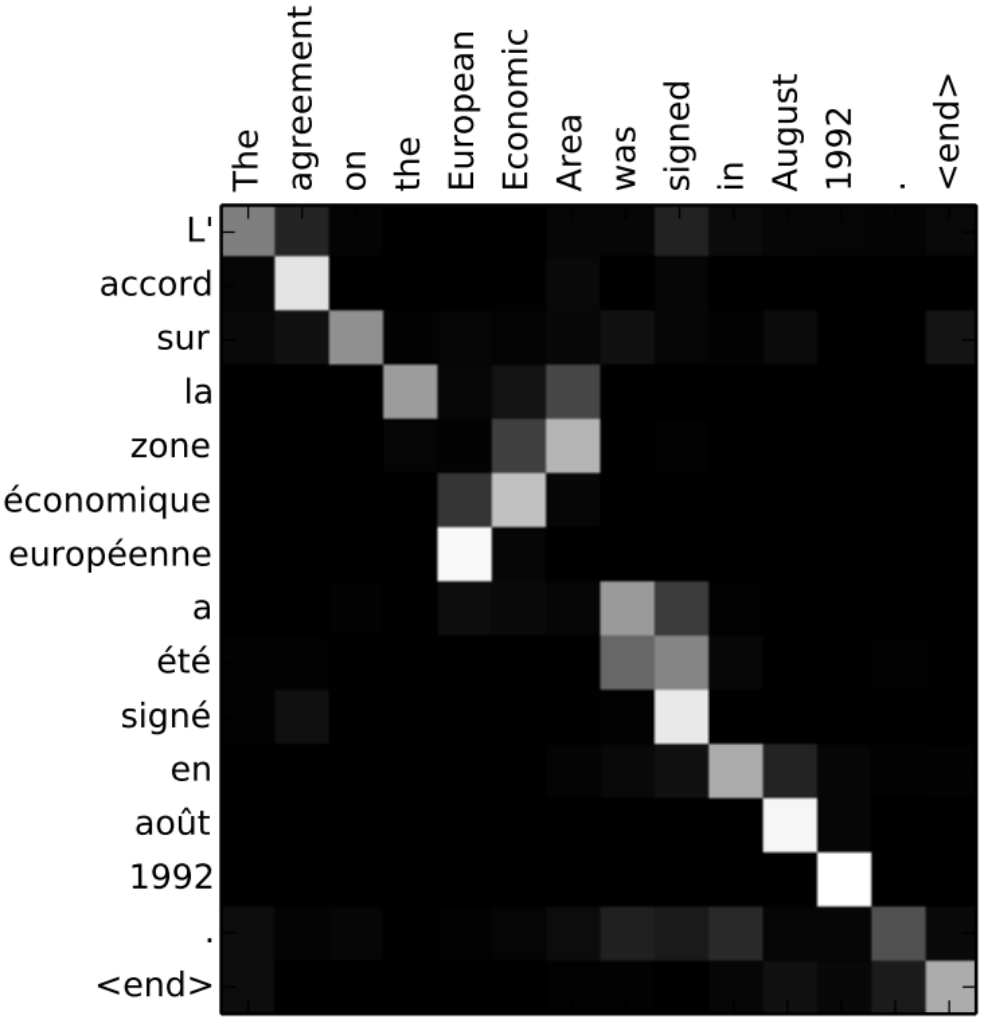


Image credit to Neural Machine Translation by Jointly Learning to Align and Translate

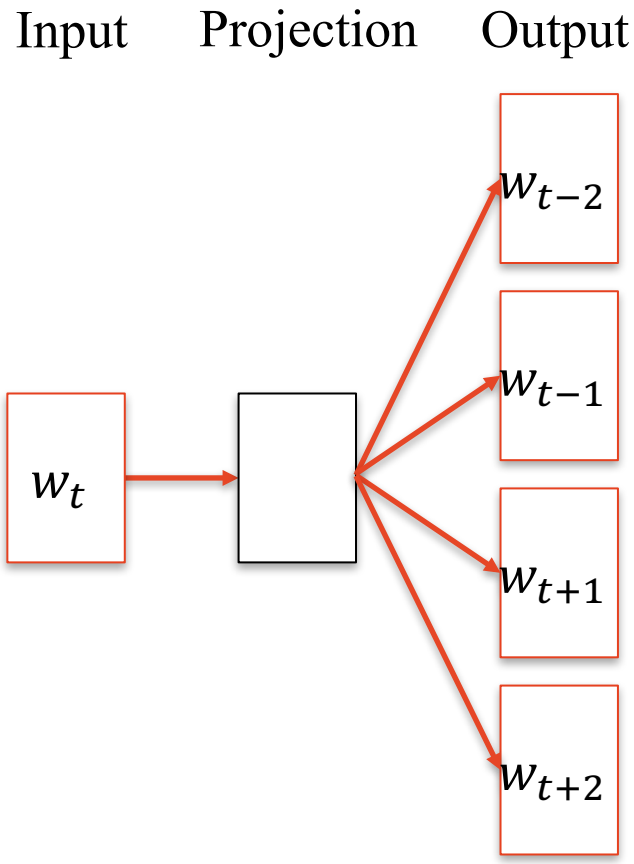
Word Embedding

- Solve dimension disaster of one-hot encoding
- Contain some semantic information
  - vec("Berlin") - vec("Germany") + vec("France") = vec("Paris")

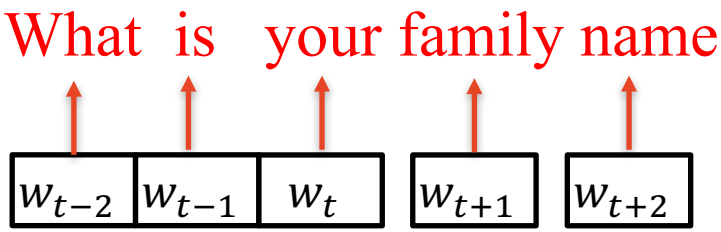
Czech + currency	Vietnam + capital	German + airlines	Russian + river	French + actress
koruna	Hanoi	airline Lufthansa	Moscow	Juliette Binoche
Check crown	Ho Chi Minh City	carrier Lufthansa	Volga River	Vanessa Paradis
Polish zolty	Viet Nam	flag carrier Lufthansa	upriver	Charlotte Gainsbourg
CTK	Vietnamese	Lufthansa	Russia	Cecile De

Four closest tokens to the sum of two vectors

Word Embedding



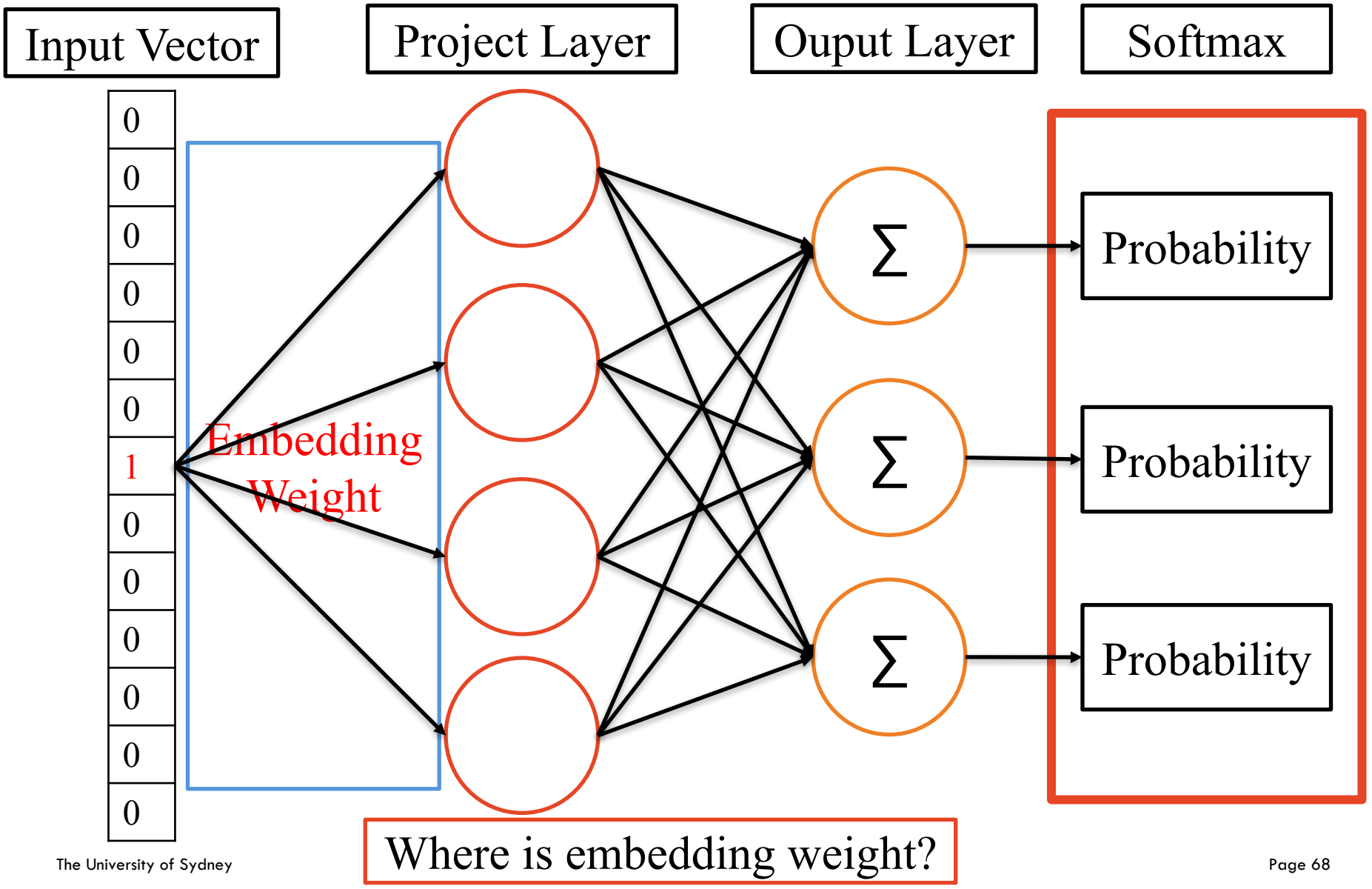
Skip-gram Model



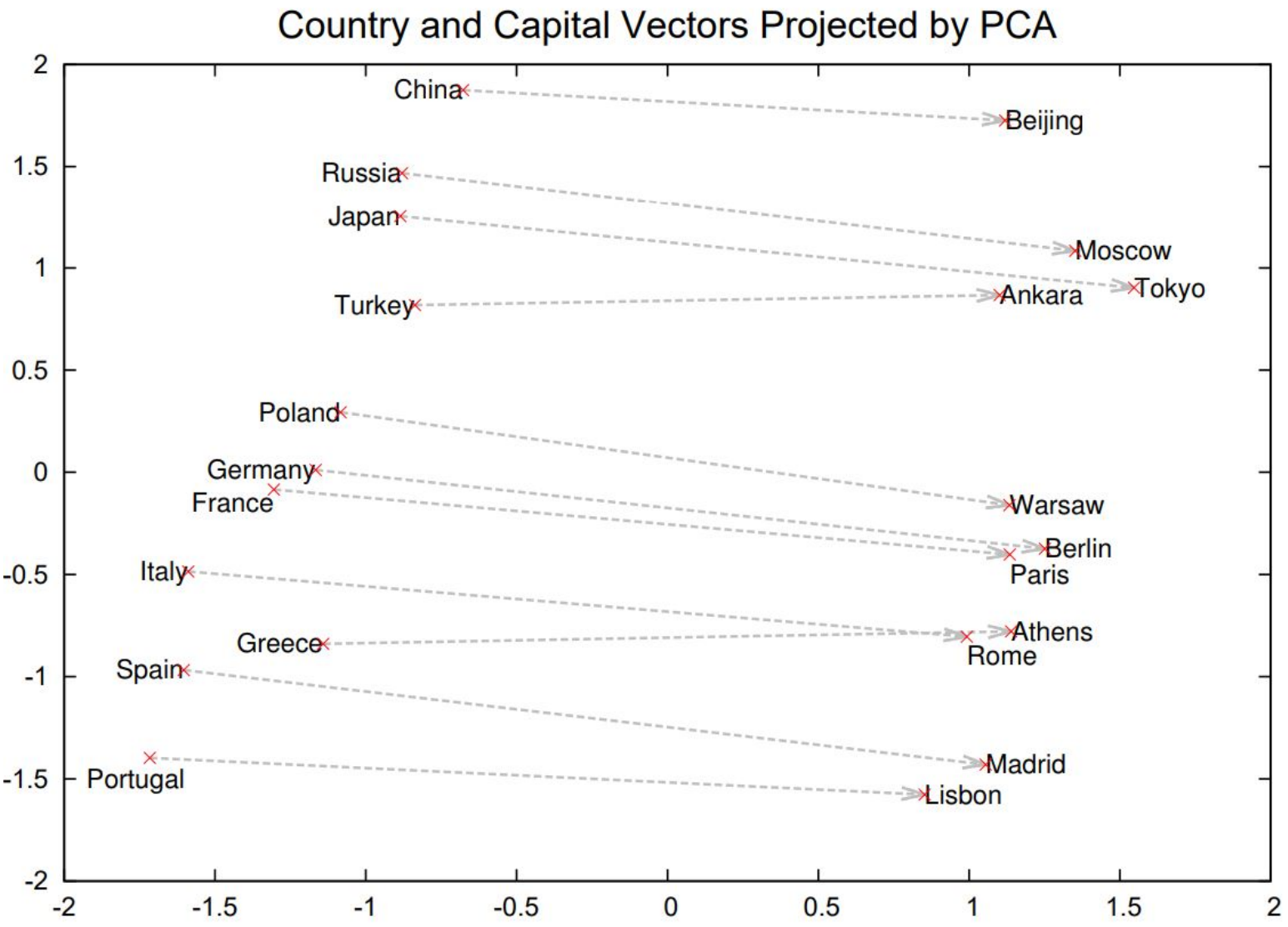
$$\max \frac{1}{5} \sum_{t=1}^{5} \sum_{-2 \le j \le 2, j \ne 0} \log p(w_{t+j} | w_t)$$

Distributed Representations of Words and Phrases and their Compositionality

Word Embedding



Word Embedding



BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding

Jacob Devlin **Ming-Wei Chang** **Kenton Lee** **Kristina Toutanova**
Google AI Language
{jacobdevlin,mingweichang,kentonl,kristout}@google.com

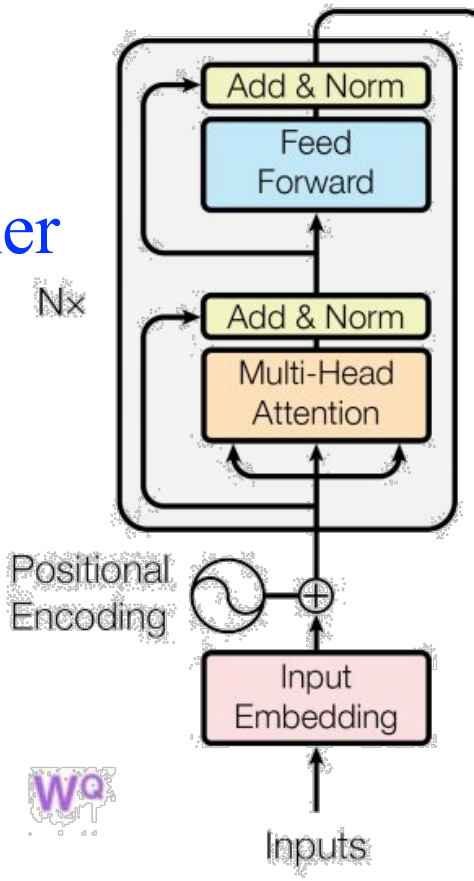
SQuAD1.1 Leaderboard

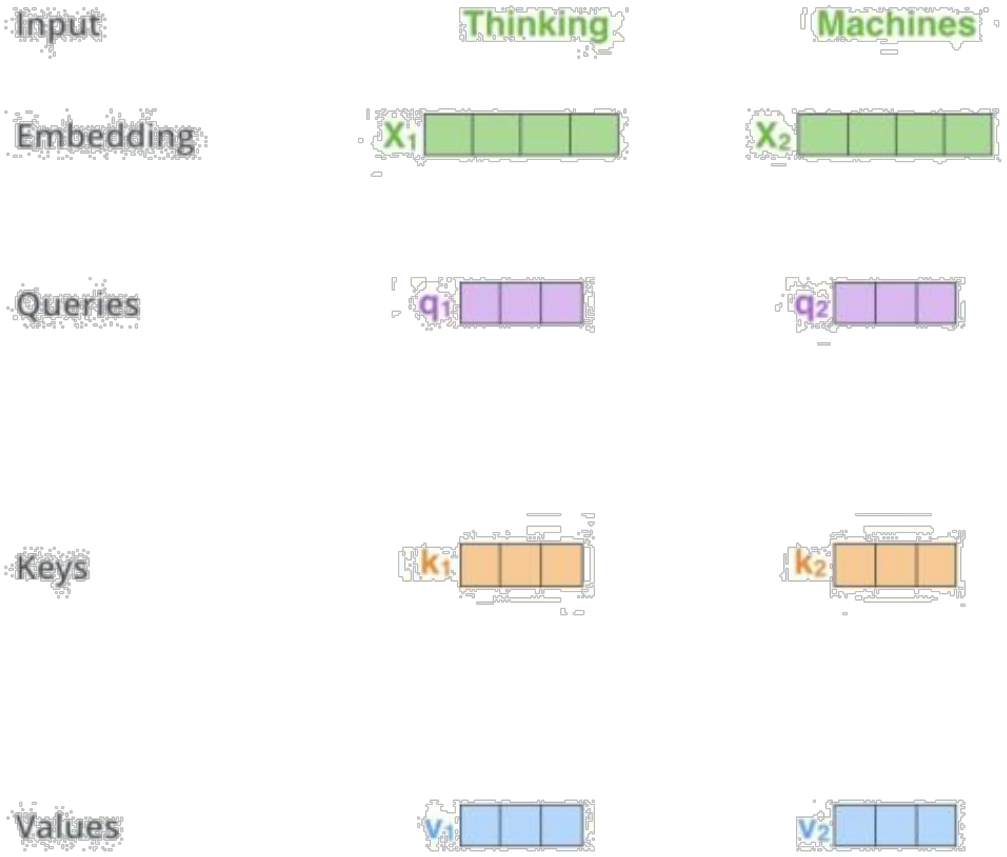
Rank	Model	EM	F1
	Human Performance *Stanford University* (Rajpurkar et al. '16)	82.304	91.221
1 Oct 05, 2018	BERT (ensemble) *Google AI Language* https://arxiv.org/abs/1810.04805	**87.433**	**93.160**
2 Sep 09, 2018	nlnet (ensemble) *Microsoft Research Asia*	85.356	91.202
3 Jul 11, 2018	QANet (ensemble) *Google Brain & CMU*	84.454	90.490

NLP's ImageNet moment

Attenion Is All You Need

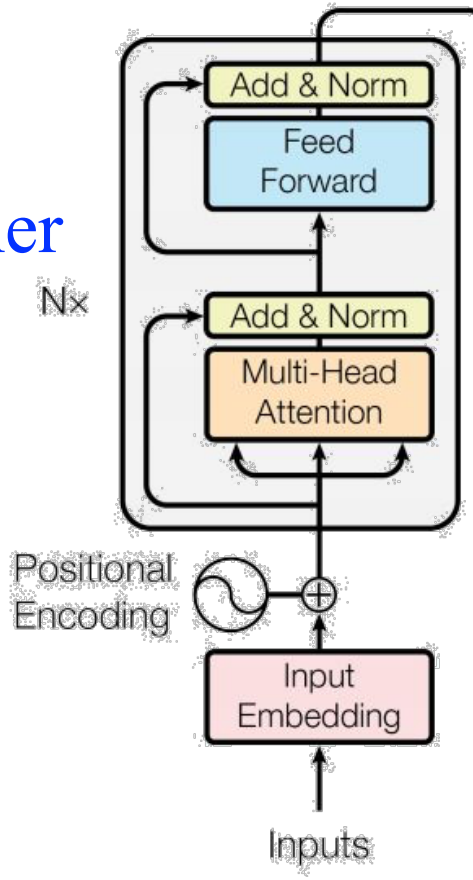
Transformer

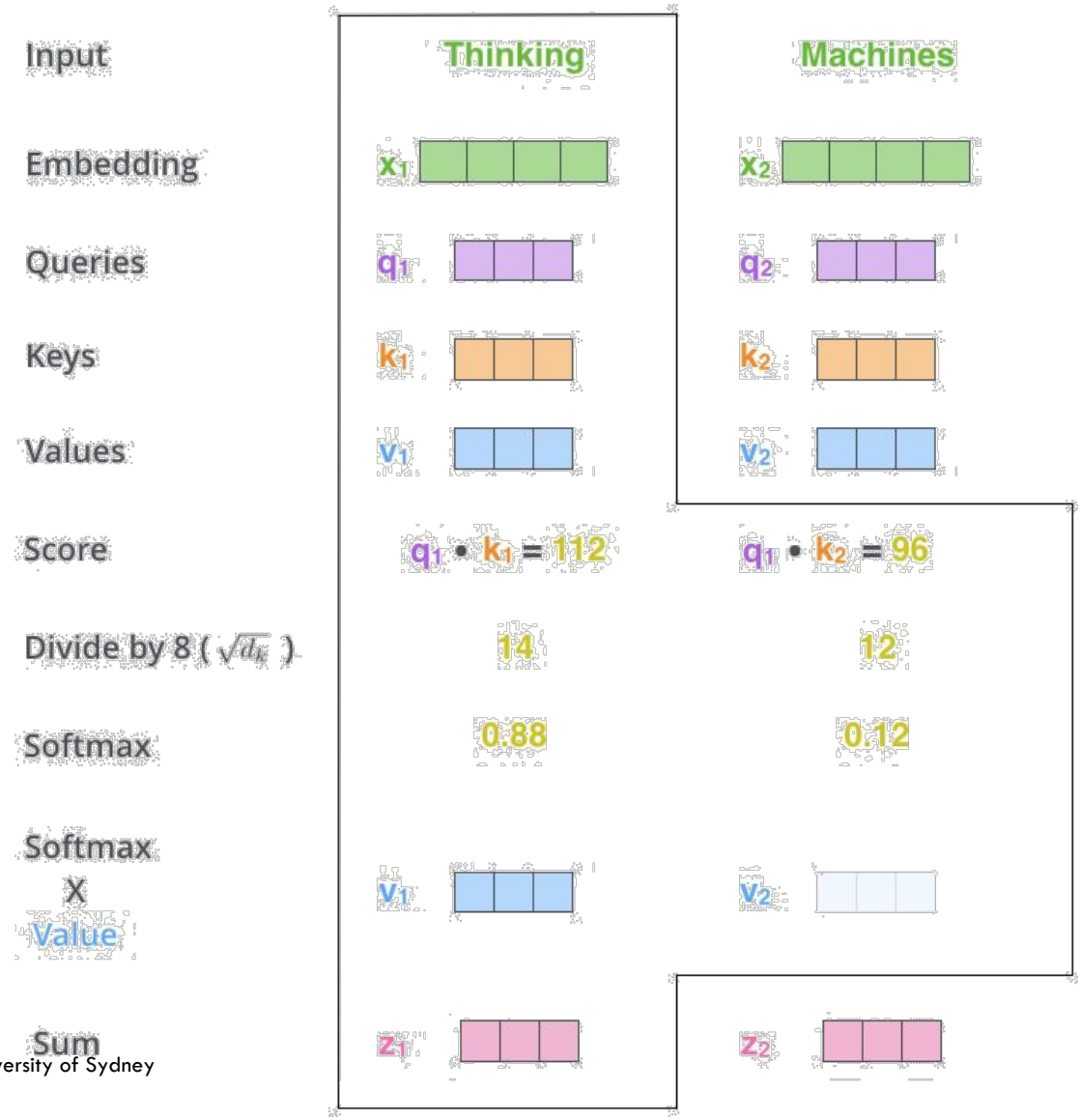




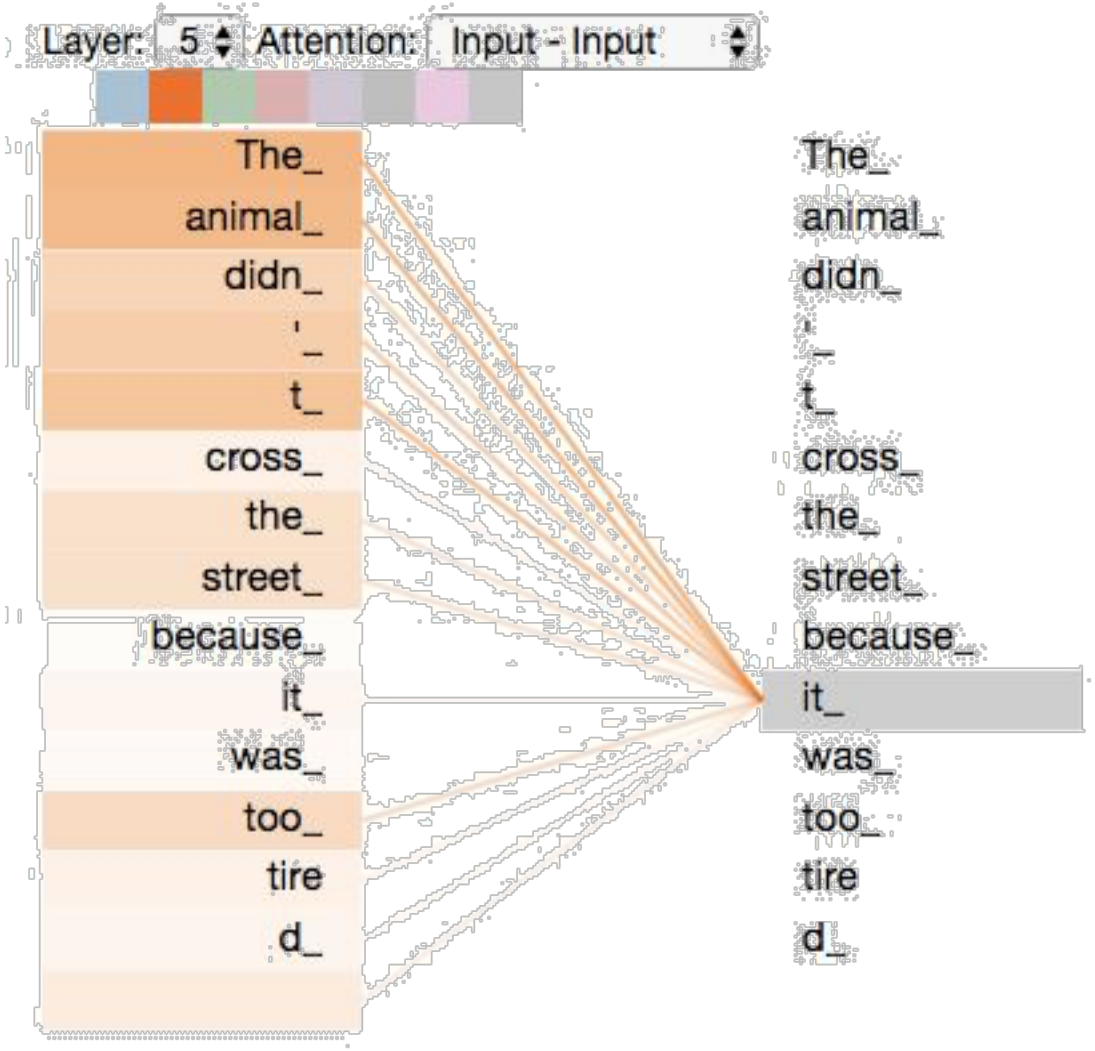
Attenion Is All You Need

Transformer

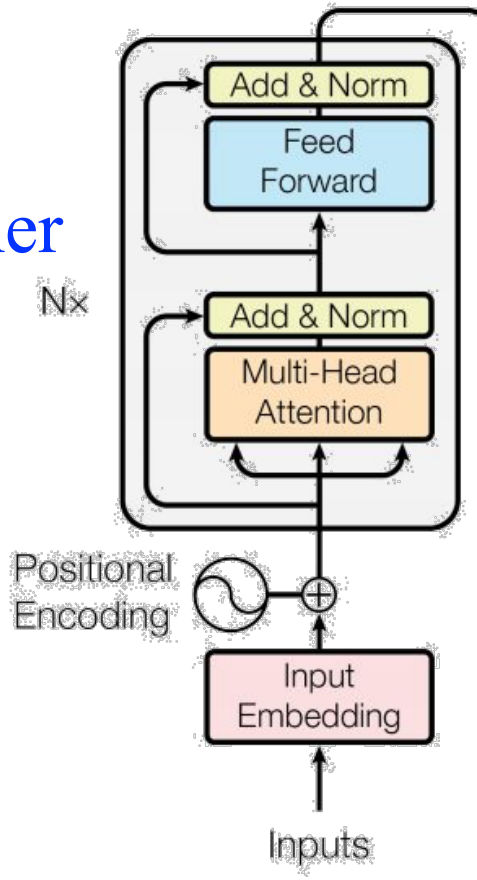




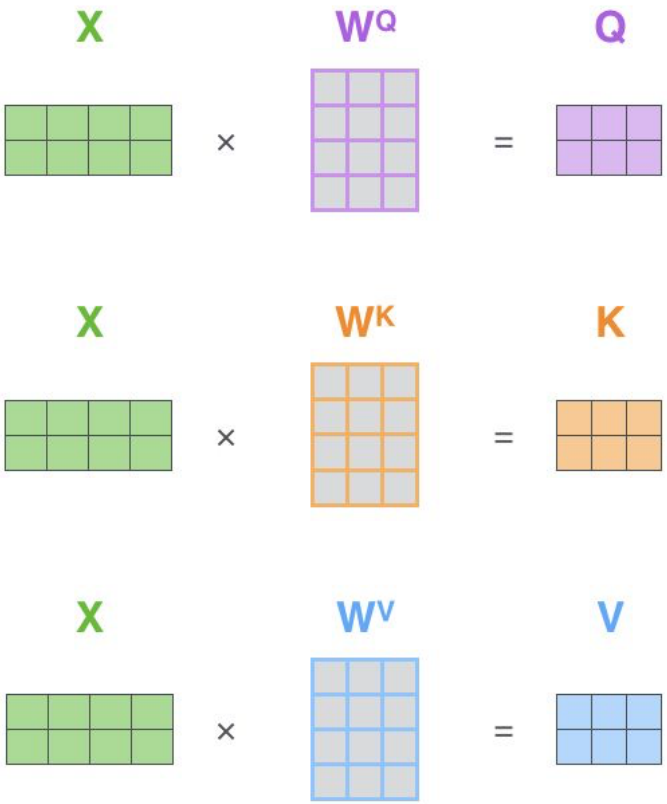
Attenion Is All You Need



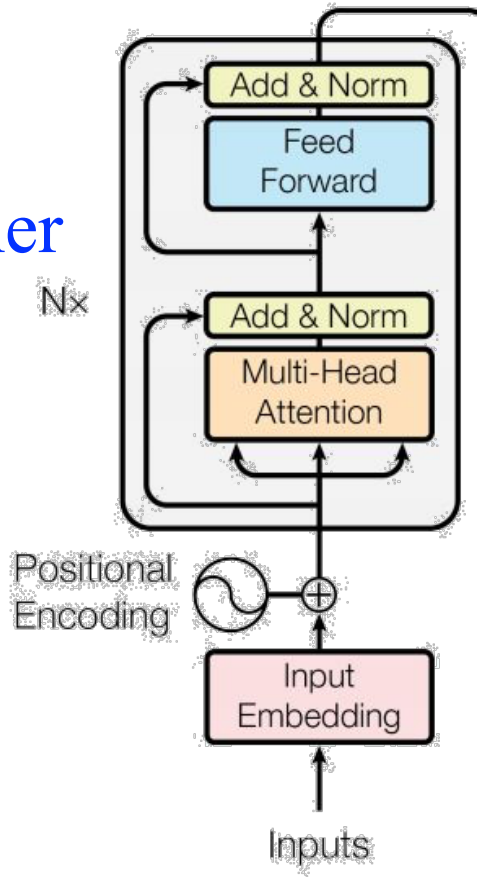
Transformer



Attenion Is All You Need



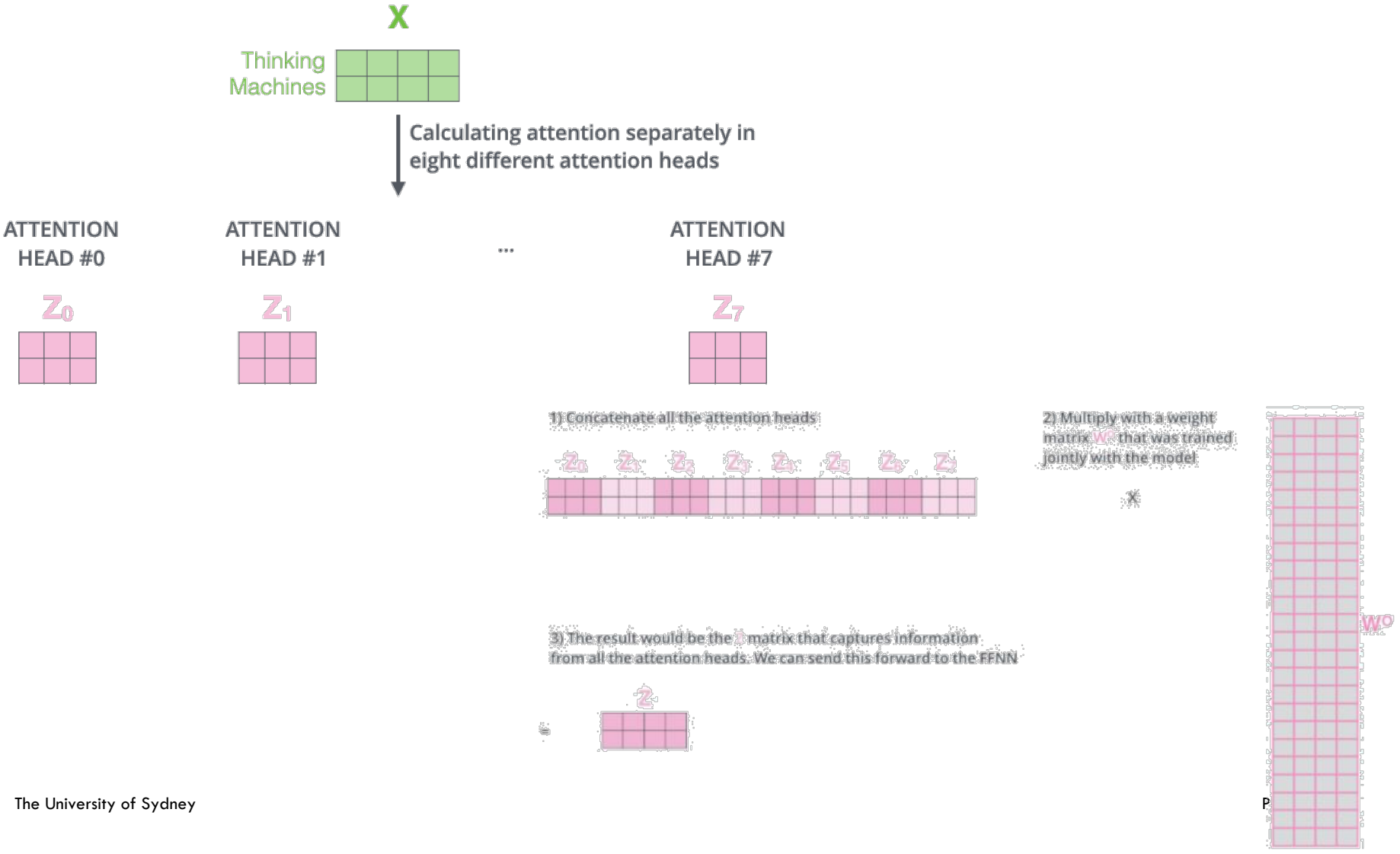
Transformer



$$\text{softmax}\left(\frac{Q \times K^T}{\sqrt{d_k}}\right) V = Z$$

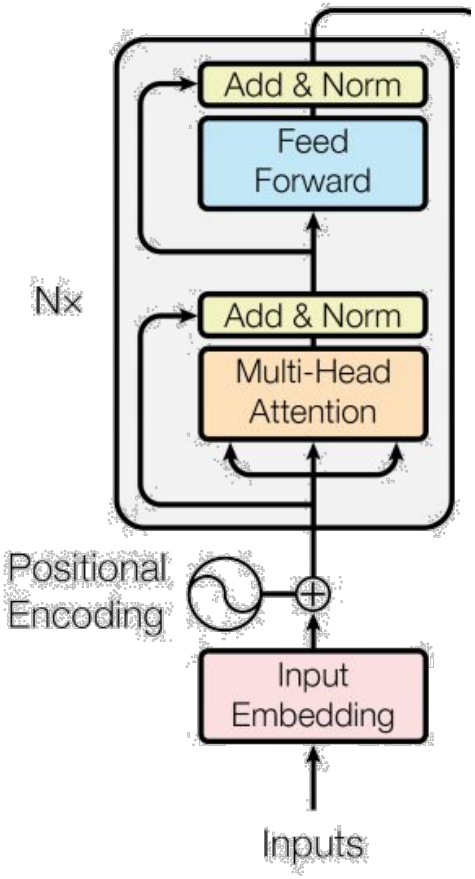
Attenion Is All You Need

Multiple heads

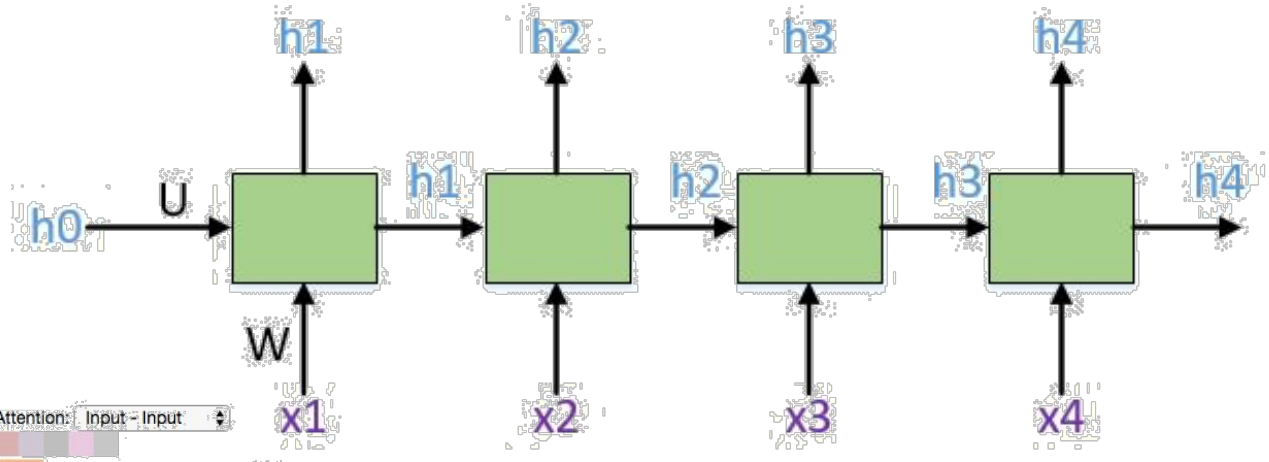


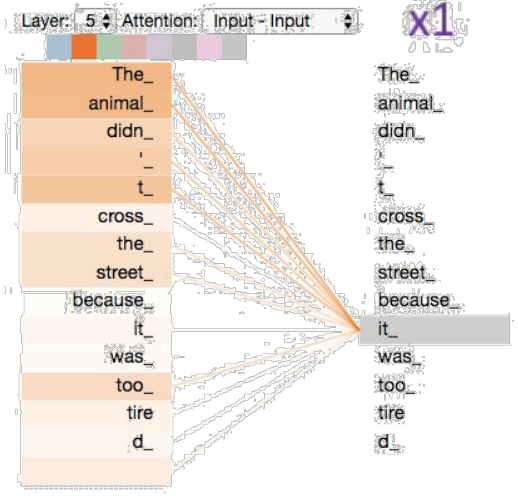
Attenion Is All You Need

Transformer



Recurrent Neural Networks

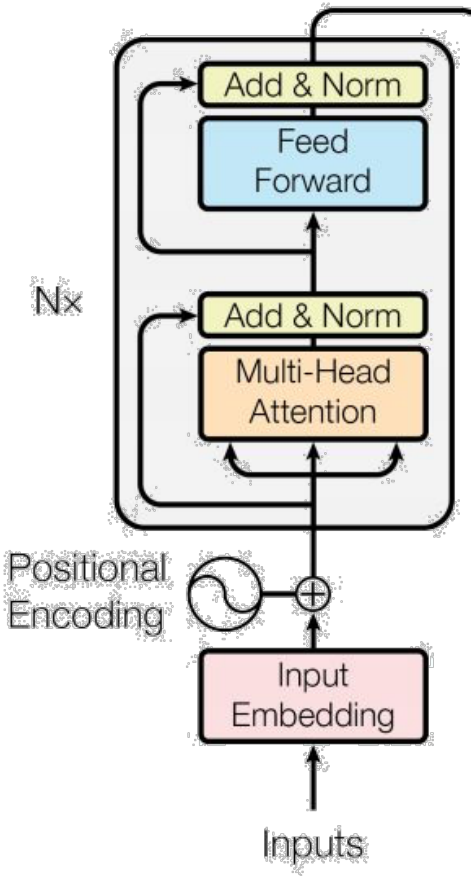




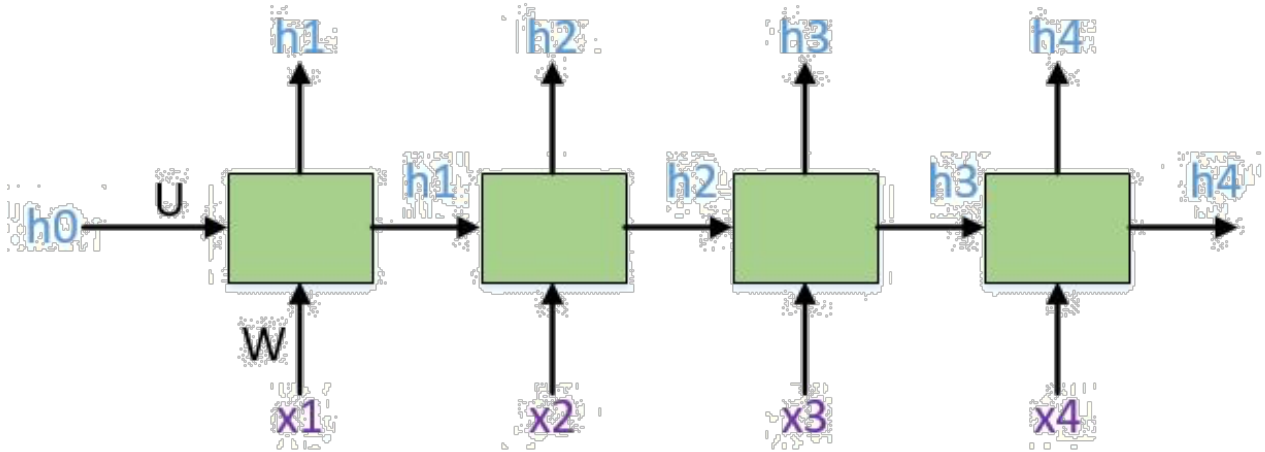
1. Distance
2. Context
3. Sequence

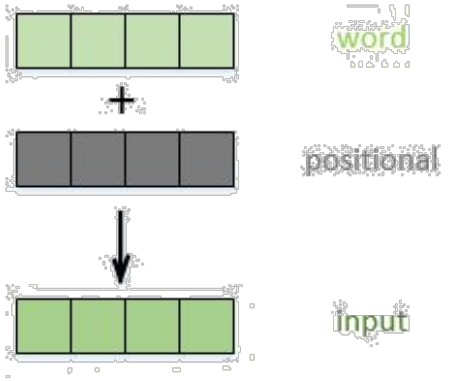
Attenion Is All You Need

Transformer



Recurrent Neural Networks

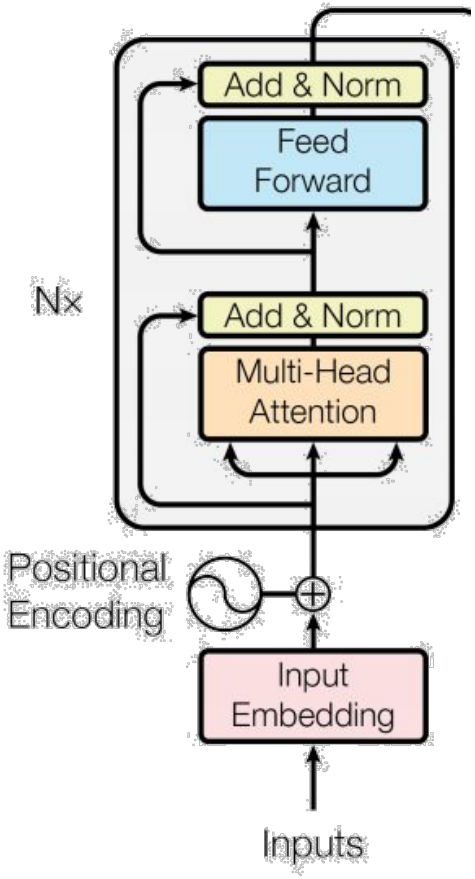


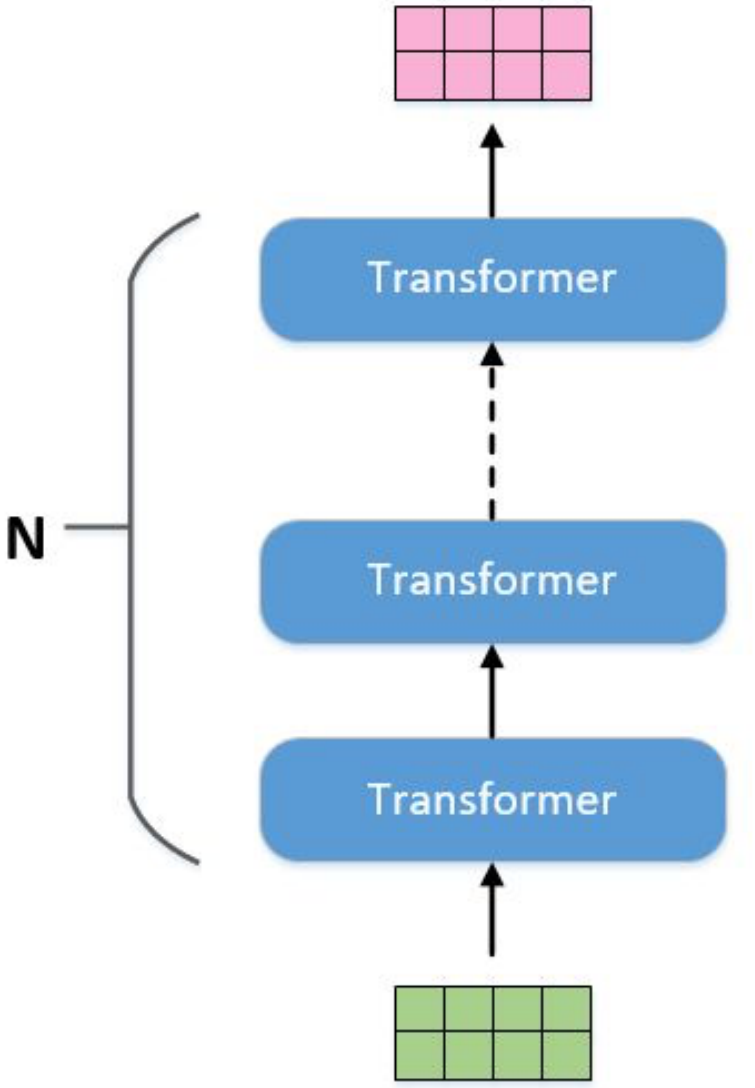


1. Distance
2. Context
3. Sequence

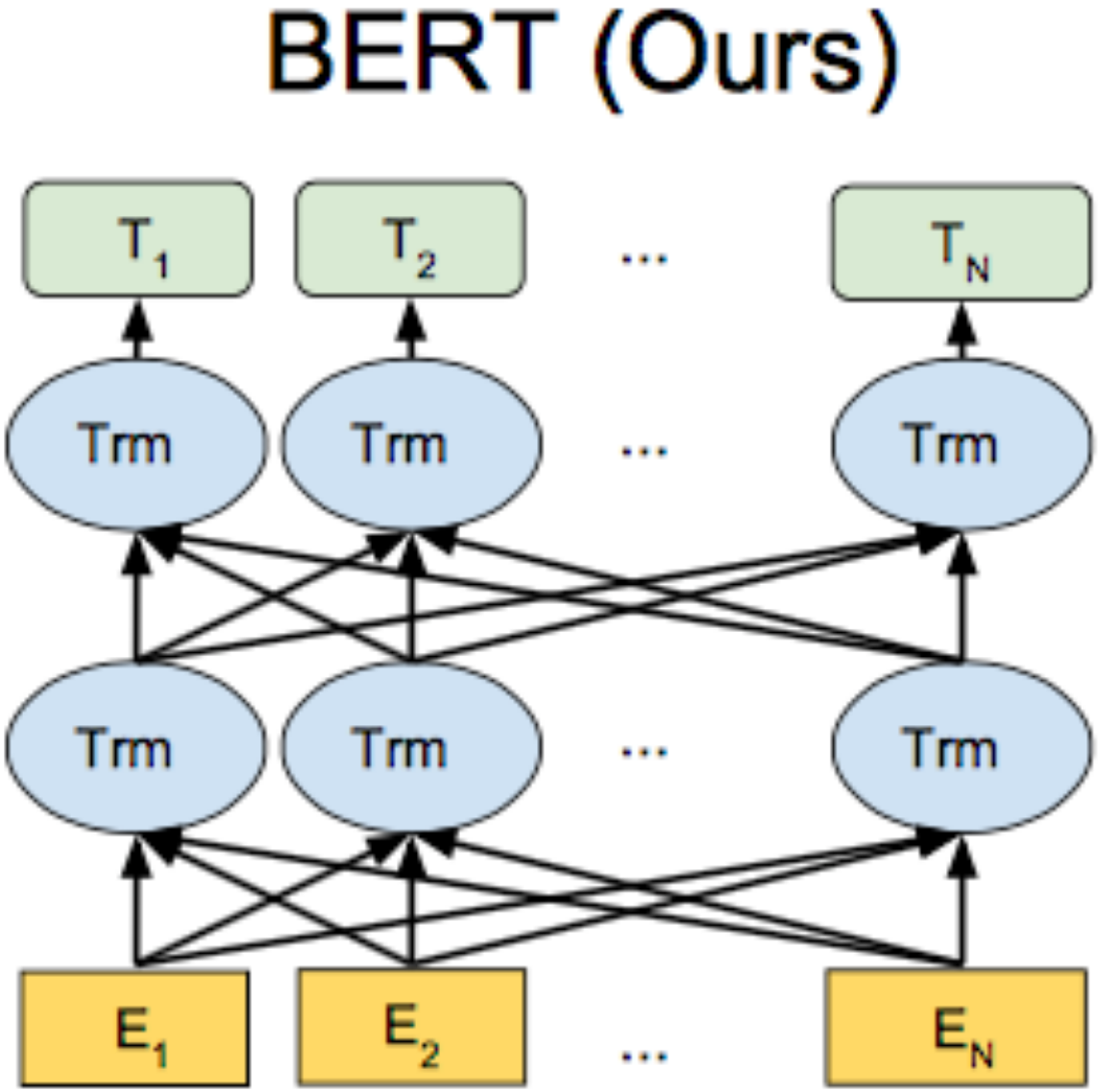
Attenion Is All You Need

Transformer





BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding



BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding

Task #1: Masked LM

We take a straightforward approach of masking some percentage of the input tokens at random, and then predicting only those masked tokens.

my dog is hairy → my dog is [MASK]

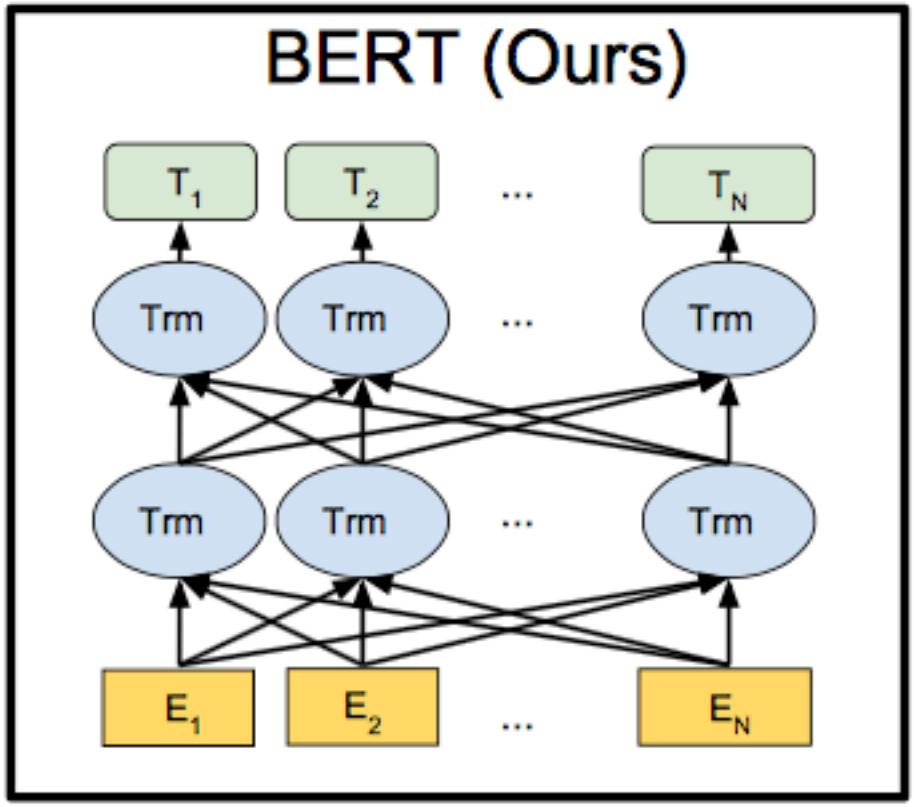
Task #2: Next Sentence Prediction

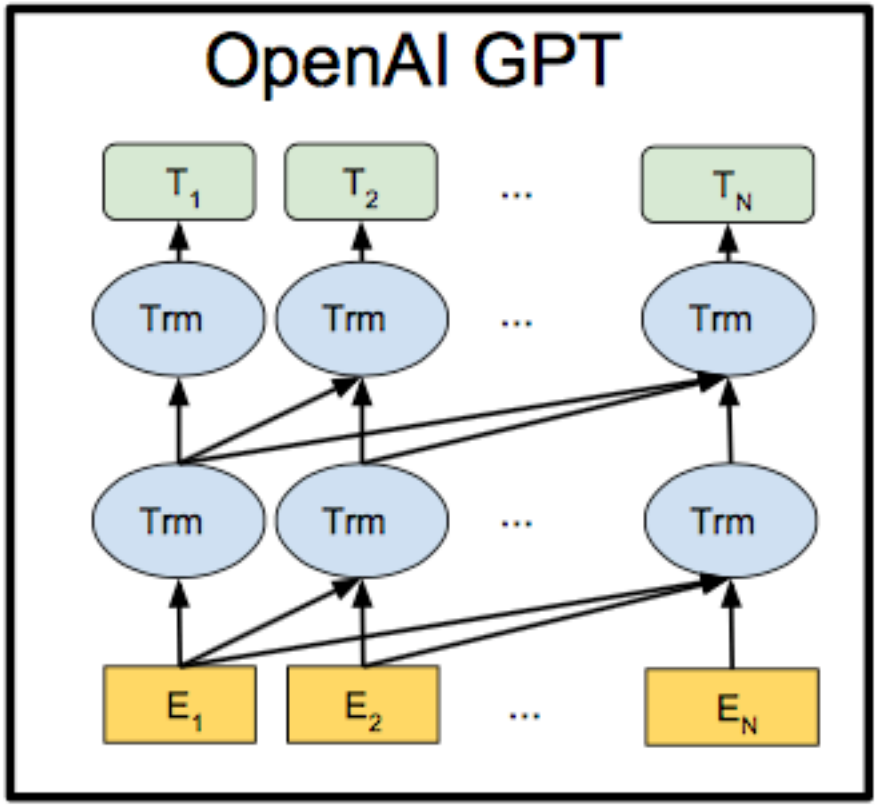
In order to train a model that understands sentence relationships, we pre-train a binarized *next sentence prediction* task .

Input = [CLS] the man went to [MASK] store [SEP]
he bought a gallon [MASK] milk [SEP]
Label = IsNext

Input = [CLS] the man [MASK] to the store [SEP]
penguin [MASK] are flight ##less birds [SEP]
Label = NotNext

BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding





1. Bidiectional