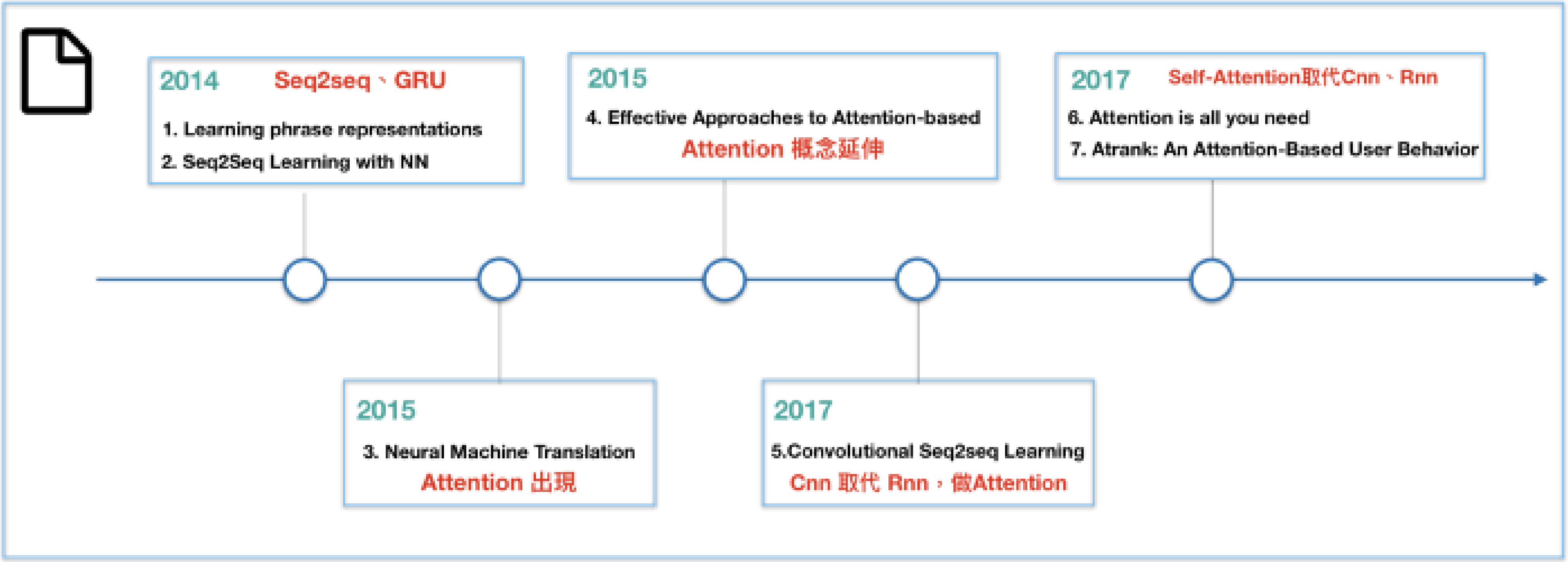
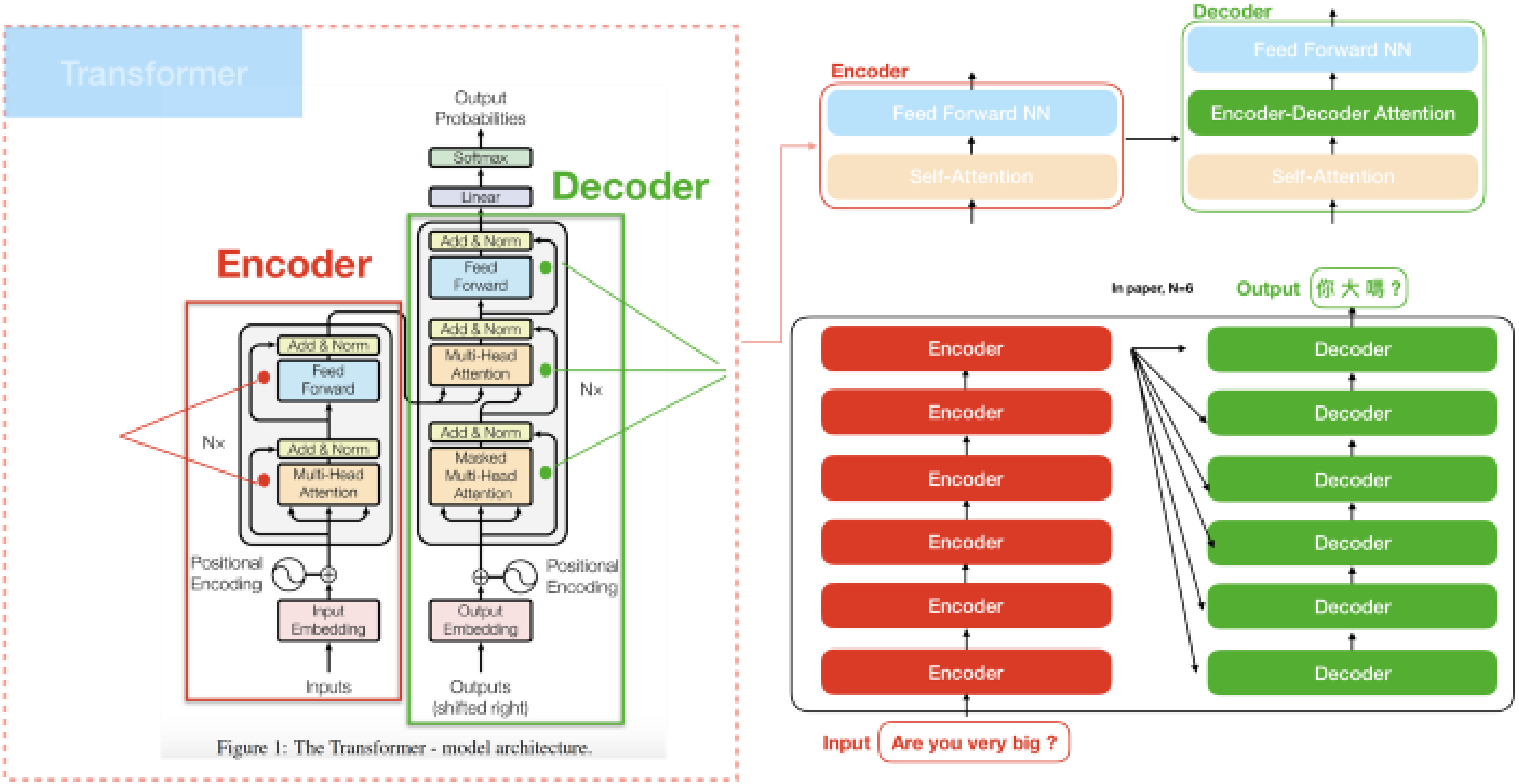
Part.3

Transformer & BERT

History

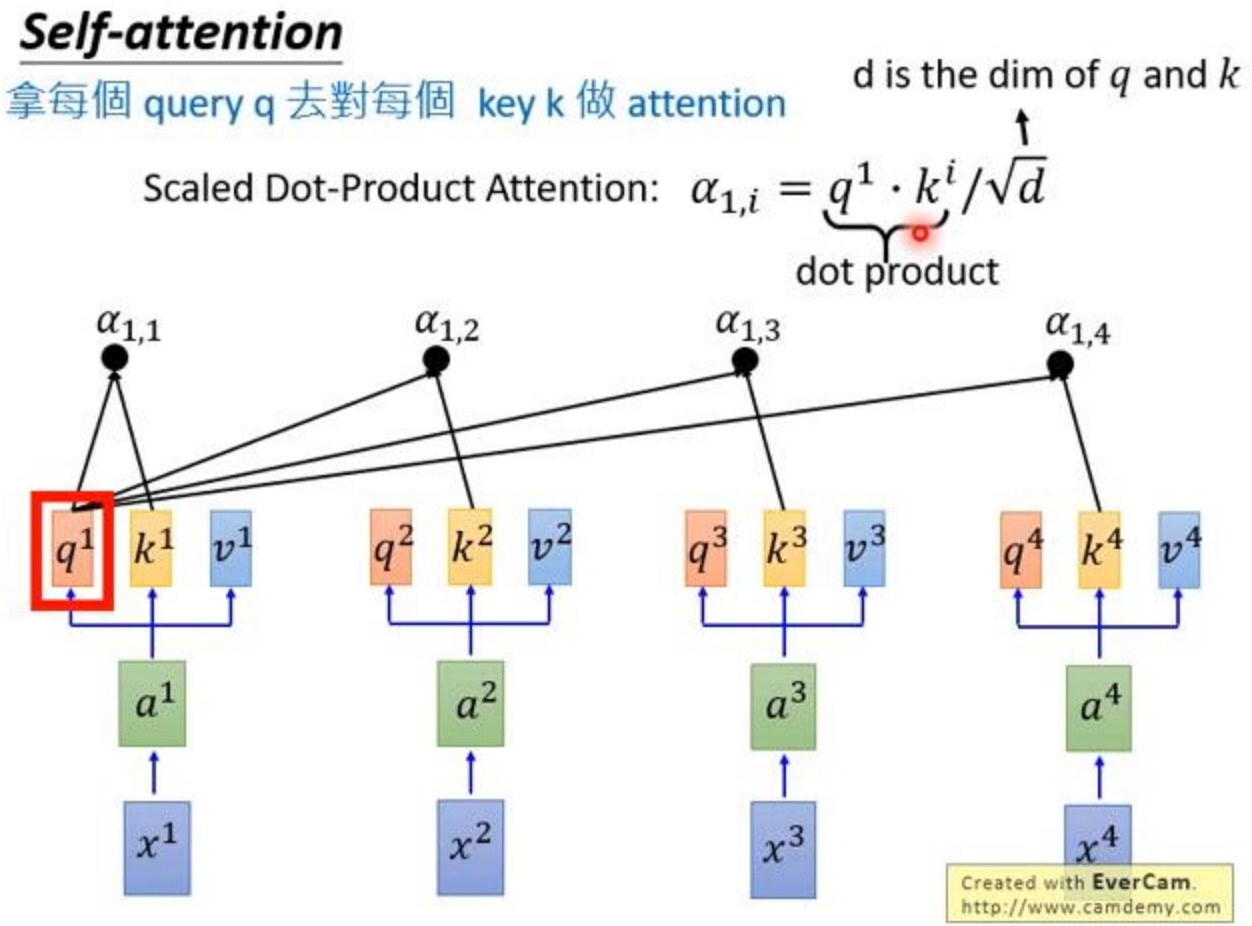

2014 Seq2seq、GRU
1. Learning phrase representations
2. Seq2Seq Learning with NN
2015
3. Neural Machine Translation
Attention 出現
2015
4. Effective Approaches to Attention-based
Attention 概念延伸
2017
5.Convolutional Seq2seq Learning
Cnn 取代 Rnn，做Attention
2017 Self-Attention取代Cnn、Rnn
6. Attention is all you need
7. Atrank: An Attention-Based User Behavior

Transformer

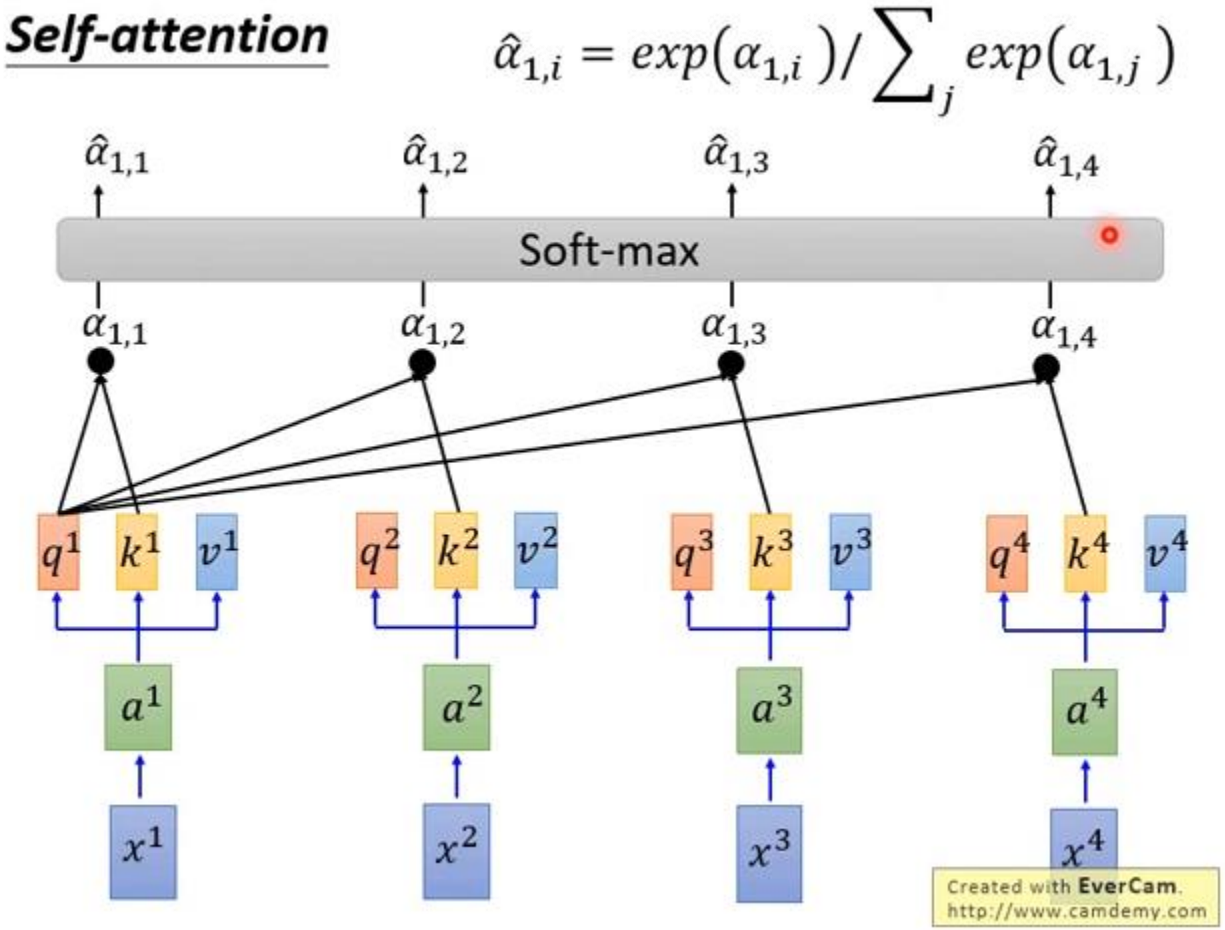


Scaled Dot-Product Attention

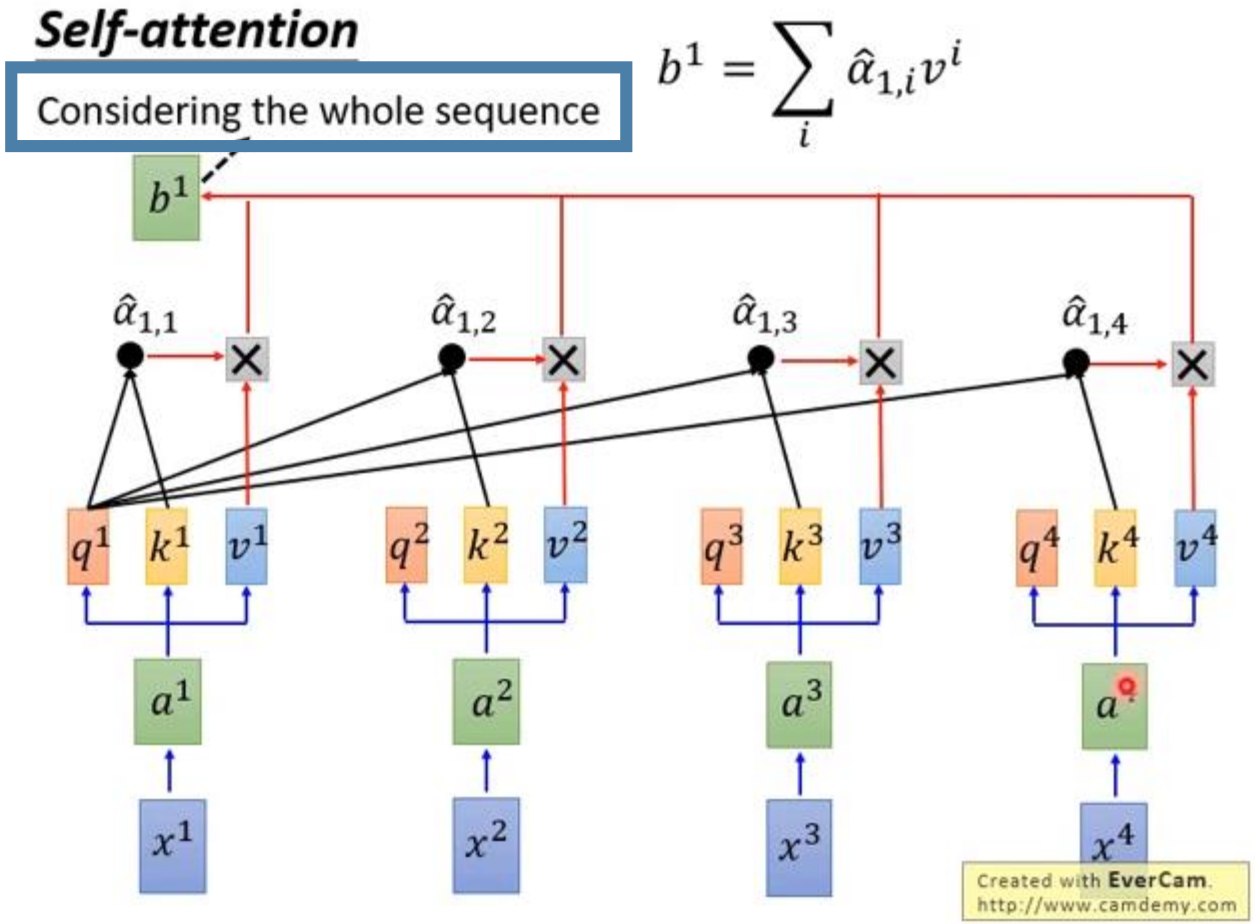
- **表達Q跟K的匹配程度**



- **取softmax得到 Attention score**

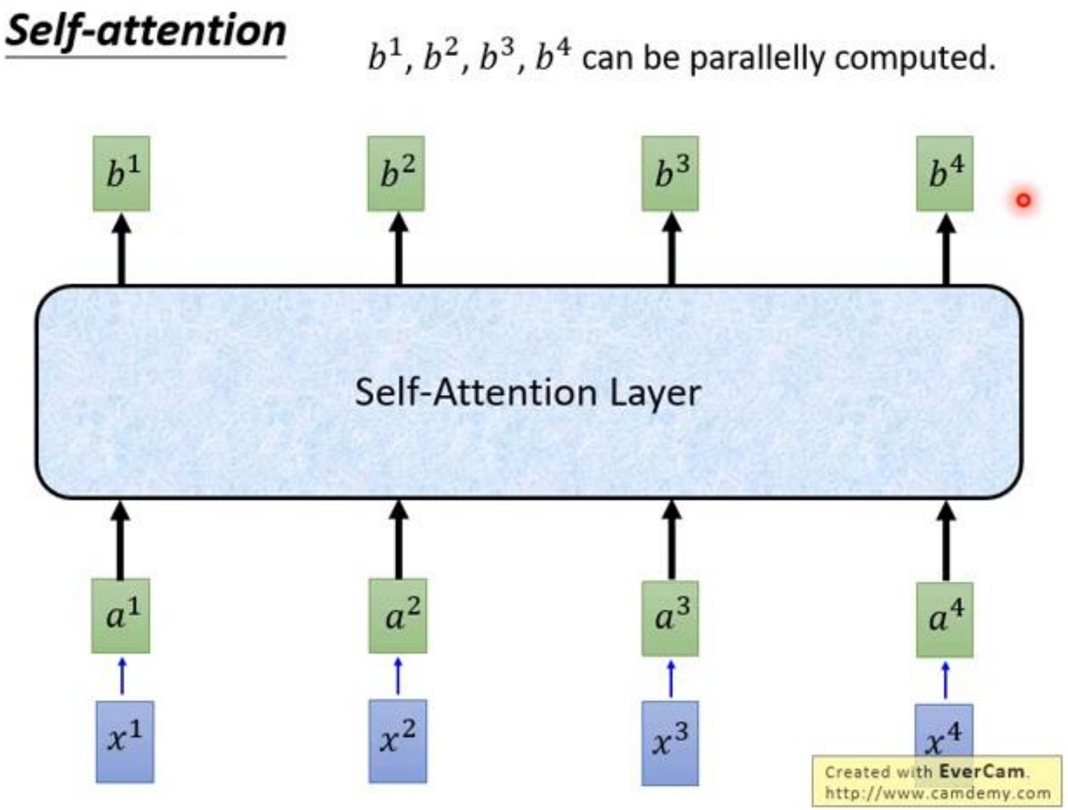


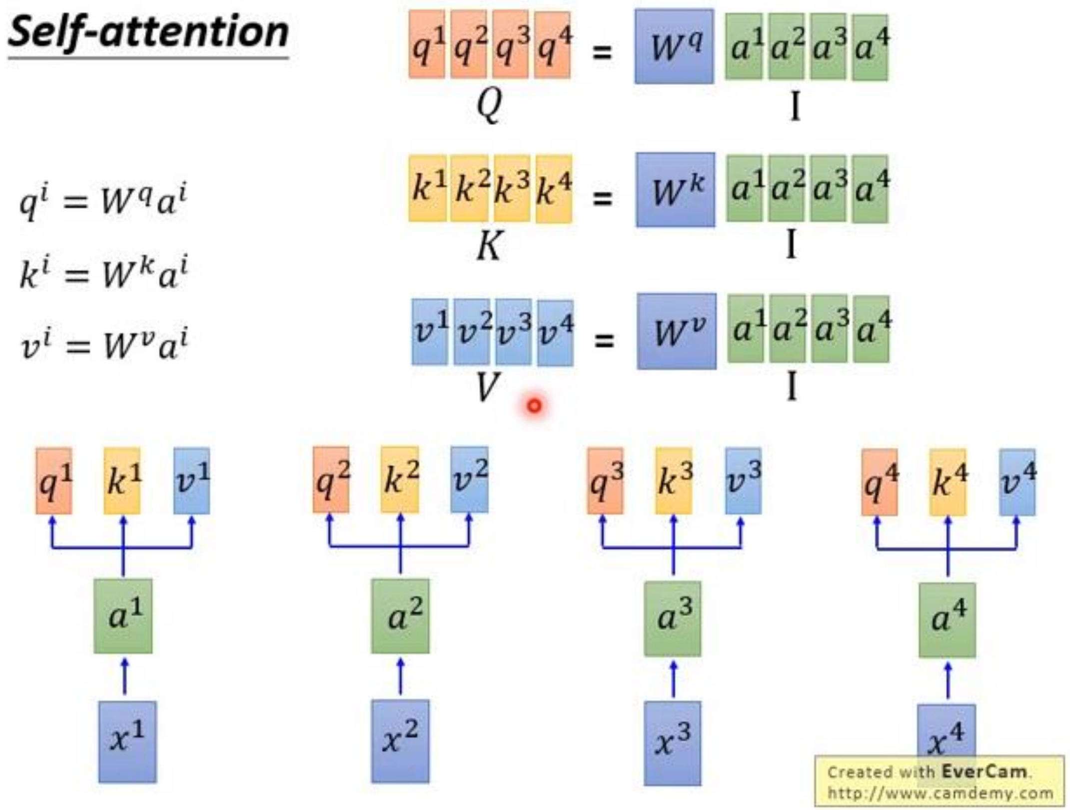
- **Weighted sum**



Scaled Dot-Product Attention

- **平行處理 (矩陣運算)**



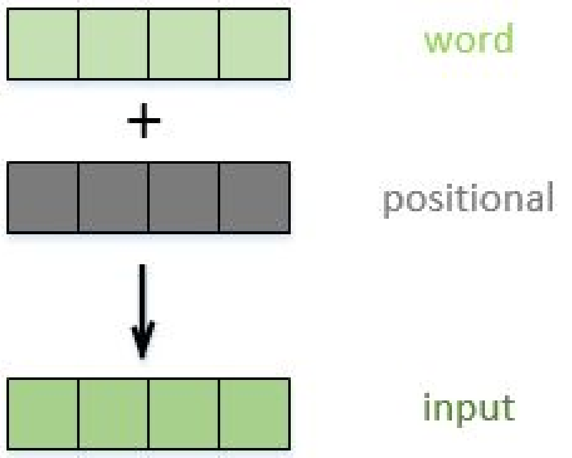


Positional Encoding

- **Self-attention沒有考慮先後順序**
- **額外加入位置的訊息**
- **同時考慮語意和詞在句字中的位置**
- **PE公式：**

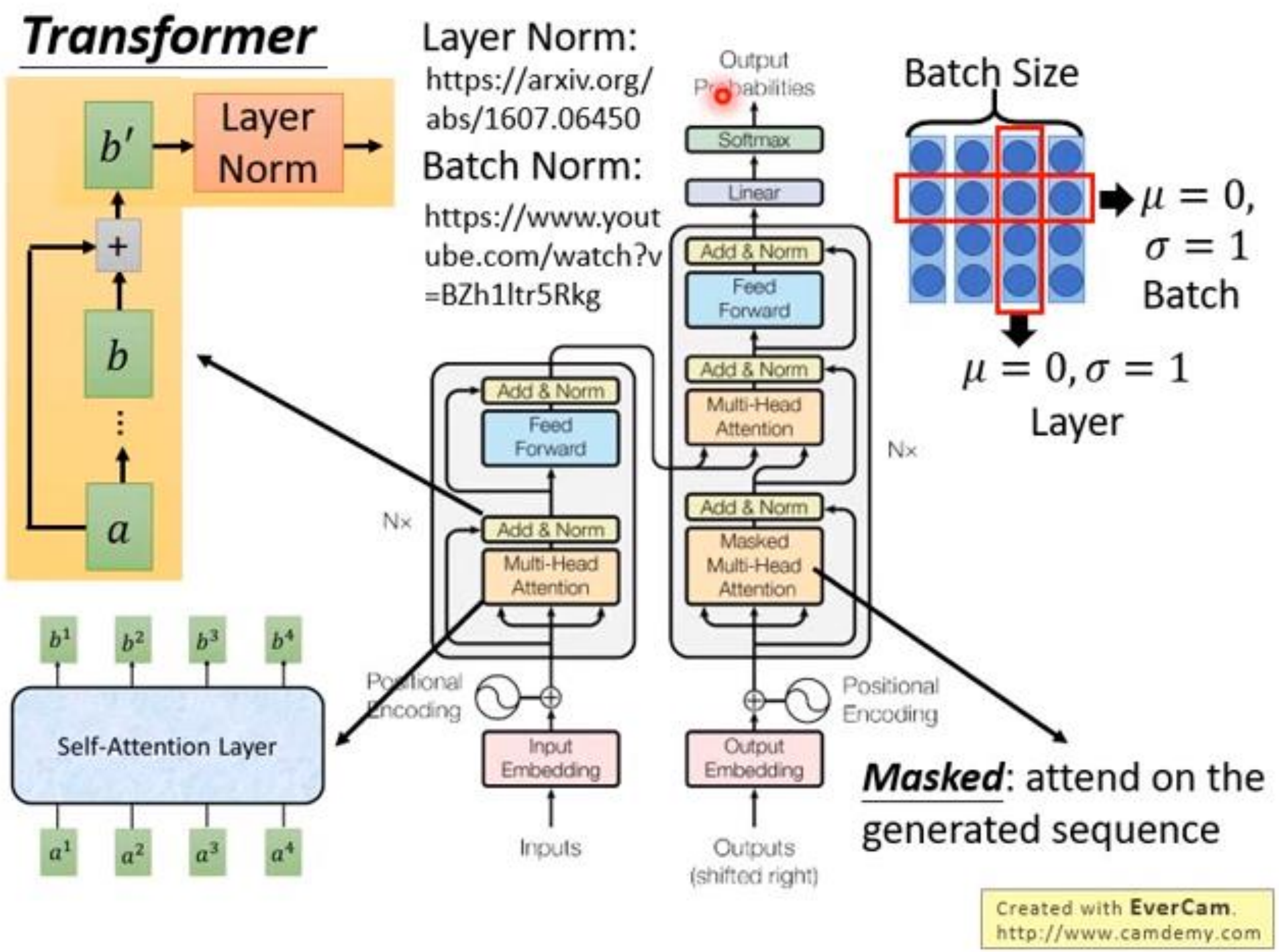
$$PE_{(pos,2i)} = sin(pos/10000^{2i/d_{model}})$$

$$PE_{(pos,2i+1)} = cos(pos/10000^{2i/d_{model}})$$



https://leemeng.tw/neural-machine-translation-with-transformer-and-tensorflow2.html

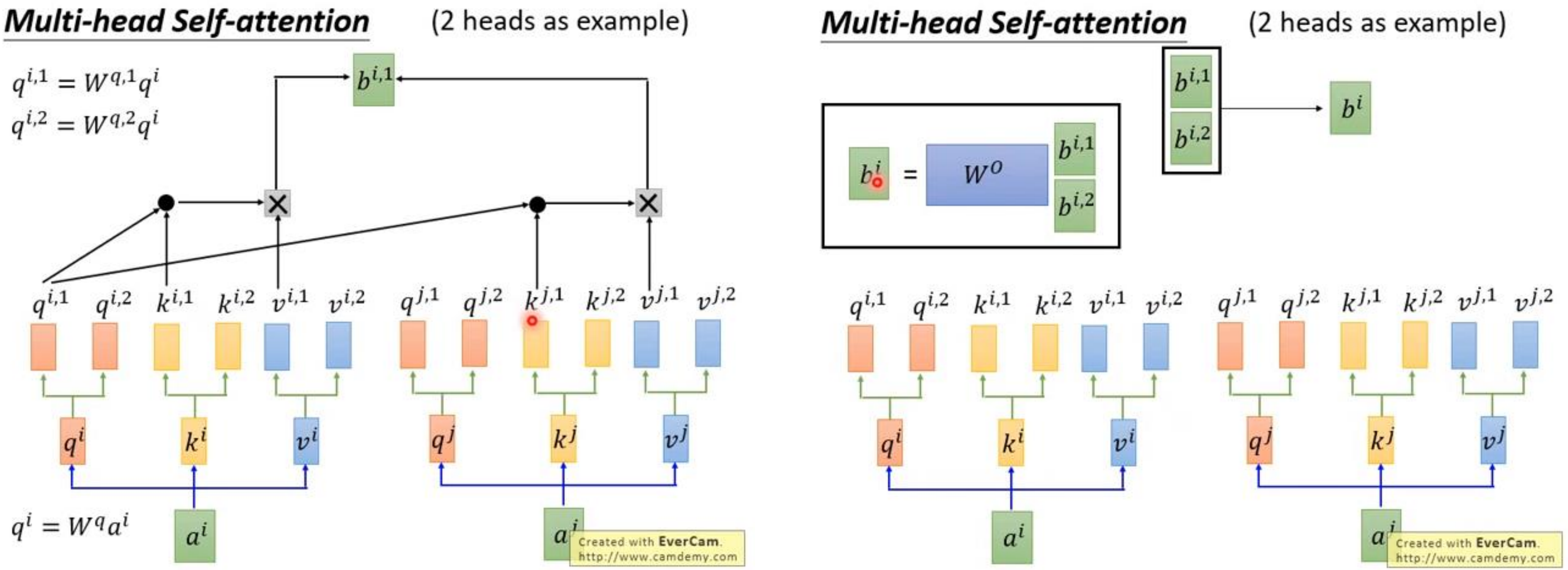
Transformer



Multi-Head Attention

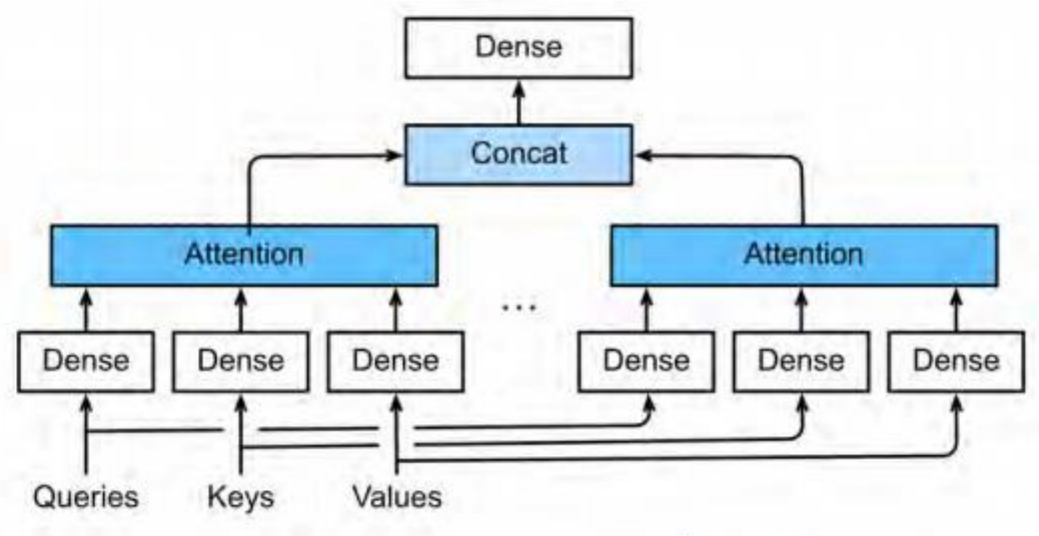
Multi-Head Attention

- **分裂q, k, v (head能各自關注不同重點)**



Multi-Head Attention

Q: query
K: key
V: value



$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$

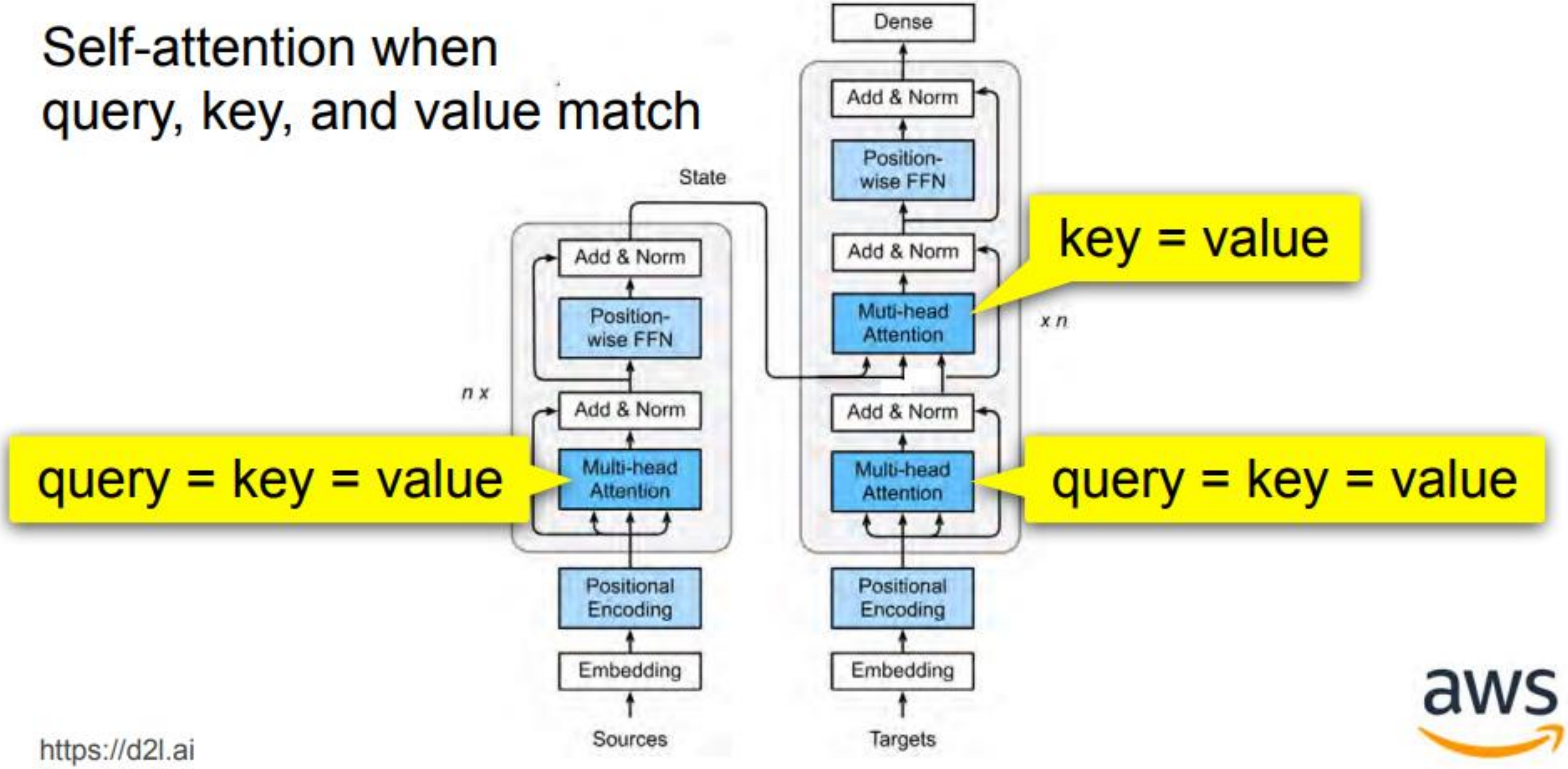
$$\text{MultiHead}(Q, K, V) = \text{Concat}(\text{head}_1, \ldots \text{head}_h)W^O$$

$$\text{where head}_i = \text{Attention}\left(QW_i^Q, KW_i^K, VW_i^V\right)$$

https://d2l.ai

Multi-Head Attention

Self-attention when query, key, and value match



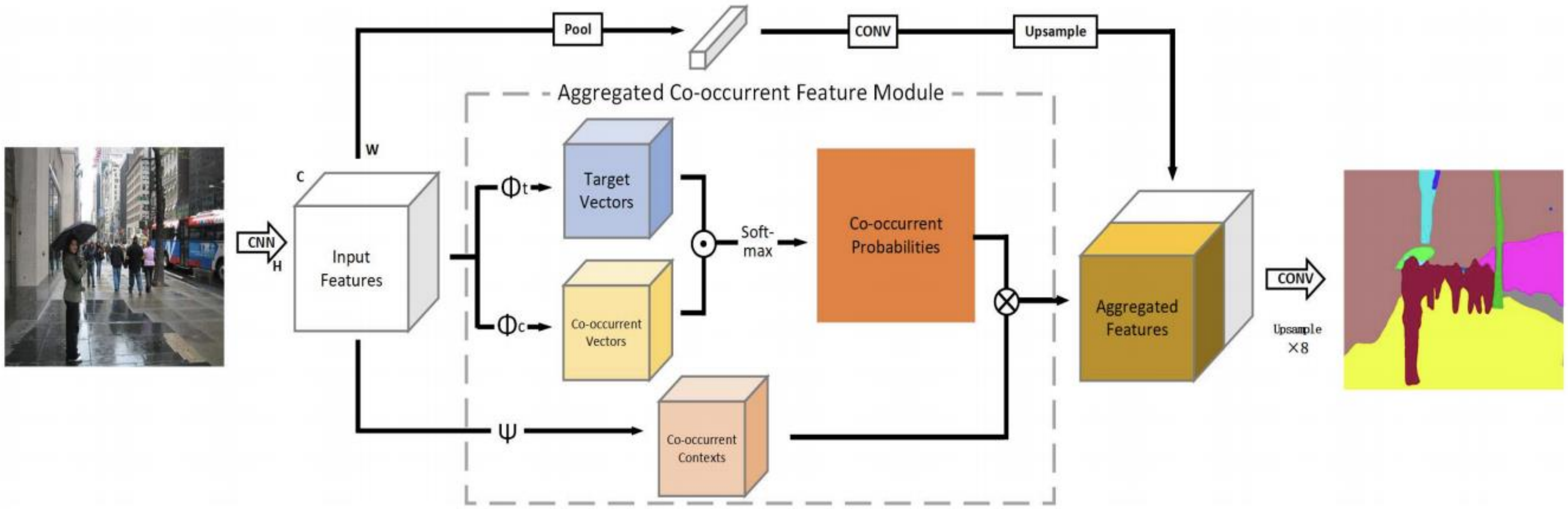
https://d2l.ai

Semantic Segmentation

Semantic Segmentation

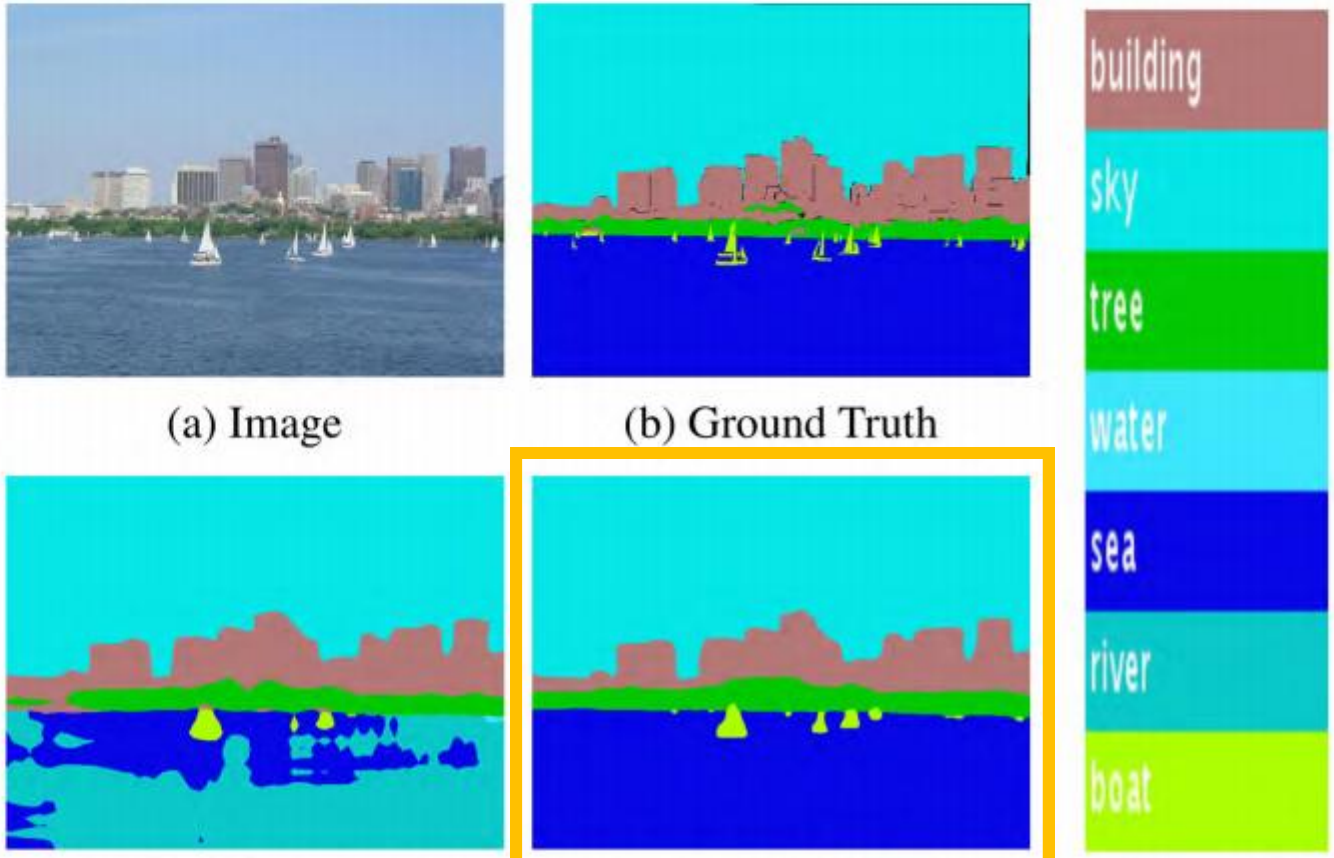
Semantic Segmentation

Multi-head attention for semantic segmentation (Zhang et al., '19)



Multi-Head Attention

Classify pixels co-occurring with boat as sea rather than water



(a) Image (b) Ground Truth (c) FCN (baseline) (d) CFNet (ours) (e) legend

Q & A

- 回顧一下Transformer是甚麼?
- 把Attention換成Self Attention有什麼好處?

BERT
Bidirectional Encoder Representations from Transformers
(Devlin et al, 2018)

SOTA on 11 NLP tasks

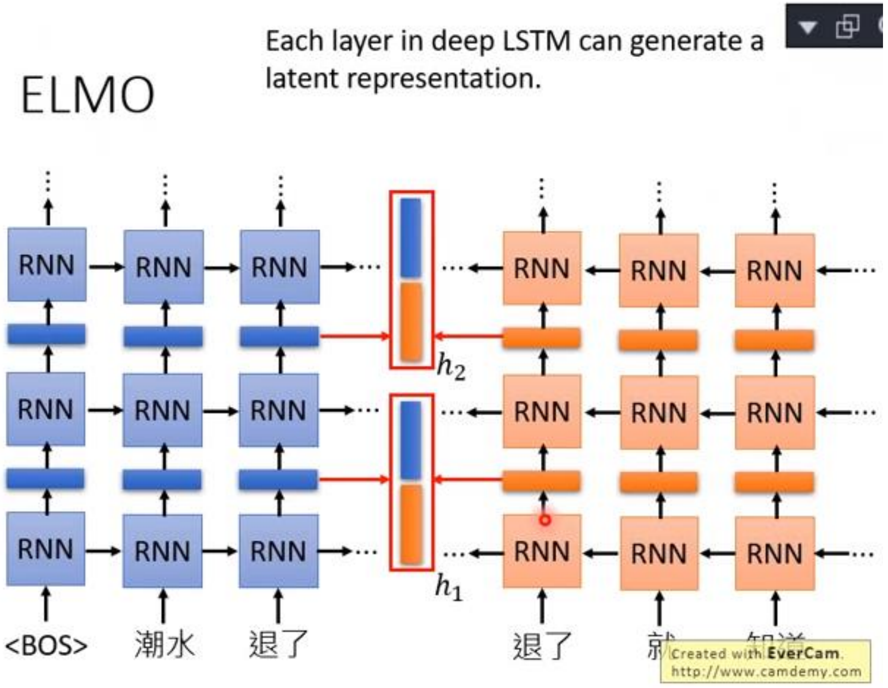
courses.d2l.ai/berkeley-stat-157/index.html

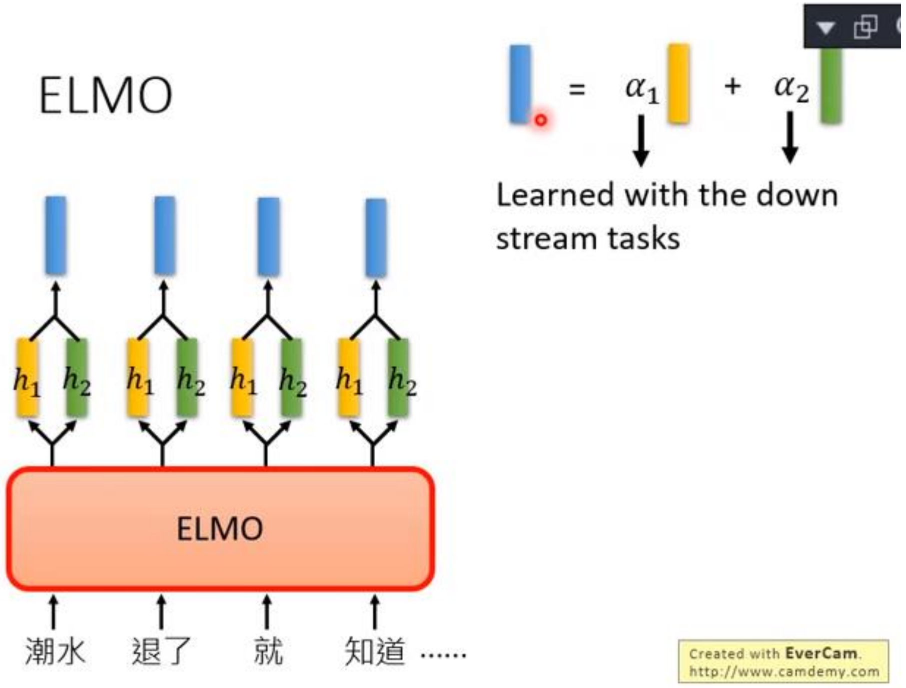
aws

https://leemeng.tw/attack_on_bert_transfer_learning_in_nlp.html

ELMO

- **Embeddings from Language Models (為了解決一詞多義)**
- **BERT 前的 Language Model**



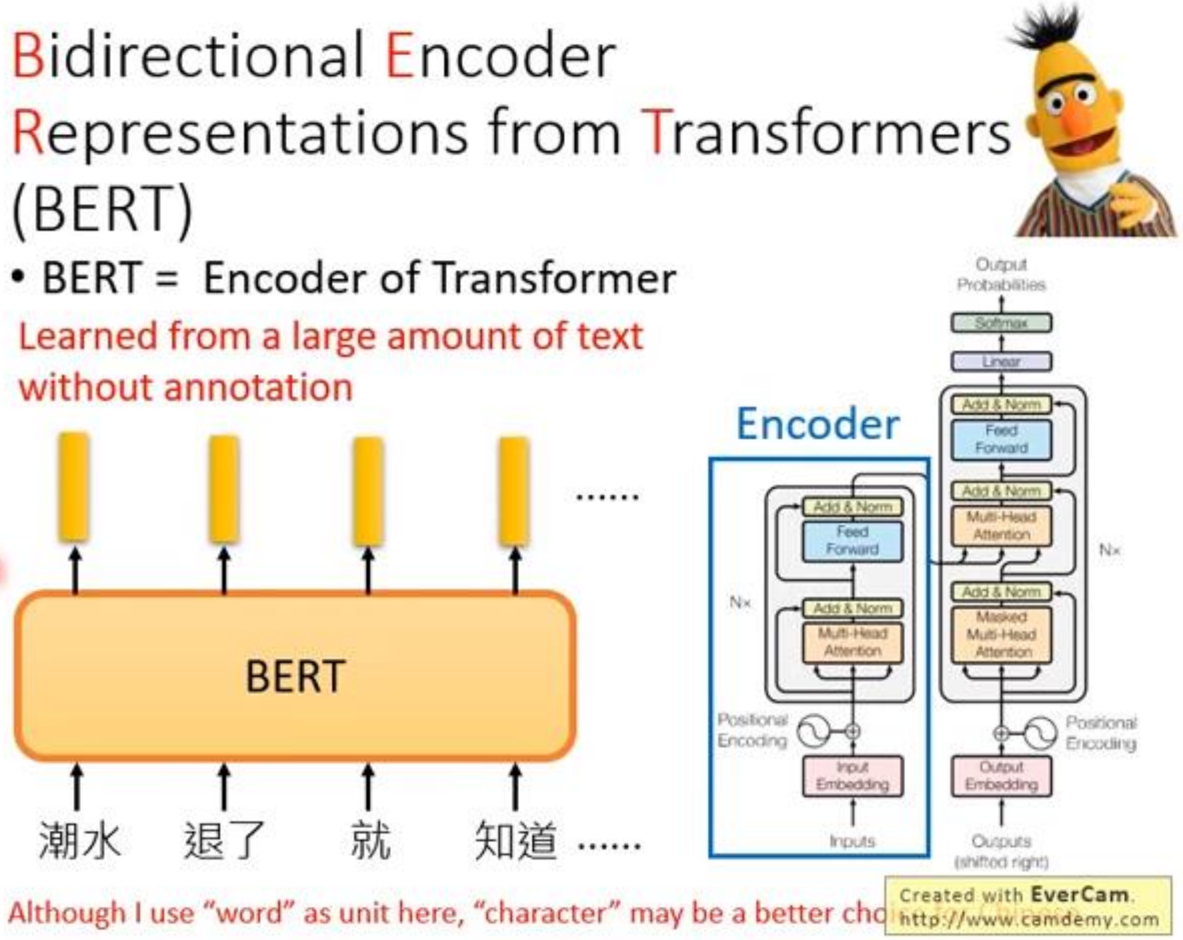


ELMo

- ELMo dynamically determines word embedding in downstream task.
- ELMo generates three embeddings:
  - word embedding
  - 1st LSTM layer embedding
  - 2st LSTM layer embedding
- Pre-training -> get three embeddings (v_1, v_2, v_3) per word.
- Fine tuning -> freeze embeddings and train weights (w_1, w_2, w_3) for (v_1, v_2, v_3) per word.
- The final embedding is $w_1 v_1 + w_2 v_2 + w_3 v_3$

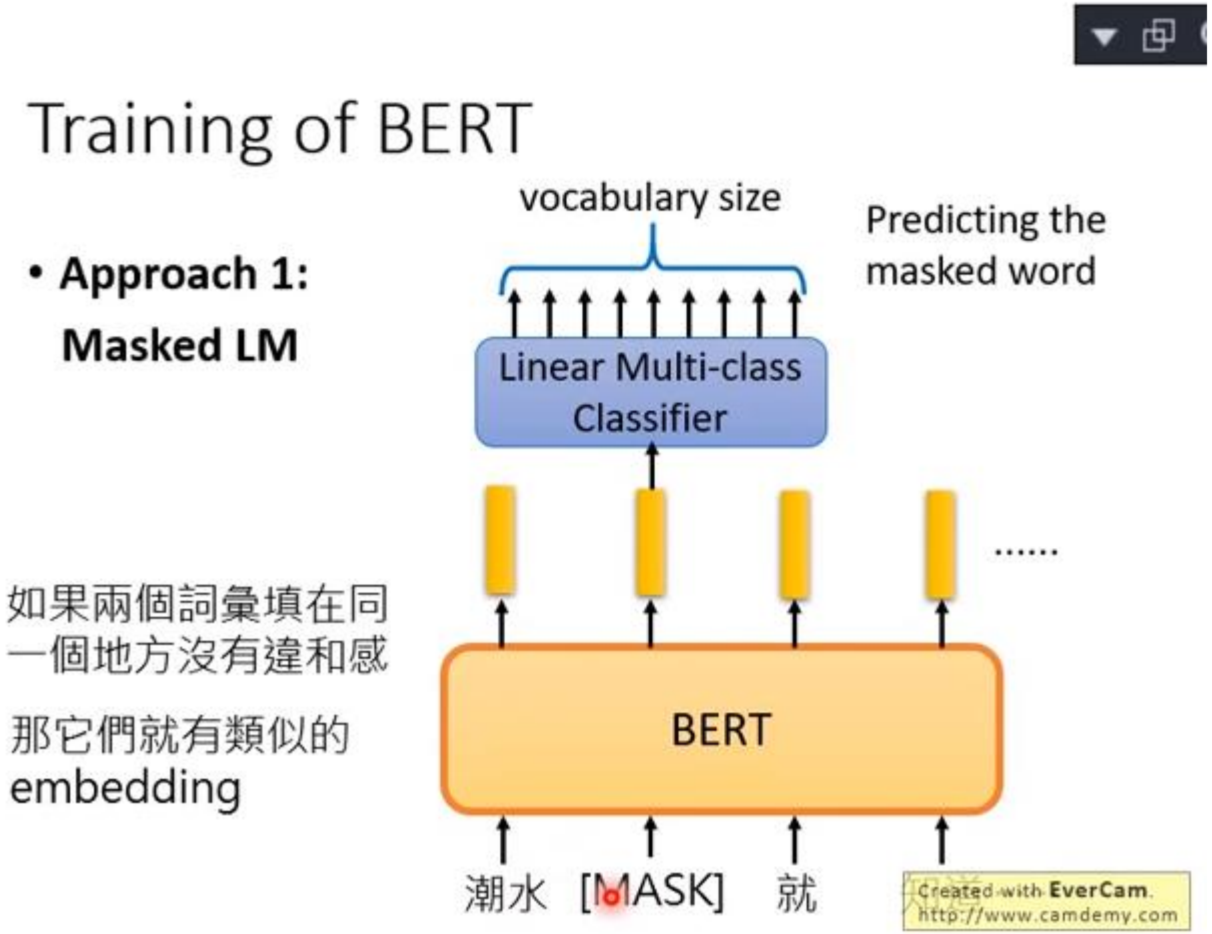
BERT

- **Transformer 的 Encoder**
- **輸出一串Embedding**



Approach 1 – Masked Language Model

- **預測被mask的詞彙**



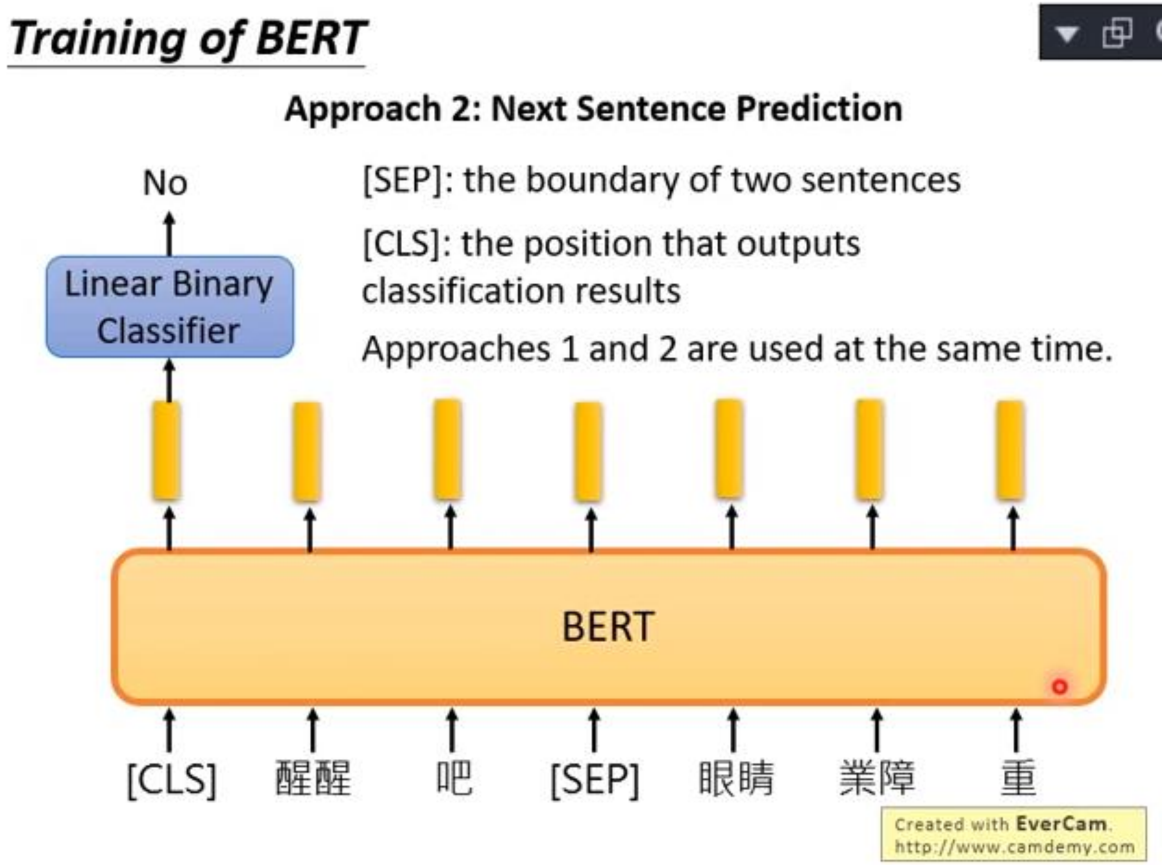
Approach 1 - Masked Language Model

- Estimate $p(x_i \mid x_{[1:i-1]}, x_{[i+1:n]})$ rather than $p(x_i \mid x_{[1:i-1]})$
  - Randomly mask 15% of all tokens and predict token
  - 80% of them - replace token with <mask>
  - 10% of them - replace with <random token>
  - 10% of them - replace with <token>

Alex	is	obnoxious	but	the	tutorial	is	awesome.
Alex	is	obnoxious	but	the	<mask>	is	awesome.
Alex	is	obnoxious	but	the	<banana>	is	awesome.
Alex	is	obnoxious	but	the	<tutorial>	is	awesome.

Approach 2 - Next Sentence Prediction

- **兩種方法同時使用的效果最好**



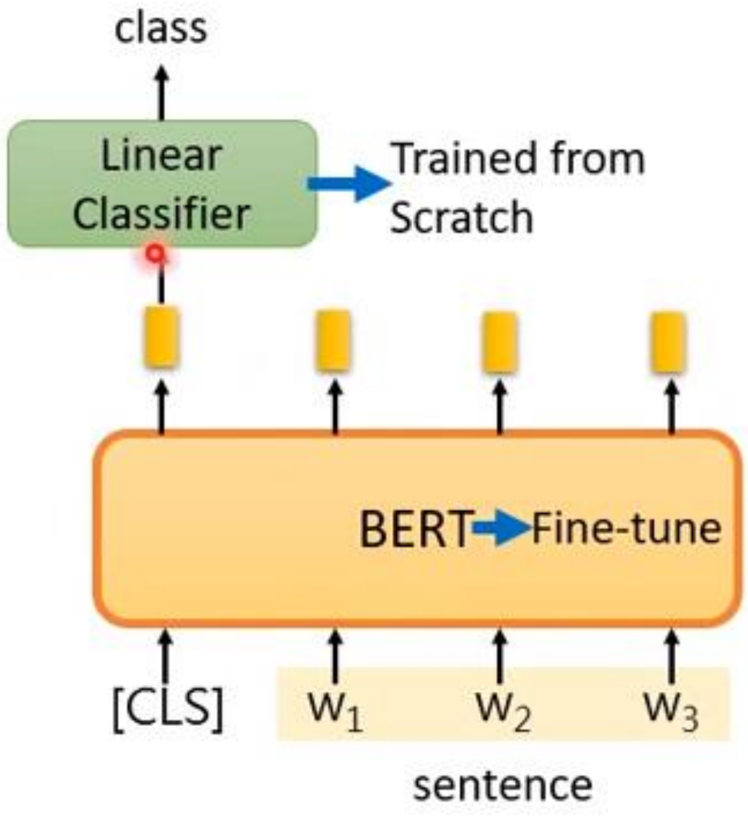
Approach 2 – Next Sentence Prediction

- Predict next sentence
  - 50% of the time, replace it by random sentence
  - Feed the Transformer output into a dense layer to predict if it is a sequential pair.
- **Learn logical coherence**

<BOS> Alex is obnoxious <SEP> I don’t like his shirt
<BOS> Alex is obnoxious <SEP> Look a Martian

How to use BERT

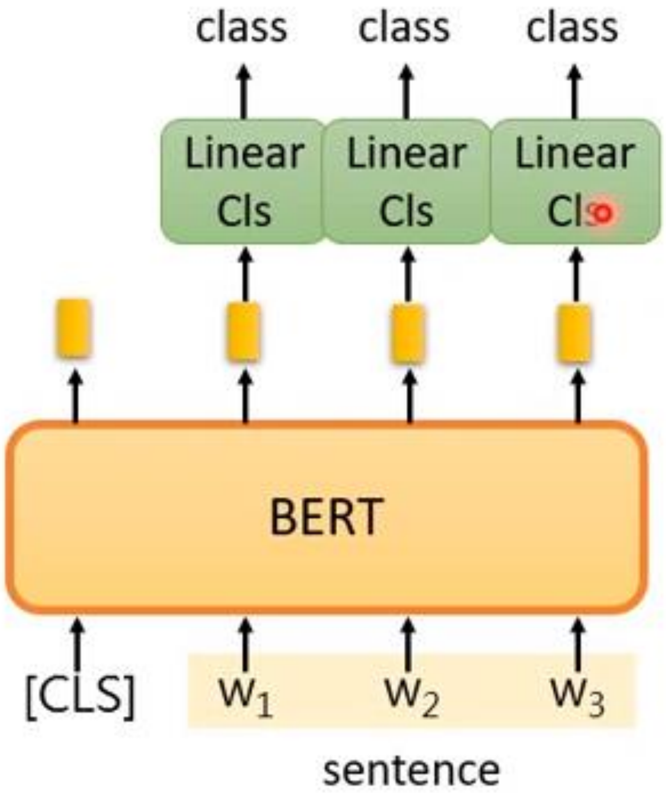
How to use BERT – Case 1



Input: single sentence, output: class

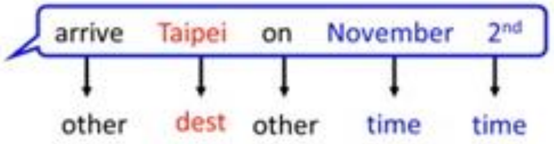
Example:
Sentiment analysis (our HW),
Document Classification

How to use BERT – Case 2

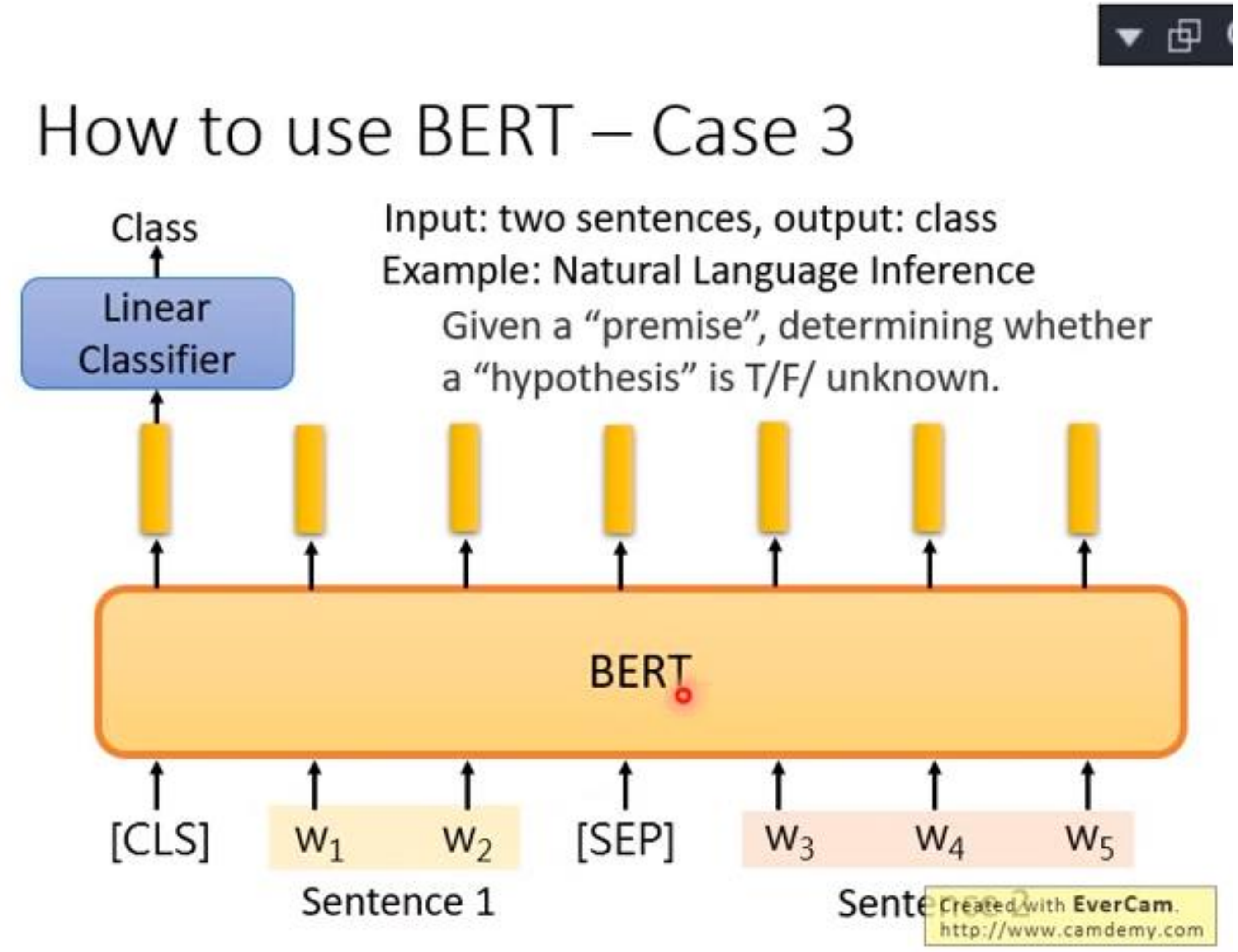


Input: single sentence,
output: class of each word

Example: Slot filling



How to use BERT



How to use BERT

How to use BERT – Case 4

- Extraction-based Question Answering (QA) (E.g. SQuAD)

Document: $D = \{d_1, d_2, \cdots, d_N\}$

Query: $Q = \{q_1, q_2, \cdots, q_M\}$

output: two integers (s, e)

Answer: $A = \{d_s, \cdots, d_e\}$

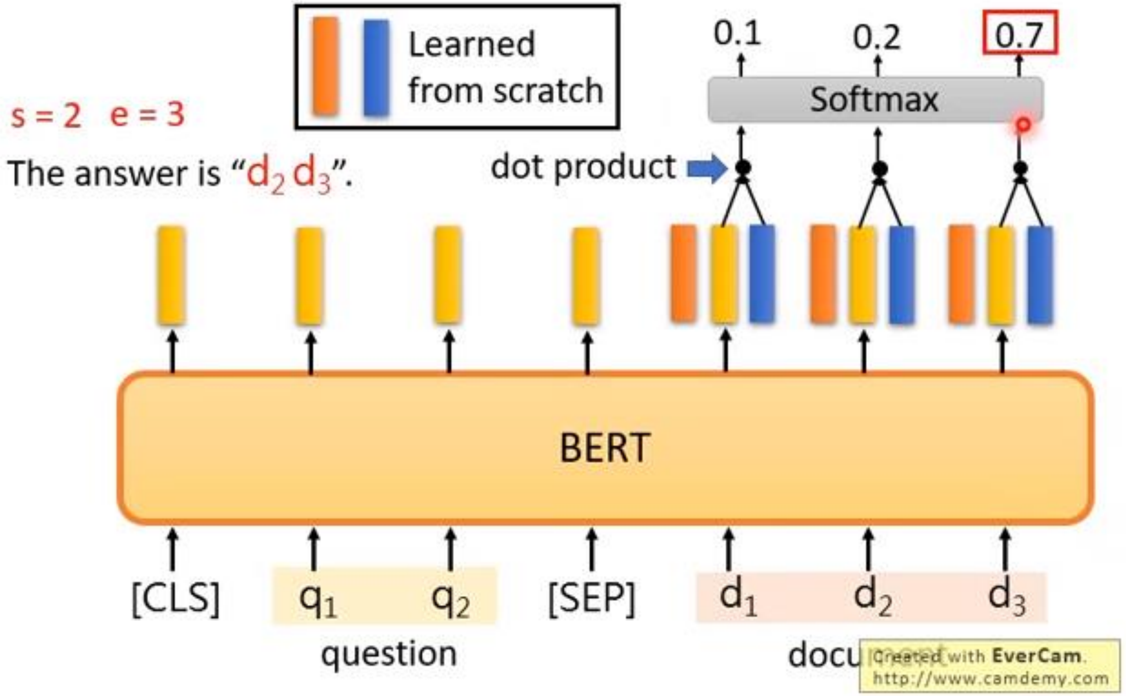
In meteorology, precipitation is any product of the condensation of 17 spheric water vapor that falls under gravity. The main forms of precipitation include drizzle, rain, sleet, snow, graupel and hail… Precipitation forms as smaller droplets coalesce via collision with other rain drops or ice crystals within a cloud. Short, intense periods of rain 77 atte 79 cations are called "showers".

What causes precipitation to fall?
gravity $s = 17, e = 17$

What is another main form of precipitation besides drizzle, rain, snow, sleet and hail?
graupel

Where do water droplets collide with ice crystals to form precipitation?
within a cloud $s =$

How to use BERT – Case 4



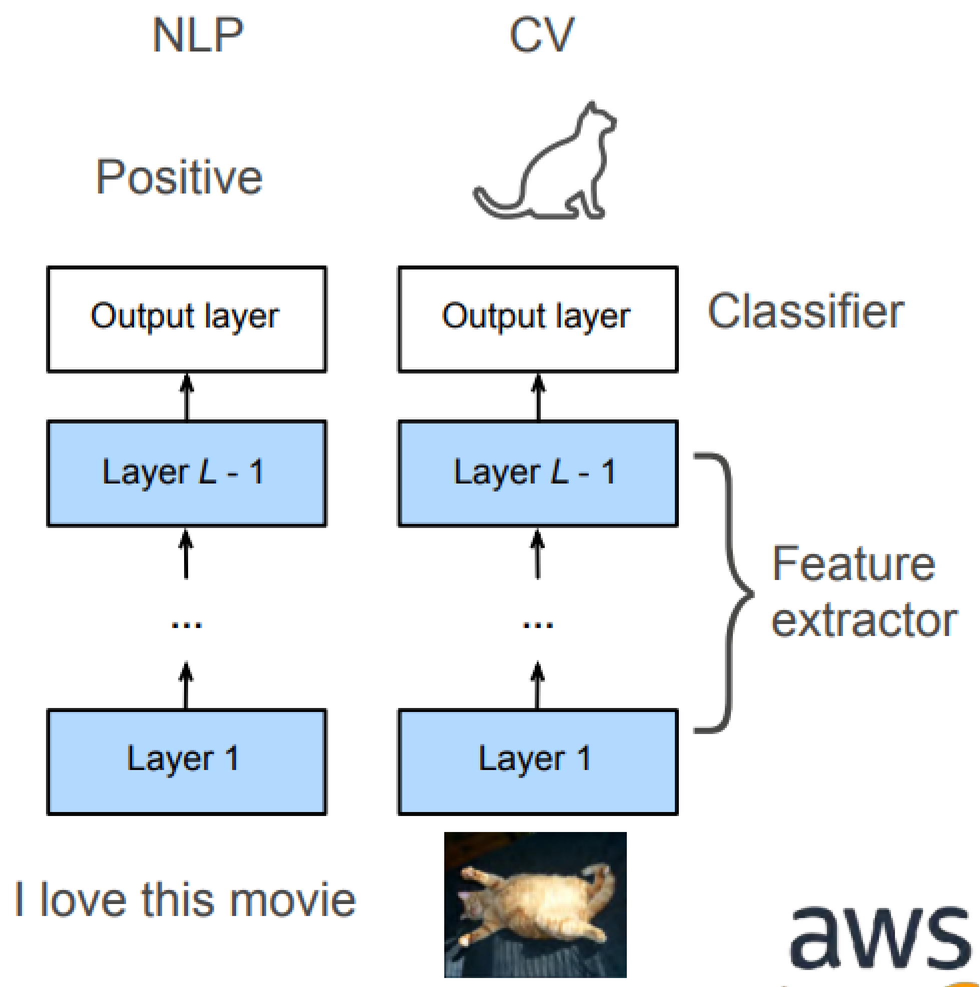
GPT

- **GPT-2的參數量遠大於GPT (至少10倍)**
- **Transformer 的 Decoder**

回顧pdf

Motivation

- Fine-tuning for NLP (learning a prior for NLP)
- Pre-trained model captures prior
- Only add one (or more) output layers for new task



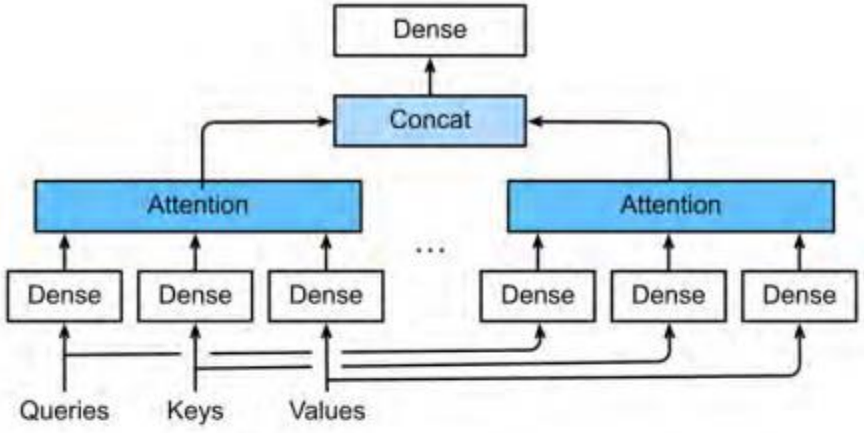
GPT uses Transformer Decoder (Radford et al., '18)

- Pre-train language model, then fine-tune on each task
- **Trained on full length documents**
- 12 blocks, 768 hidden units, 12 heads
- SOTA for 9 NLP tasks

- Language model only looks **forward**
  - `I went to the bank to deposit some money.`
  - `I went to the bank to sit down.`

Architecture

- (Big) transformer encoder
- Train on large corpus (books, wikipedia) with > 3B words

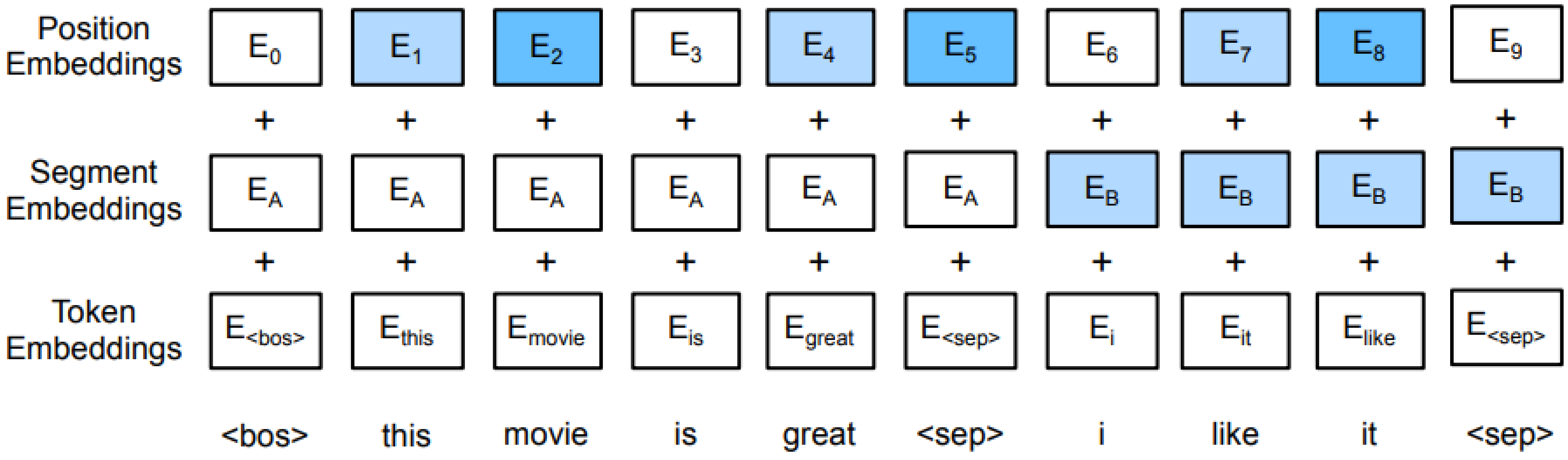


	blocks	hidden units	heads	parameters
small	12	768	12	110M
large	24	1024	16	340M

courses.d2l.ai/berkeley-stat-157/index.html

Input Encoding

- Each example is a pair of sentences
- Add segment embedding and position embedding



GPT2 (it gets even bigger, Radford et al., ‘19)

- Pretrained on 8M webpages (WebText, 40GB)
- Without fine-tuning SOTA on 7 language models

	blocks	hidden units	parameters
small	12	768	110M
large	24	1024	340M
GPT2	48	1600	1.5B

GPT2 Demo (gluon-nlp.mxnet.io)

```
$python sampling_demo.py --model 117M
Please type in the start of the sentence
>>> average human attention span is even shorter than that of a
goldfish
-----  Begin Sample 0 -----
average human attention span is even shorter than that of a
goldfish strutting its way down the jaws. An estimate by the USA
TODAY Science team of 80 human-sized models reveals that a complex
jaw becomes a grandiose mitesaur in 100 million years, less than an
exothermic Holocene huge sea lion, and towering 500 meters tall.

Similar mitesaur-sized jaws would burden as trillions

Scientists would expect a lost at least four million times as much
time in the same distances ocean as other mammals
```

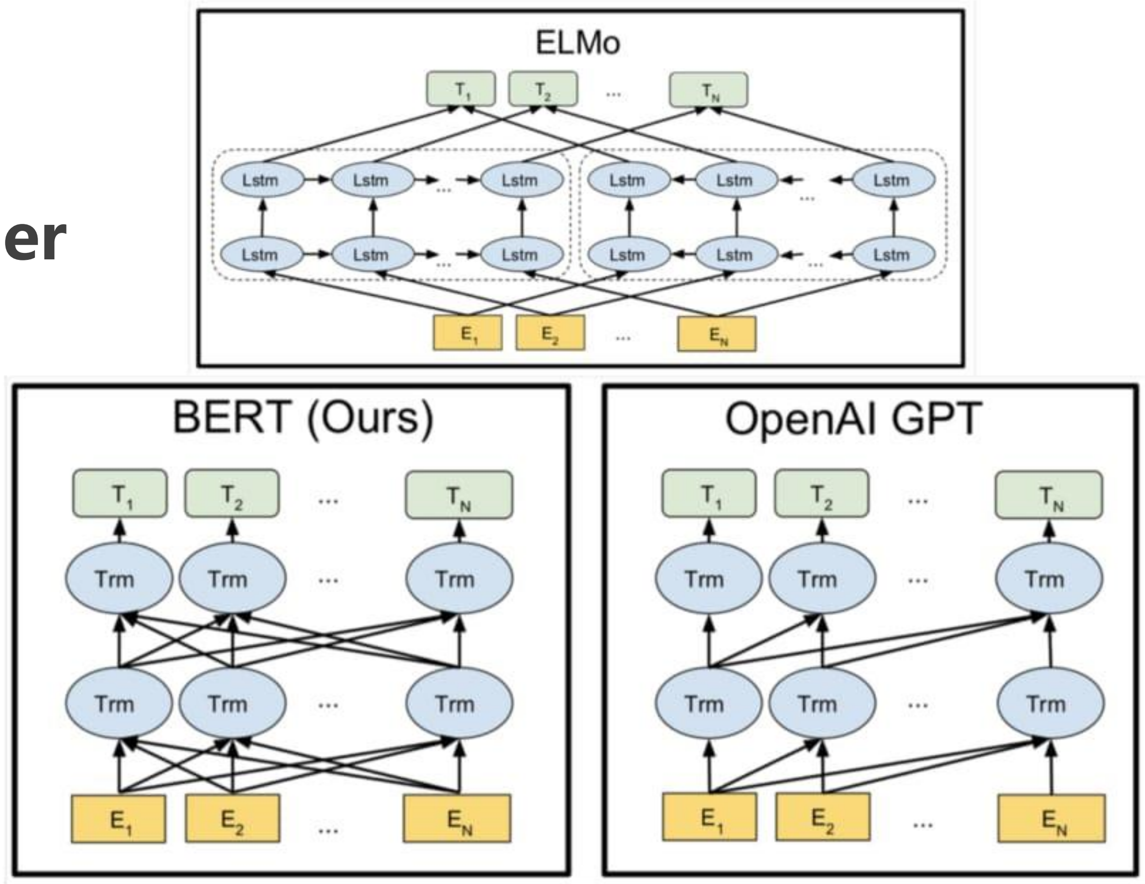
Q & A

- **Transformer, ELMO, GPT, BERT的目的 & 結構?**

三者間的關係

- **ELMO: 動態Embedding**
- **GPT: 簡單使用Transformer的Decoder**
- **BERT: 使用Transformer的Encoder**

  與克漏字來訓練



謝謝聆聽

Thank you