



# Computer Vision

CS308

Feng Zheng

SUSTech CS Vision Intelligence and Perception



南方科技大学

SOUTHERN UNIVERSITY OF SCIENCE AND TECHNOLOGY



# Content

- Introduction to Vision-Language Learning
- Overview of Vision-Language Research
- Sequence-to-sequence Models
  - RNNs
  - Transformers



# Vision-Language Learning



# Vision-Language Learning

- Vision and language are two fundamental capabilities of human intelligence. **Humans** routinely perform tasks through the interactions between vision and language.

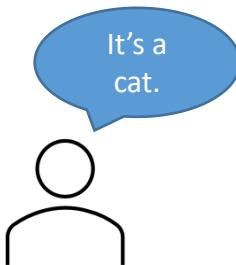
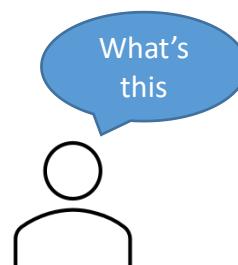


hand stupid actual younger round imagin...  
isid hard qual cold vali expect review bore scene check atmospher...  
fact picc pretti comfort headset suck leav hear break hope  
and movi piec watch tell charact exper... amaz beauti mont  
old mess hate wast cool tast act experi... drop play total feel volu...  
price film owner item funny charg... dish mi  
ian server enjoi terrib... return avoid lighte...  
anag fresh sound monel tabl star pair impress start staff fail please...  
dai flavor minut worth charg us littl... saladine easi... simpli ra...  
buffet talk awesom cast batterifood redict probabl... people direct receiv... fast overal... perform awshort...  
recept bought time servic product stai hour horribl... deal  
are fine delici... soon life like seen tri... seen wait fit friendli fant...  
week absolut poorlef look actor extrem... reason night come famili... happi menu scrip... cost su...  
logu ni plug love recommend e devic... lack definit wear pizza call stor... perfect cell black crap friend... cost su...

Learning



Reasoning

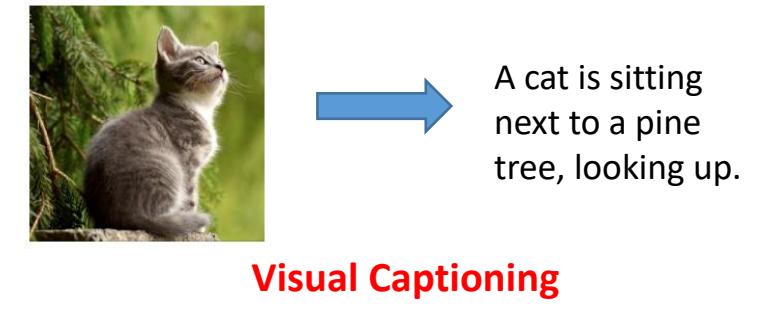
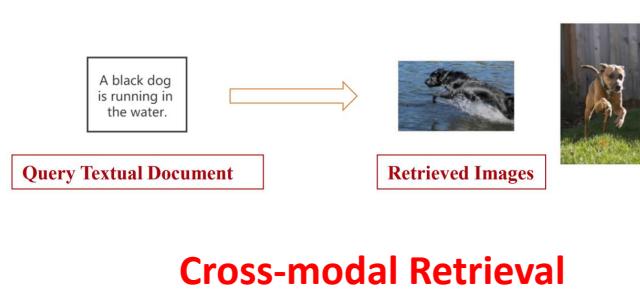
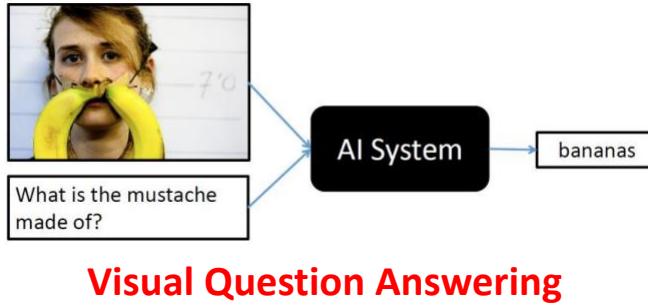


Mei, Tao, Wei Zhang, and Ting Yao. "Vision and language: from visual perception to content creation."  
Miech, Antoine, et al. "Howto100m: Learning a text-video embedding by watching hundred million narrated video clips."  
Cat image is retrieved from [here](#)



# Vision-Language Learning

- Vision-language learning aims to build models that can process and relate information from vision and language.
- V+L research is about how to train a smart AI system that **can see and talk**.



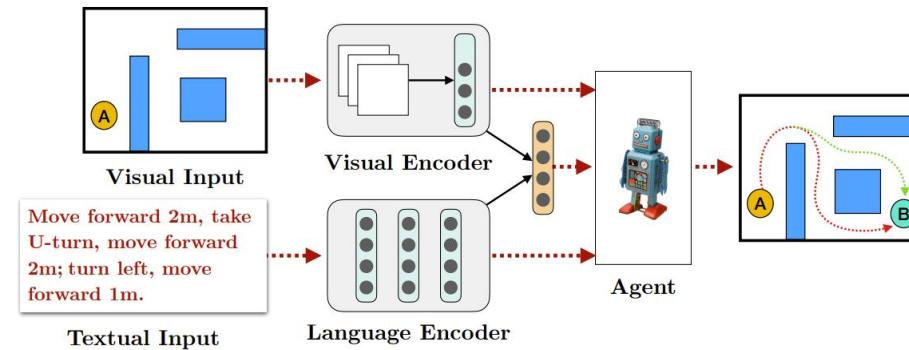


# Vision-Language Learning

- Vision-language learning aims to build models that can process and relate information from vision and language.
- V+L research is about how to train a smart AI system that can see and talk.



Text-to-image Synthesis



Vision-Language Navigation

Midjourney: <https://www.midjourney.com/home/>  
Stable Diffusion 2.0. <https://stability.ai/blog/stable-diffusion-v2-release>

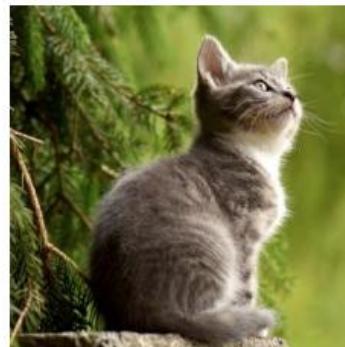


# Overview of Vision-Language Research



# Visual Captioning

- Generating natural language description of an image/video.
- Applications:
  - Alt-text generation (from PowerPoint).
  - Generating summaries for Videos (YouTube)
  - Content-based image retrieval.
  - .....



A cat is sitting next to a pine tree, looking up.

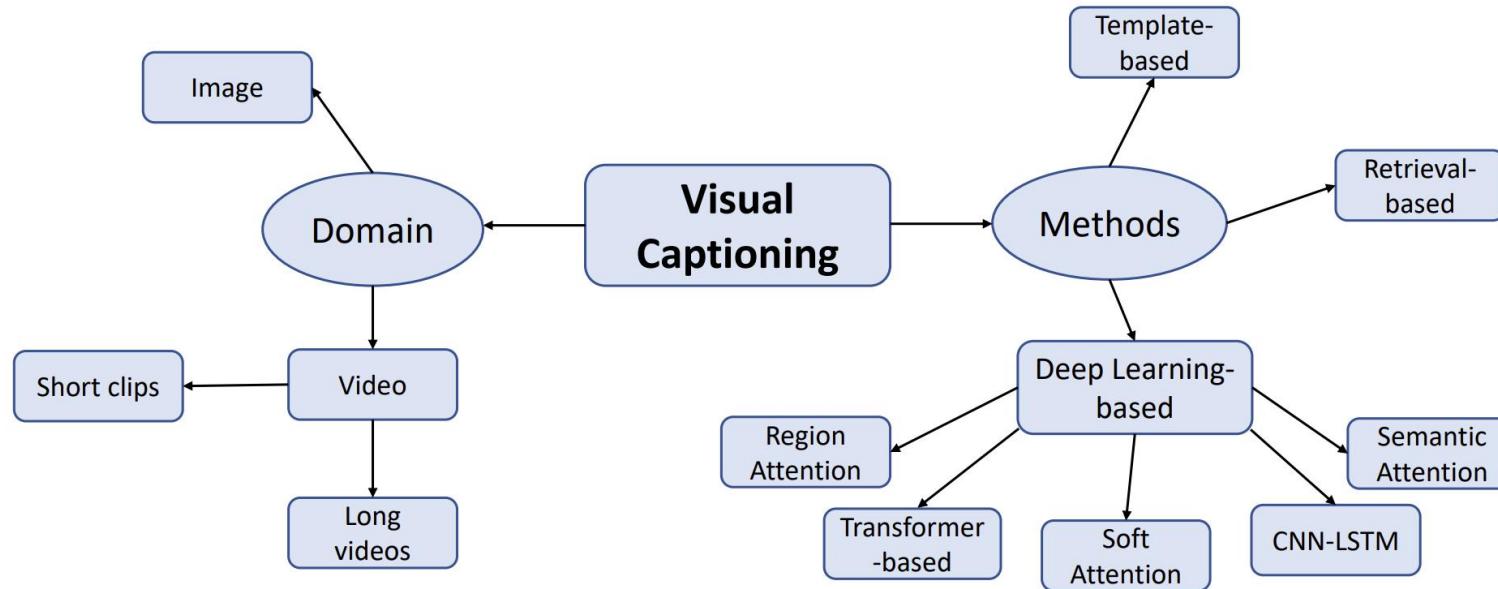


# Visual Captioning

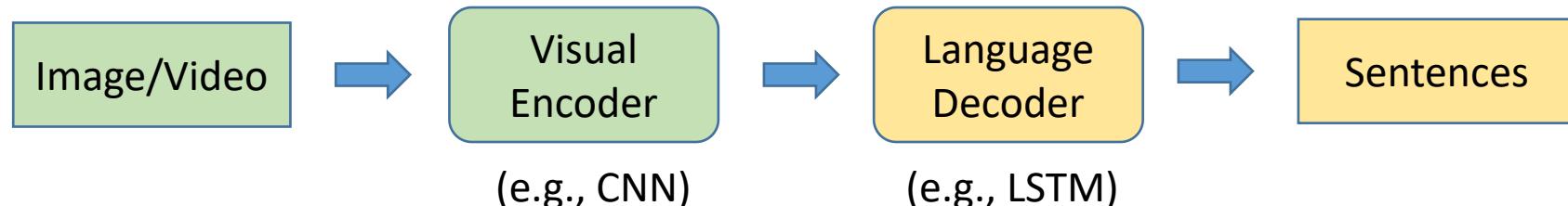
- Datasets:
  - **MS COCO**: 12,0000 images with 5 free form sentences per image.
  - **Flickr30K**: 30k images collected from Flickr with 158k captions provided by human annotators.
  - **MSVD**: 1970 short videos collected from YouTube. Each video has about 40 English human annotations.
  - .....
- How to evaluate ?
  - There is no “correct” answer for describing an image by a sentence.
  - A predicted sentence is evaluated against a set of human annotated sentences.



# Visual Captioning Taxonomy



- Commonly used framework: Encoder-Decoder





# Cross-modal Retrieval

- Support similarity search for multi-modal data, e.g., the retrieval of images in response to a query textual document or vice versa

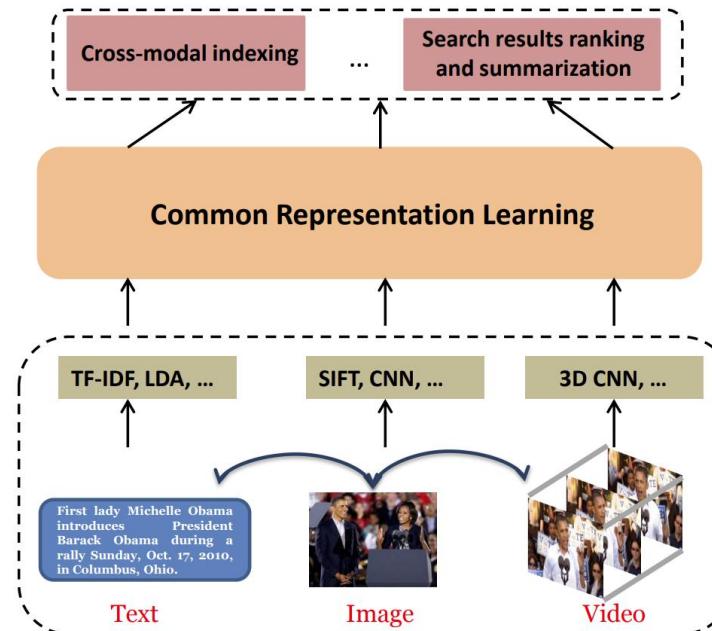
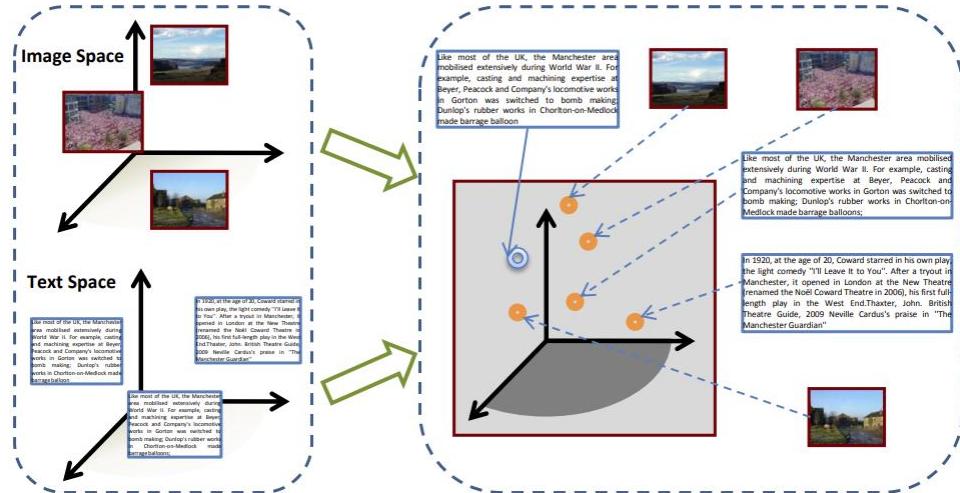


Fig. 3. The general framework of cross-modal retrieval.



# Cross-modal Retrieval

- Datasets

- Wiki, INRIA-Websearch, Flickr30K, Pascal-VOC, ...



**Geography and Places**

The population of Turkey stood at 71.5 million with a growth rate of 1.31% per annum, based on the 2008 Census. It has an average population density of 92 persons per km<sup>2</sup>. The proportion of the population residing in urban areas is 70.5%. People within the 15–64 age group constitute 66.9% of the total population, the 0–14 age group corresponds 26.4% of the population, while 65 years and higher of age correspond to 7.1% of the total population. Life expectancy stands at 70.67 years for men and 75.73 years for women, with an overall average of 73.14 years for the populace as a whole. Education is compulsory and free from ages 6 to 15. The literacy rate is 95.3% for men and 79.6% for women, with an overall average of 87.4%. The low figures for women are mainly due to the traditional customs of the Arabs and Kurds who live in the southeastern provinces of the country. Article 66 of the Turkish Constitution defines a "Turk" as "anyone who is bound to the Turkish state through the bond of citizenship"; therefore, the legal use of the term "Turkish" as a citizen of Turkey is different from the ethnic definition. (...)



**Warfare**

A number of variants were built on the same chassis as the TAM tank. The original program called for the design of an infantry fighting vehicle, and in 1977 the program finished manufacturing the prototype of the "Vehículo de Combate Transporte de Personal" (Personnel Transport Combat Vehicle), or VCTP. The VCTP is able to transport a squad of 12 men, including the squad leader and nine riflemen. The squad leader is situated in the turret of the vehicle; one rifleman sits behind him and another six are seated in the chassis; the eighth manning the hull machine gun and the ninth situated in the turret with the gunner. All personnel can fire their weapons from inside the vehicle, and the VCTP's turret is armed with Rheinmetall's Rh-202 20 millimeter (.79 in) autocannon. The VCTP holds 880 rounds for the autocannon, including subcaliber armor-piercing DM63 rounds. It is also armed with a 7.62 millimeter FN MAG 60-20 mounted on the turret roof. Infantry can dismount through a door on the rear of the hull. (...)

Fig. 10. Two examples on the Wiki dataset. The text is an article describing the content of an image.

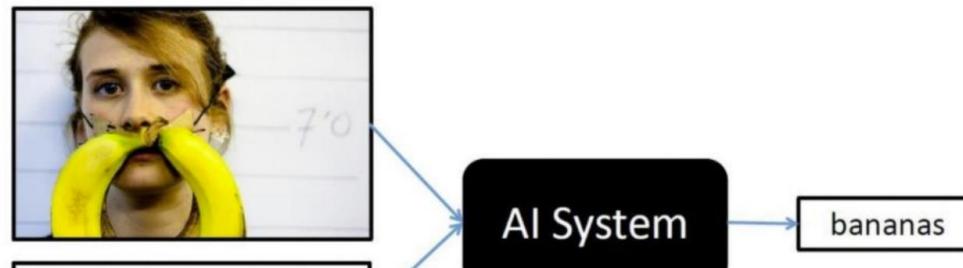


Fig. 11. Examples on the Pascal VOC dataset. The tags describing the content of an image are regarded as the text modality.



# Visual Question Answering

- Given an image and a question (text) about the image/video, the model aims to provide an accurate natural language answer



recognition

detection

classification

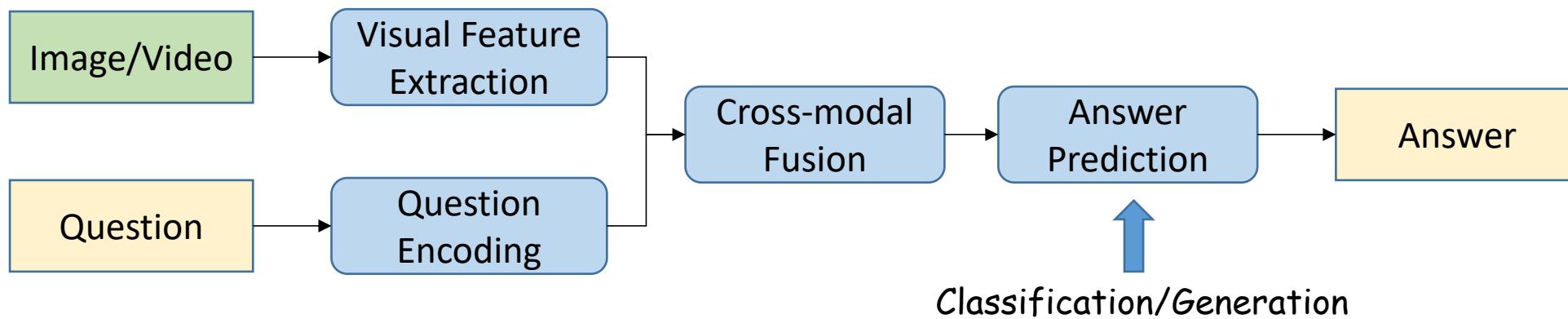
Commonsense  
reasoning

Relationship  
mining



# Visual Question Answering

- A typical framework





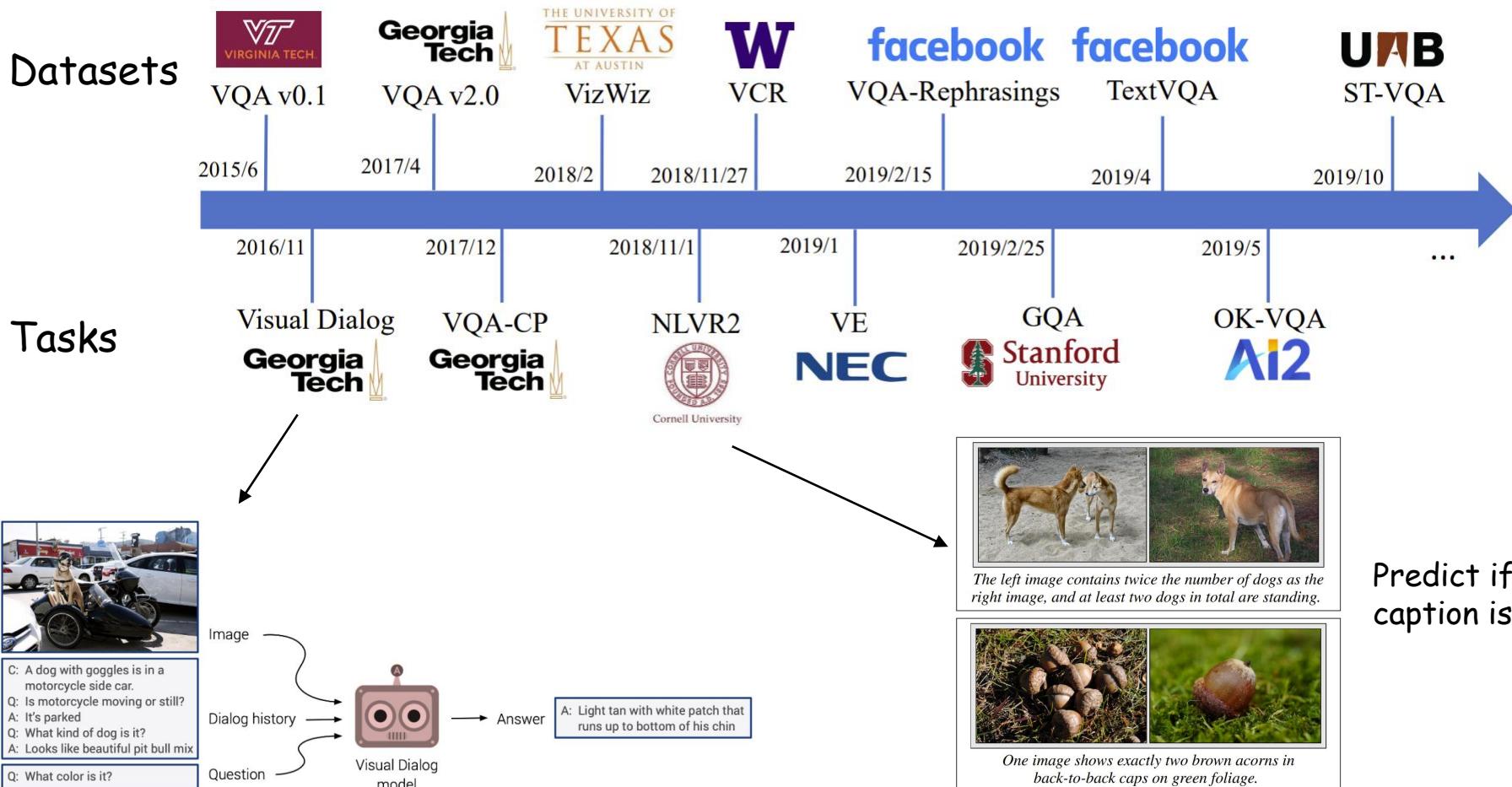
# Visual Question Answering

- Datasets and Evaluation

Dataset	Source of images	Number of images	Number of questions	Num. questions / num. images	Num. question categories	Question collection	Average quest. length	Average ans. length	Evaluation metrics
DAQUAR [51]	NYU-Depth V2	1,449	12,468	8.6	4	Human	11.5	1.2	Acc. & WUPS
COCO-QA [63]	COCO	117,684	117,684	1.0	4	Automatic	8.6	1.0	Acc. & WUPS
FM-IQA [22]	COCO	120,360	-	-	-	Human	-	-	Human
VQA-real [3]	COCO	204,721	614,163	3.0	20+	Human	6.2	1.1	Acc. against 10 humans
Visual Genome [41]	COCO	108,000	1,445,322	13.4	7	Human	5.7	1.8	Acc.
Visual7W [100]	COCO	47,300	327,939	6.9	7	Human	6.9	1.1	Acc.
Visual Madlibs [95]	COCO	10,738	360,001	33.5	12	Human	6.9	2.0	Acc.
VQA-abstract [3]	Clipart	50,000	150,000	3.0	20+	Human	6.2	1.1	Acc.
VQA-balanced [98]	Clipart	15,623	33,379	2.1	1	Human	6.2	1.0	Acc.
KB-VQA [78]	COCO	700	2,402	3.4	23	Human	6.8	2.0	Human
FVQA [80]	COCO & ImageNet	1,906	4,608	2.5	12	Human	9.7	1.2	Acc.

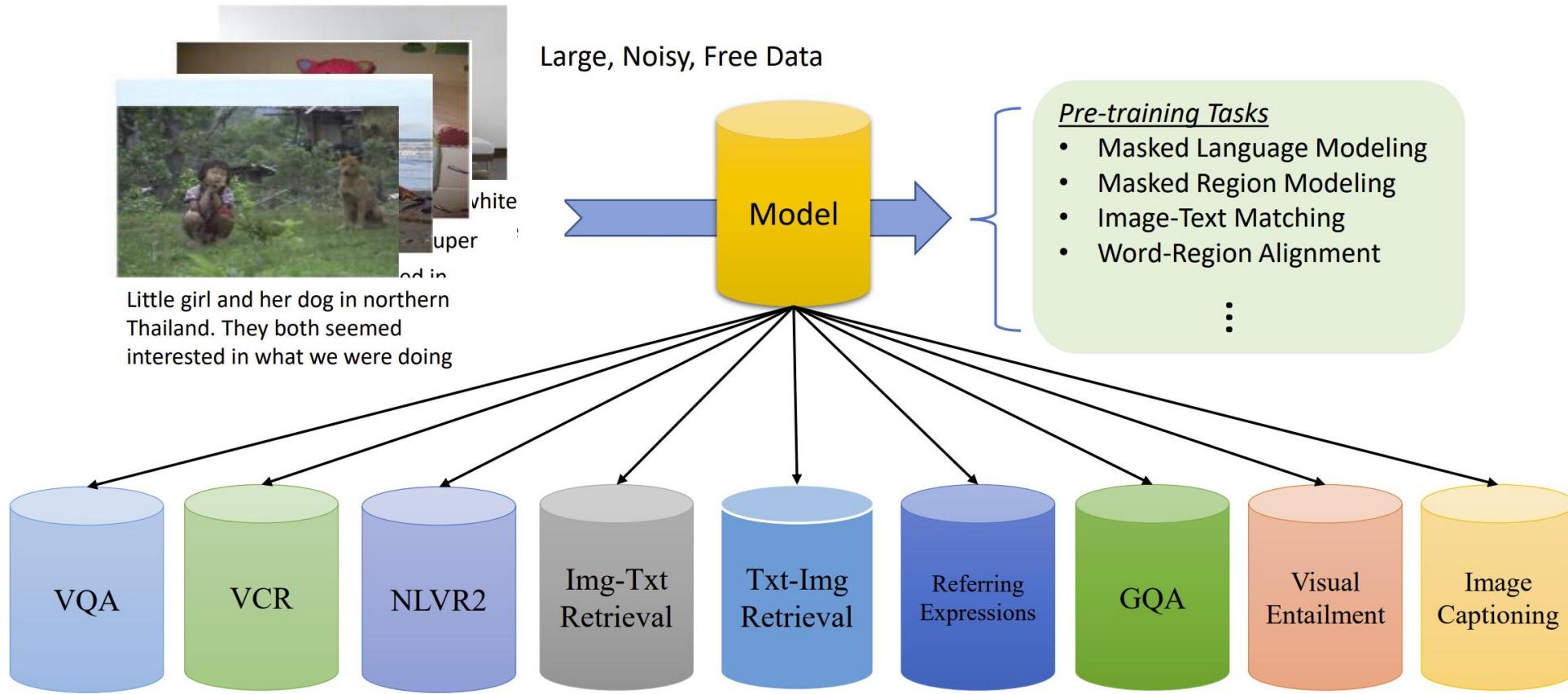


# Other VL Understanding tasks



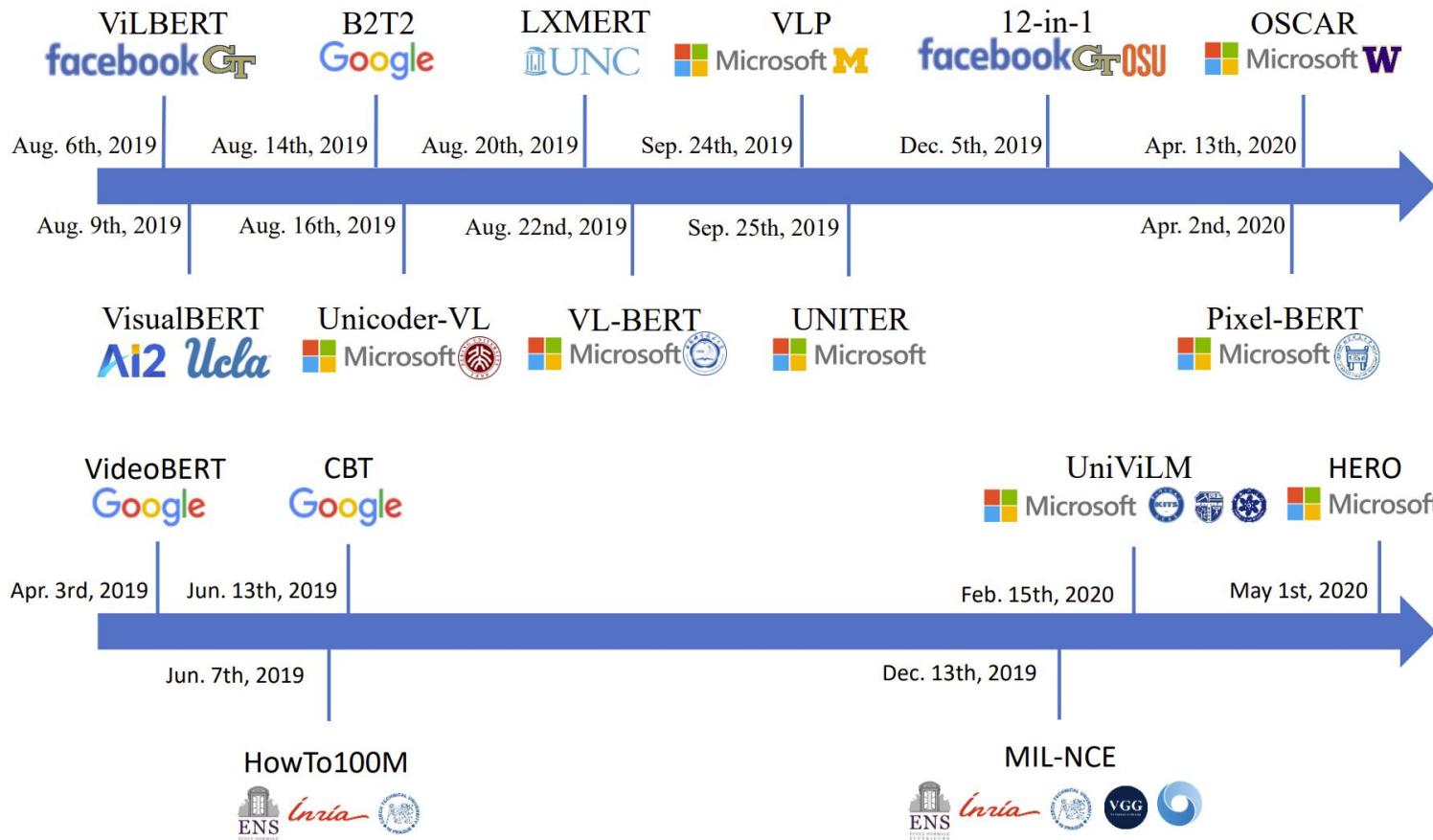


# Self-supervised Learning for VL





# Self-supervised Learning for VL



## Image Downstream Tasks

- VQA
- VCR
- NLVR2
- Visual Entailment
- Referring Expressions
- Image-Text Retrieval
- Image Captioning

## Video Downstream Tasks

- Video QA
- Video-and-Language Inference
- Video Captioning
- Video Moment Retrieval



# Sequence-to-Sequence Modeling (Seq2Seq)



# Sequence Data

- Text, image and video are all sequence data

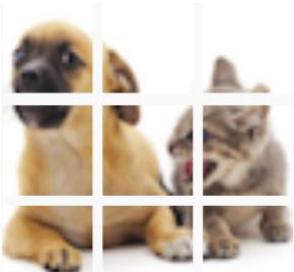


Image patches



How to model  
sequence data?



Video Frames

A cat is sitting next to a pine tree. → "A", "cat", "is", "sitting", "next", "to", "a", "pine", "tree", ":"

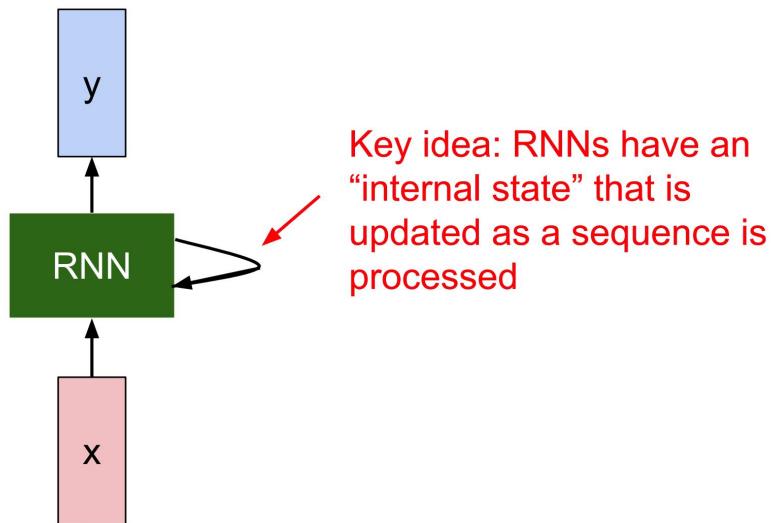
Natural Language



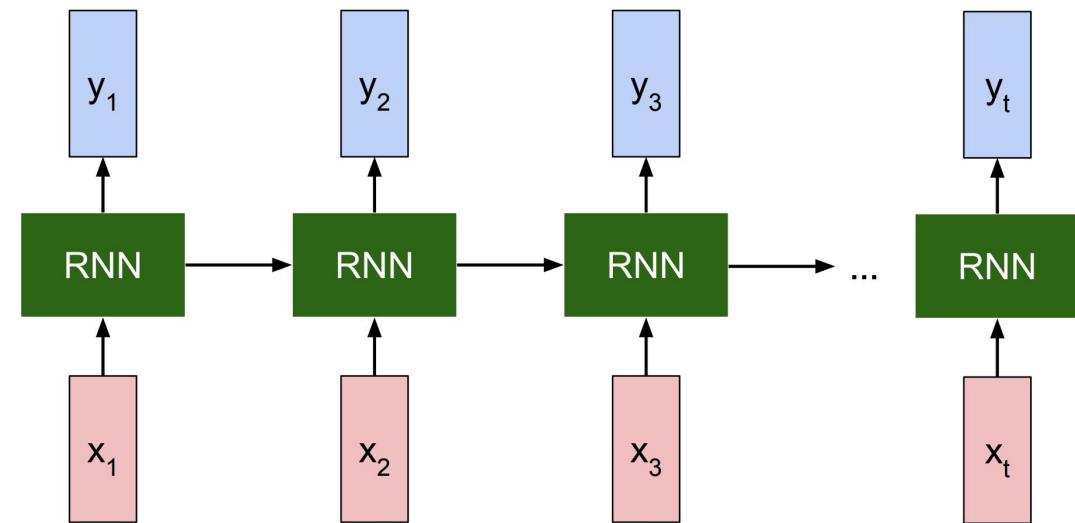
# RNN for Sequence Modeling

**Recurrent Neural Networks (RNN): connections between nodes form a directed or undirected graph along a **temporal** sequence**

Rolled RNN



Unrolled RNN





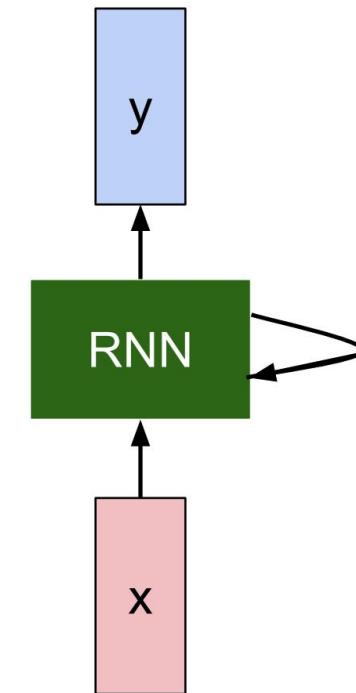
# Recurrent Neural Network

## RNN Hidden State Update

We can process a sequence of vectors  $\mathbf{x}$  by applying a **recurrence formula** at every time step:

$$h_t = f_W(h_{t-1}, x_t)$$

new state      /      old state      input vector at some time step  
some function with parameters W





# Recurrent Neural Network

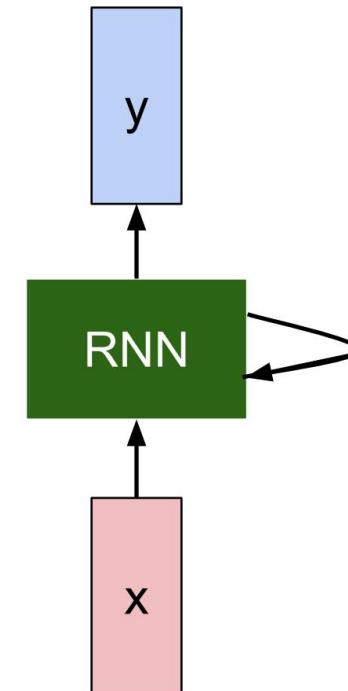
## RNN Output Generation

We can process a sequence of vectors  $\mathbf{x}$  by applying a **recurrence formula** at every time step:

$$y_t = f_{W_{hy}}(h_t)$$

output                          new state

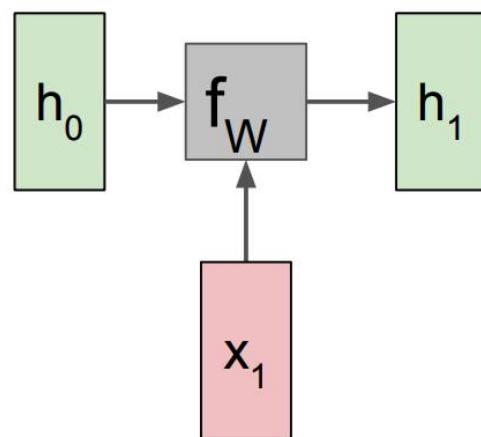
another function  
with parameters  $W_o$





# Computational Graph

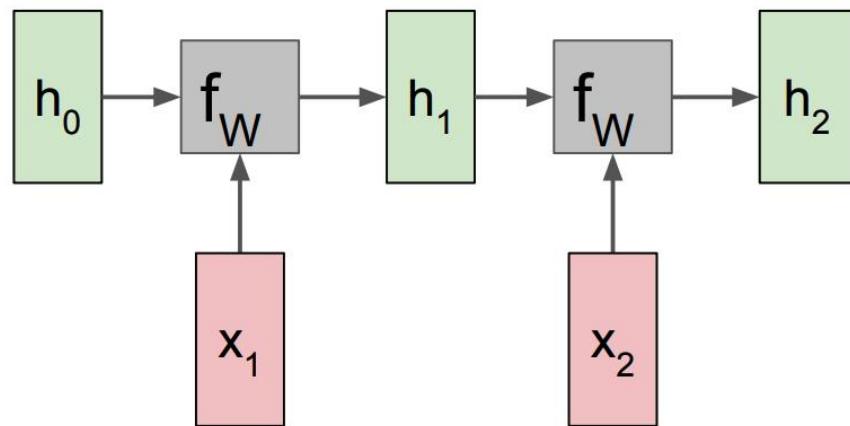
## RNN Computational Graph





# Computational Graph

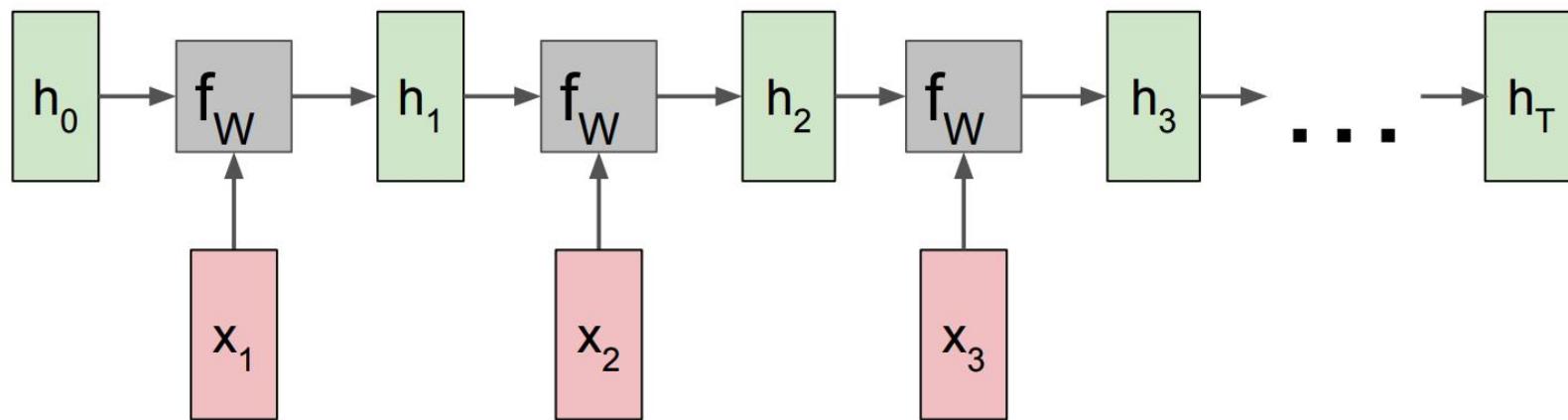
## RNN Computational Graph





# Computational Graph

## RNN Computational Graph

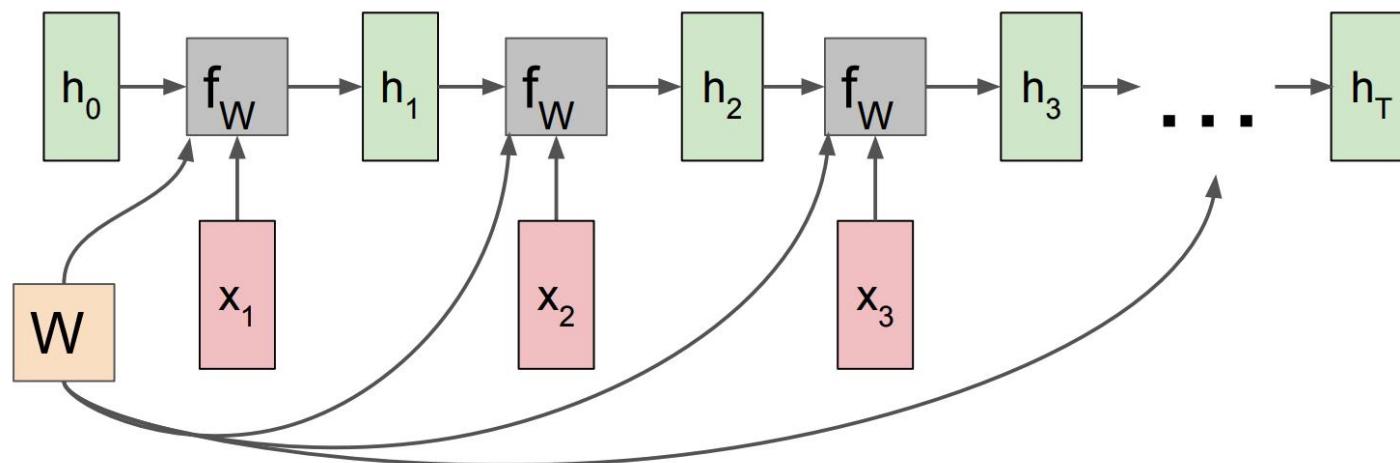




# Computational Graph

## RNN Computational Graph

Re-use the same weight matrix at every time-step

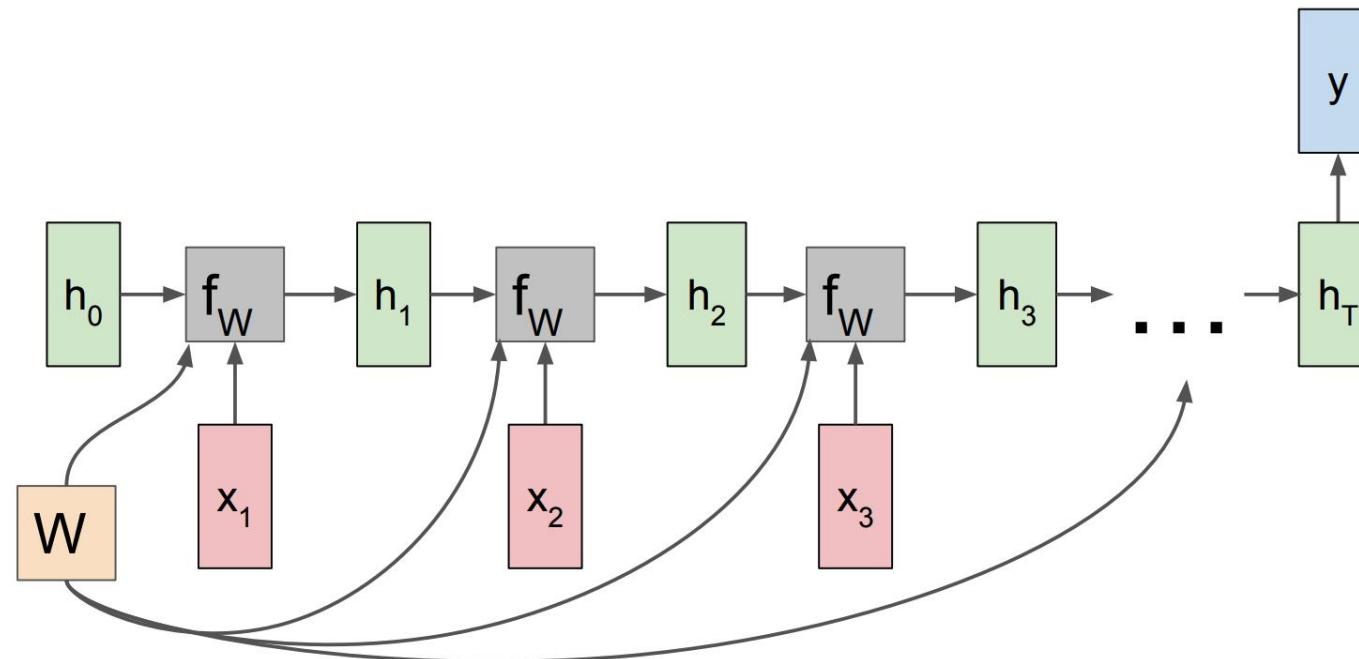




# Computational Graph

Many input to one output:

- Text **classification**
- Stock **forecasting**
- ...

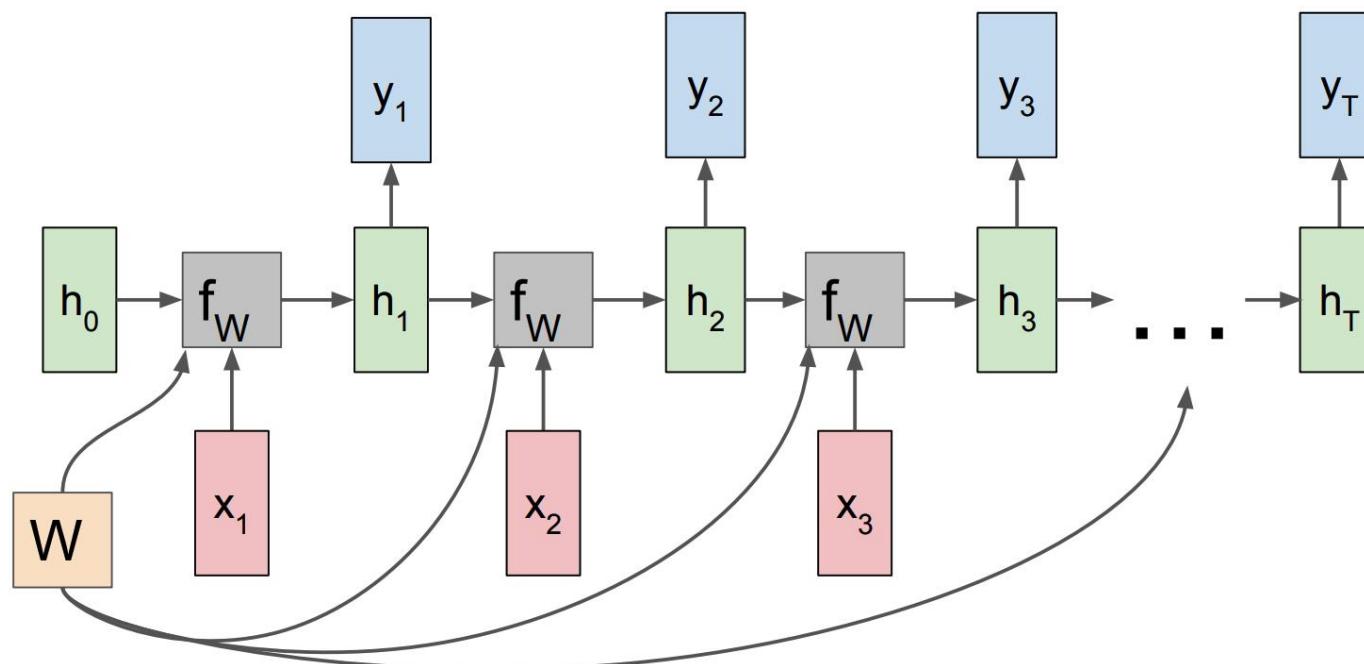




# Computational Graph

Many input to many output:

- Language translation

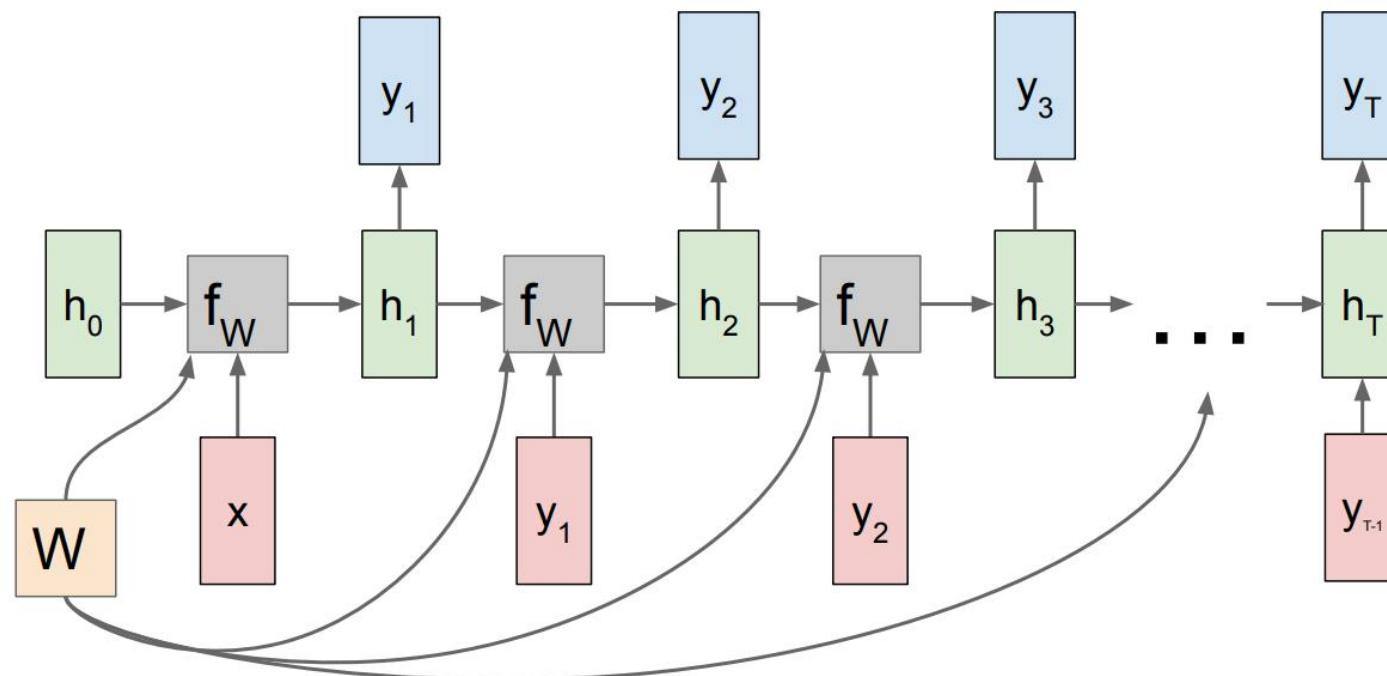




# Computational Graph

One input to many output:

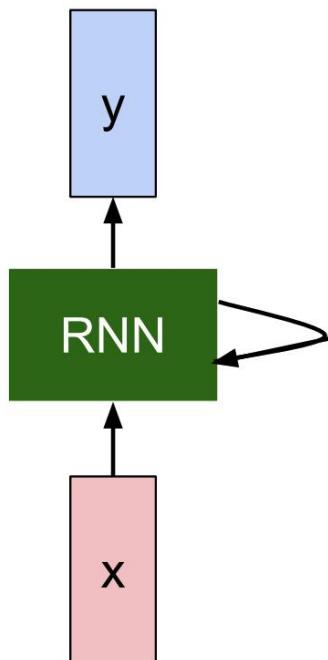
- Image caption





# RNN Summary

The state consists of a single “*hidden*” vector  $\mathbf{h}$ :



$$h_t = f_W(h_{t-1}, x_t)$$



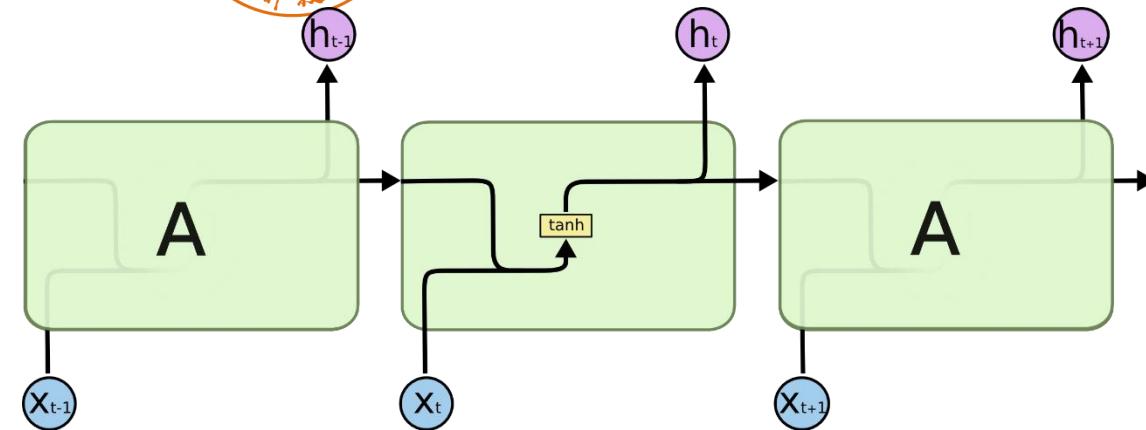
$$h_t = \tanh(W_{hh}h_{t-1} + W_{xh}x_t)$$

$$y_t = W_{hy}h_t$$

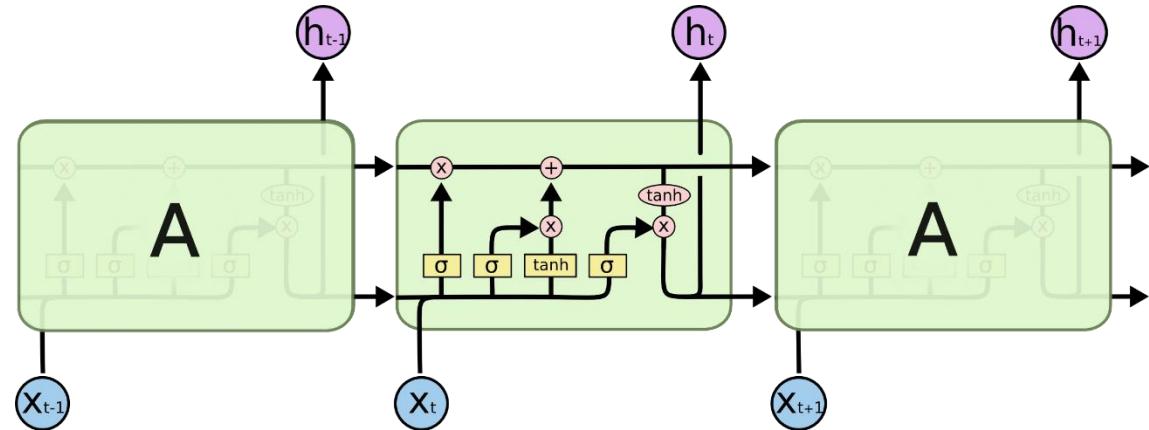
Sometimes called a “Vanilla RNN” or an “Elman RNN” after Prof. Jeffrey Elman



# Long Short-Term Memory (LSTM)



Repeating Module in a standard RNN



Repeating Module in a LSTM

Update hidden state:

$$h_t = \tanh(W_{hh}h_{t-1} + W_{xh}x_t)$$

Forget **gate**:  $f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f)$

**Input gate**:  $i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i)$

**Output gate**:  $o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o)$

**Cell memory**:  $\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C)$

**Update cell memory**:  $C_t = f_t * C_{t-1} + i_t * \tilde{C}_t$

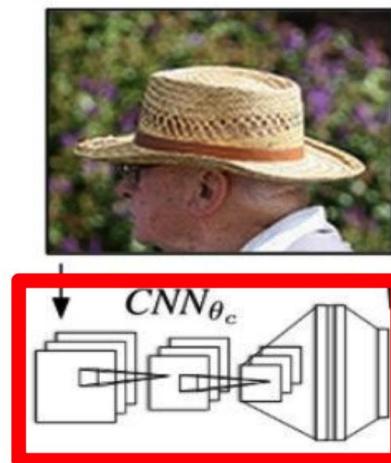
**Get the hidden state**:  $h_t = o_t * \tanh(C_t)$



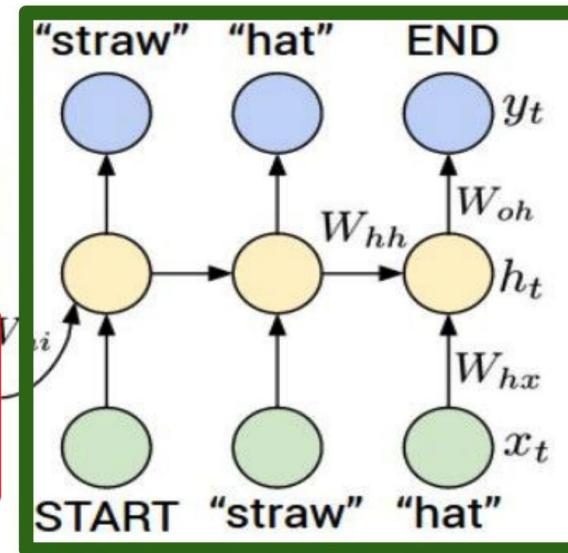
# Encoder-Decoder for Image Captioning

## Encoder-Decoder

- **CNN Encoder** for summarizing the visual data into a vector
- **RNN Decoder** for generating a word sequence.



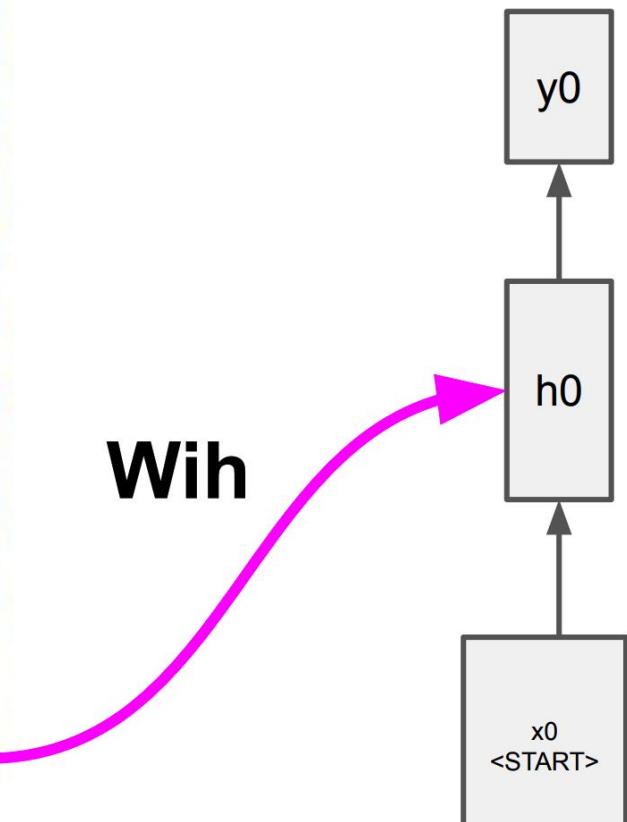
## Recurrent Neural Network



## Convolutional Neural Network



test image



**before:**

$$h = \tanh(W_{xh} * x + W_{hh} * h)$$

**now:**

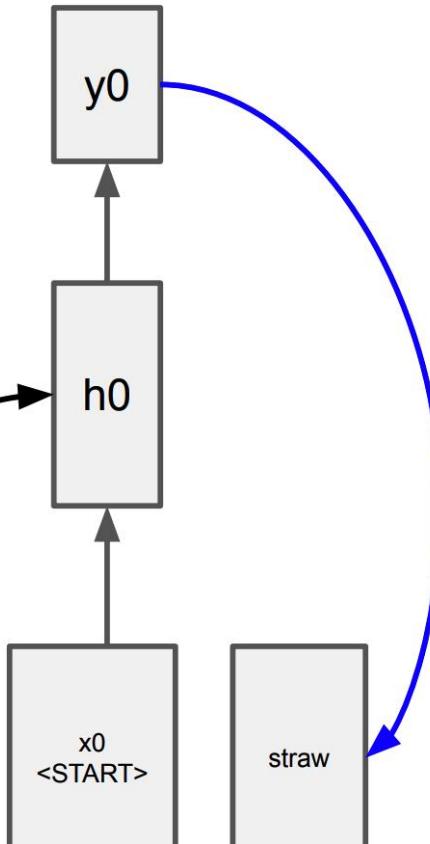
$$h = \tanh(W_{xh} * x + W_{hh} * h + W_{ih} * v)$$



test image



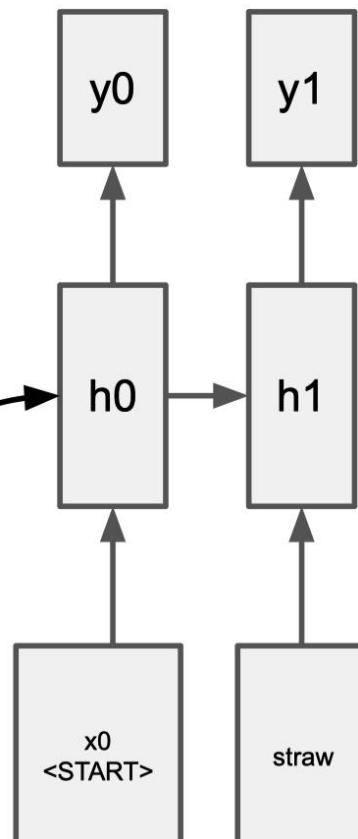
test image



sample!

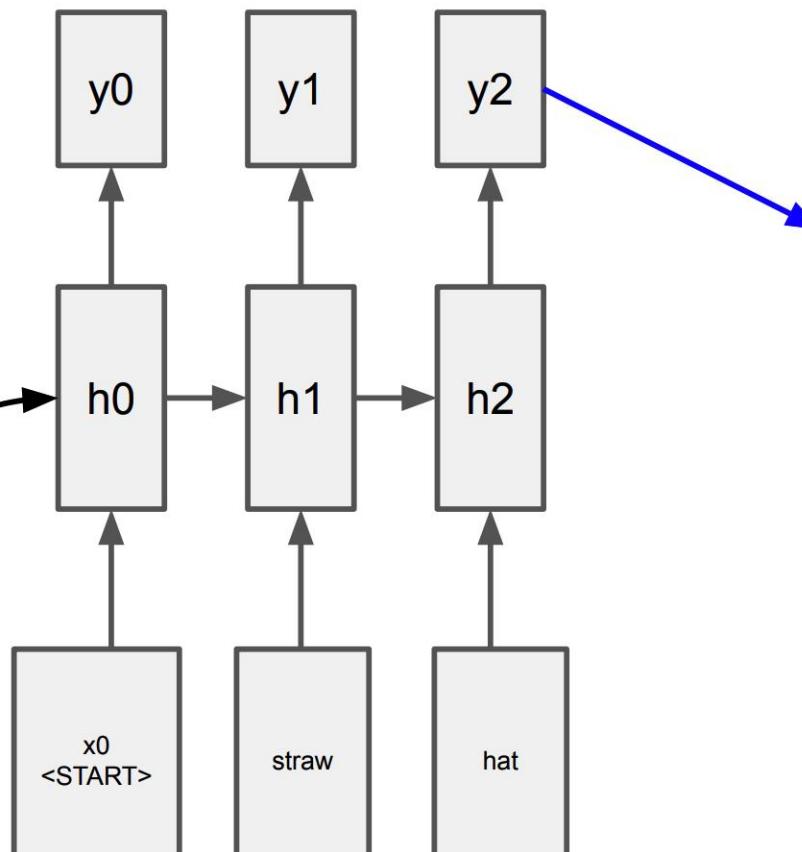


test image





test image



sample  
<END> token  
=> finish.



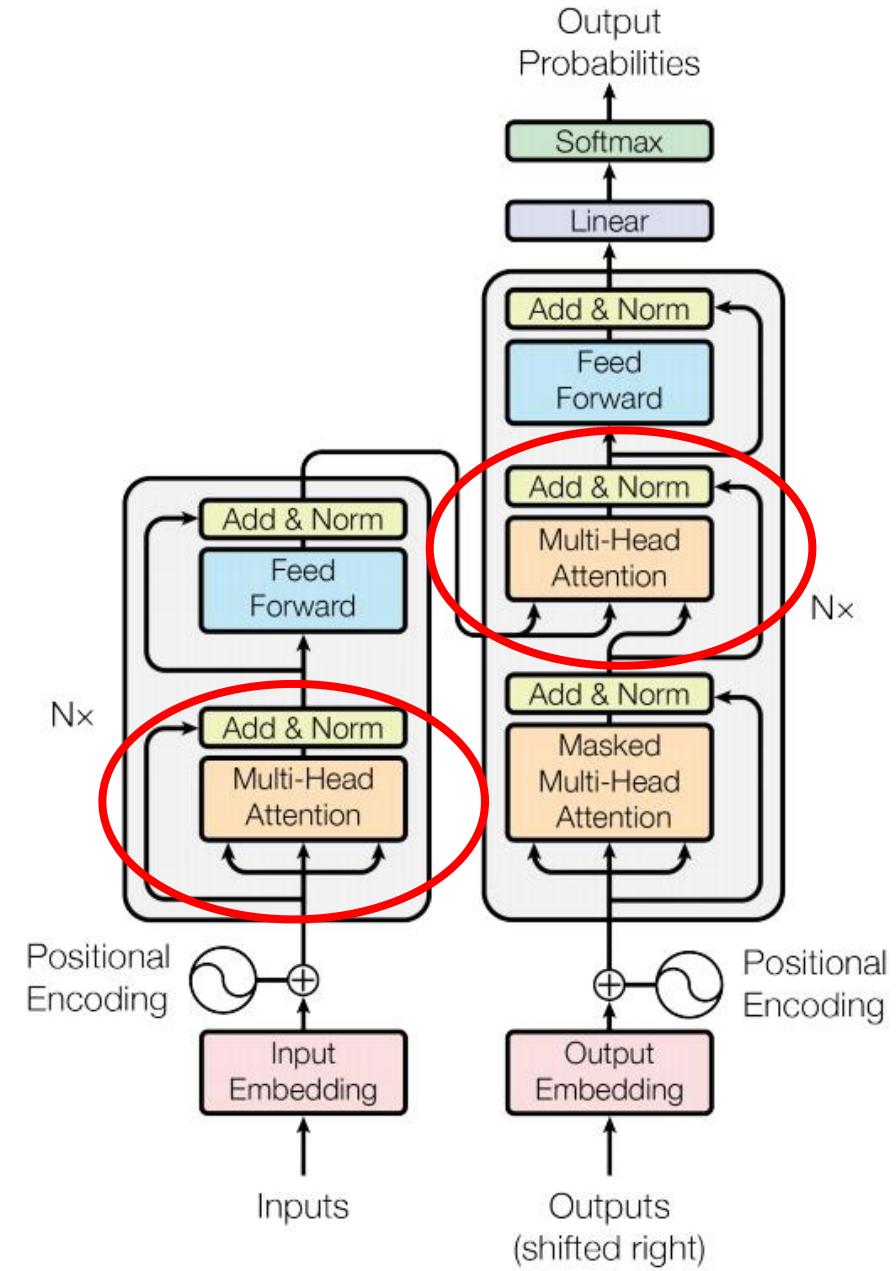
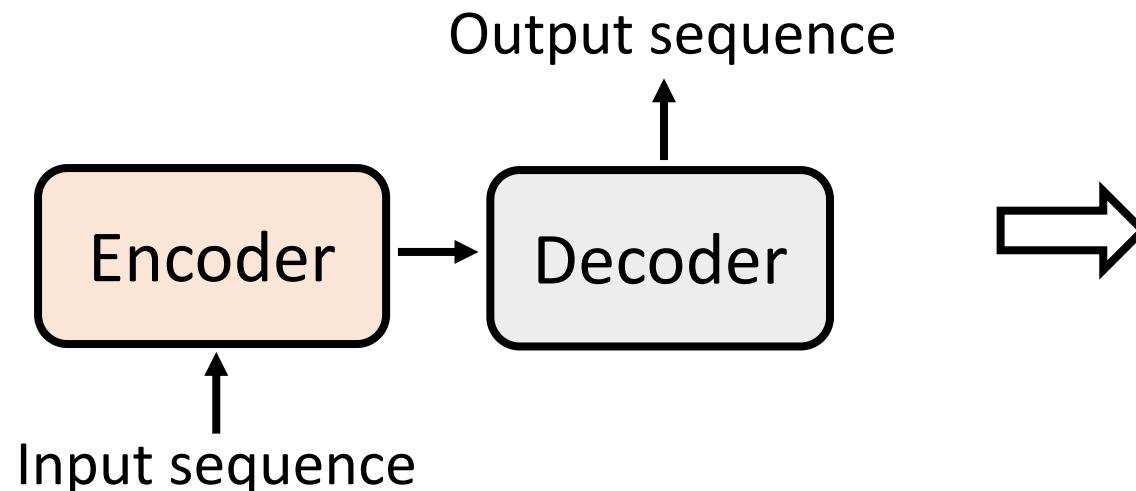
# Transformer: Beyond CNNs and RNNs

- **Overview**
- **Self-attention**
- **Encoder-decoder architecture**
- **Applications**



# Transformer

- Sequence-to-sequence (Seq2seq)
- Encoder and Decoder Stacks



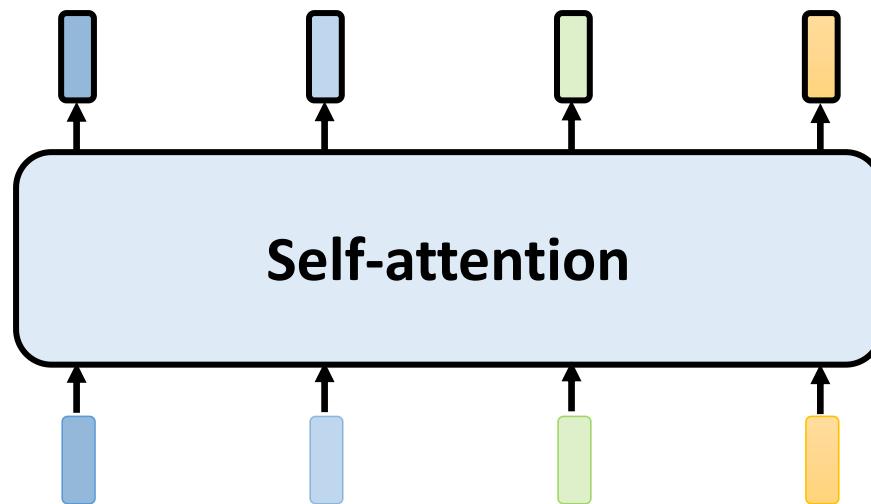
[Attention is all you need.](#)



# Self-Attention

- How to consider the **context** of the whole sequence?

with context



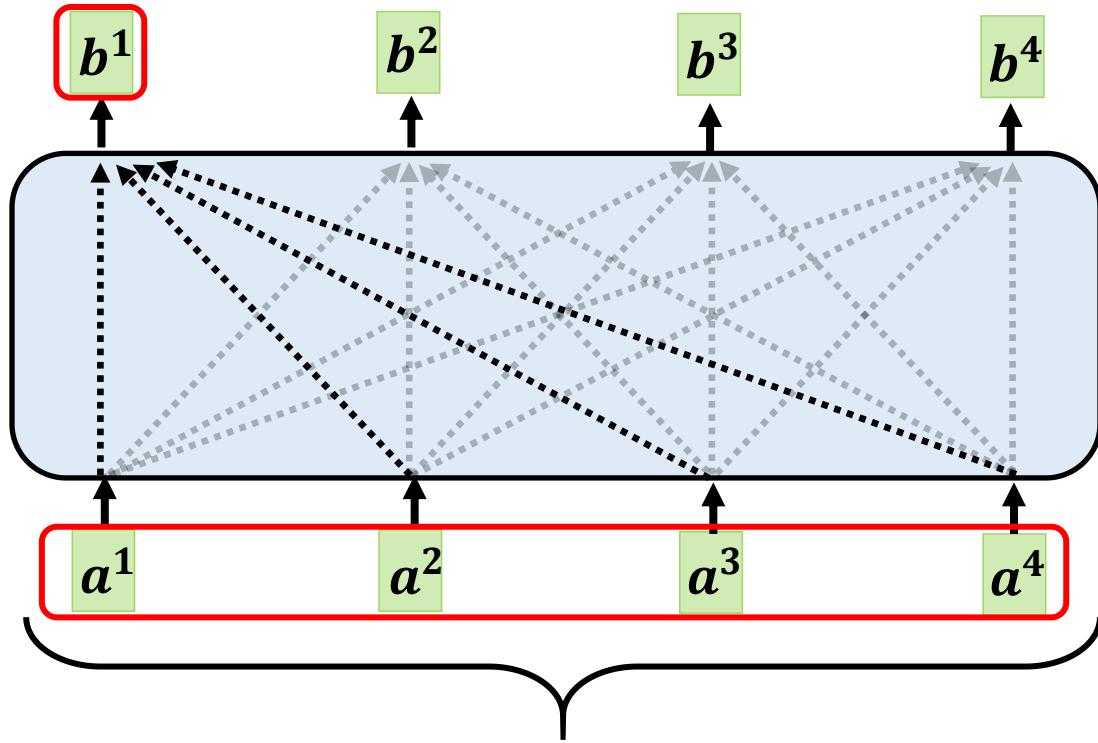
Input sequence:

“The **animal** didn't cross the **street** because **it** was too tired”

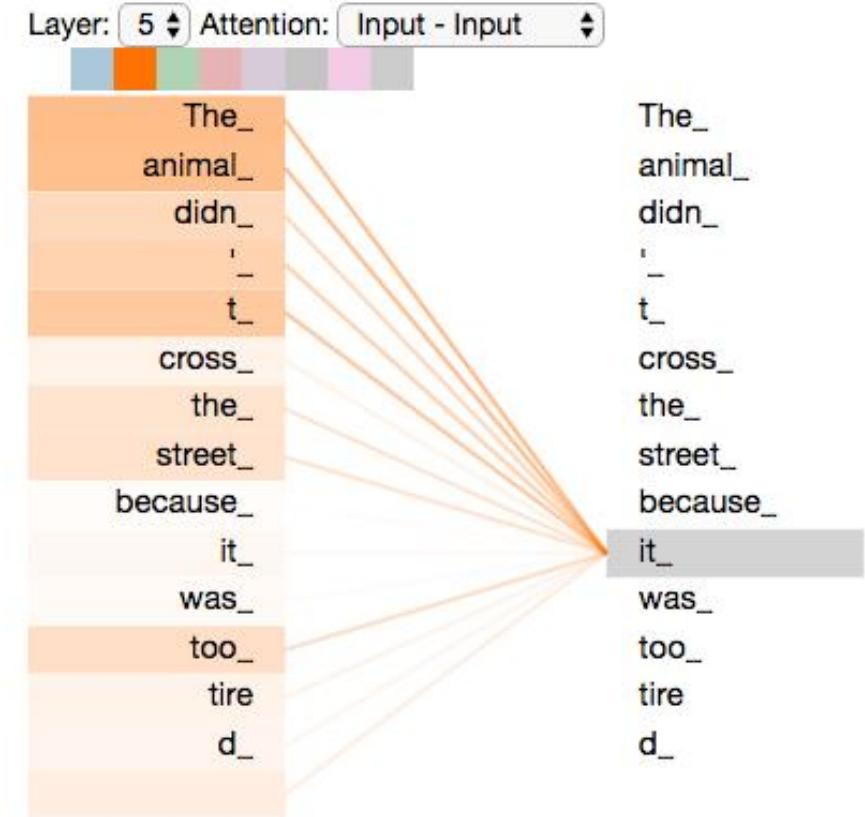
relevant?



# Self-Attention



Can be either input or a hidden layer



[Tensor2Tensor notebook](#)



# Self-Attention

- Calculate self-attention using vectors

Query:  $q^i = W^q a^i$

Key:  $k^i = W^k a^i$

Value:  $v^i = W^v a^i$

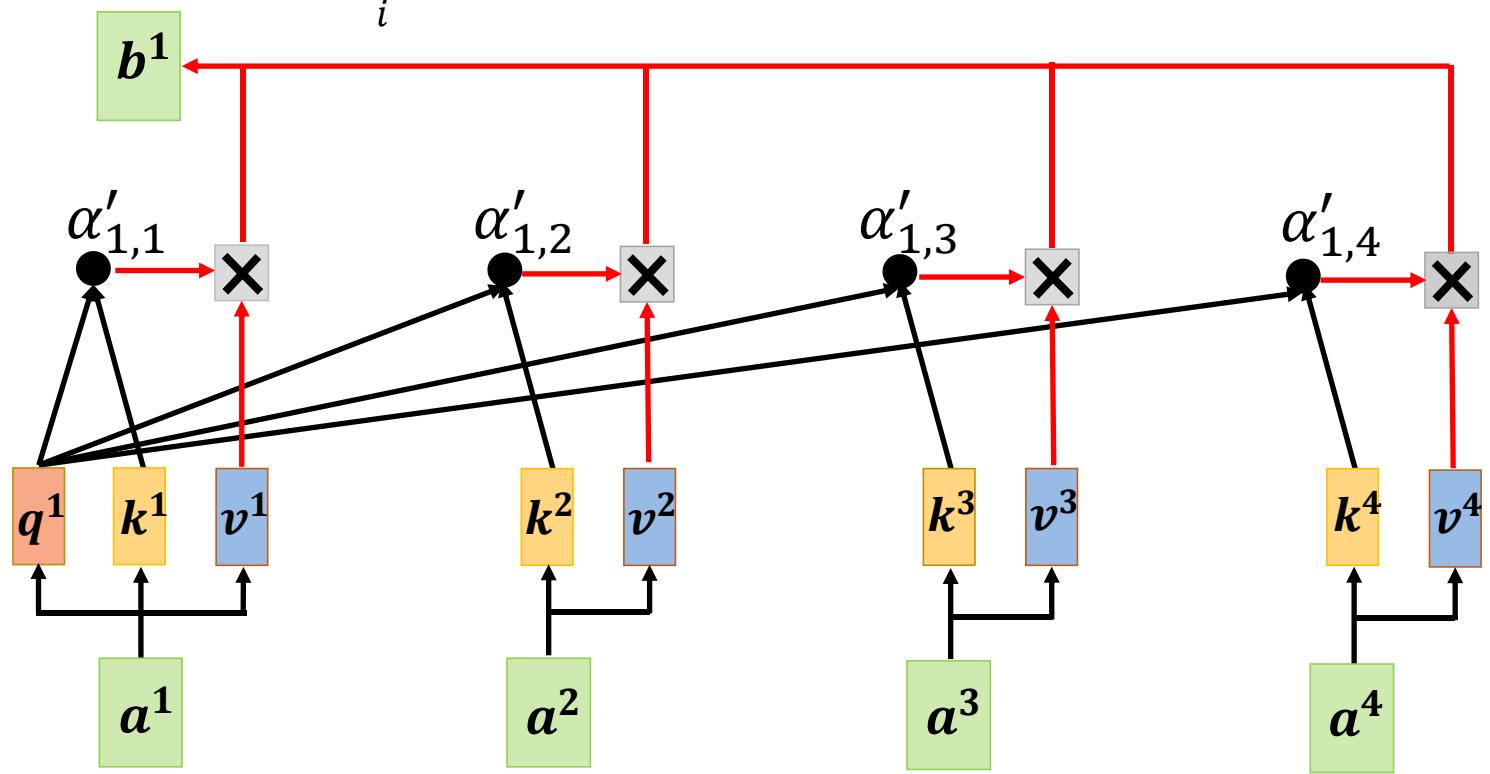
$$\alpha_{1,i} = \frac{q^1 \cdot k^i}{\sqrt{d_k}}$$

Dot-product

$$\alpha'_{1,i} = \text{Softmax}(\alpha_{1,i}) = \frac{\exp(\alpha_{1,i})}{\sum_j \exp(\alpha_{1,j})}$$

$$b^1 = \sum_i \alpha'_{1,i} v^i$$

Extract information based on attention scores

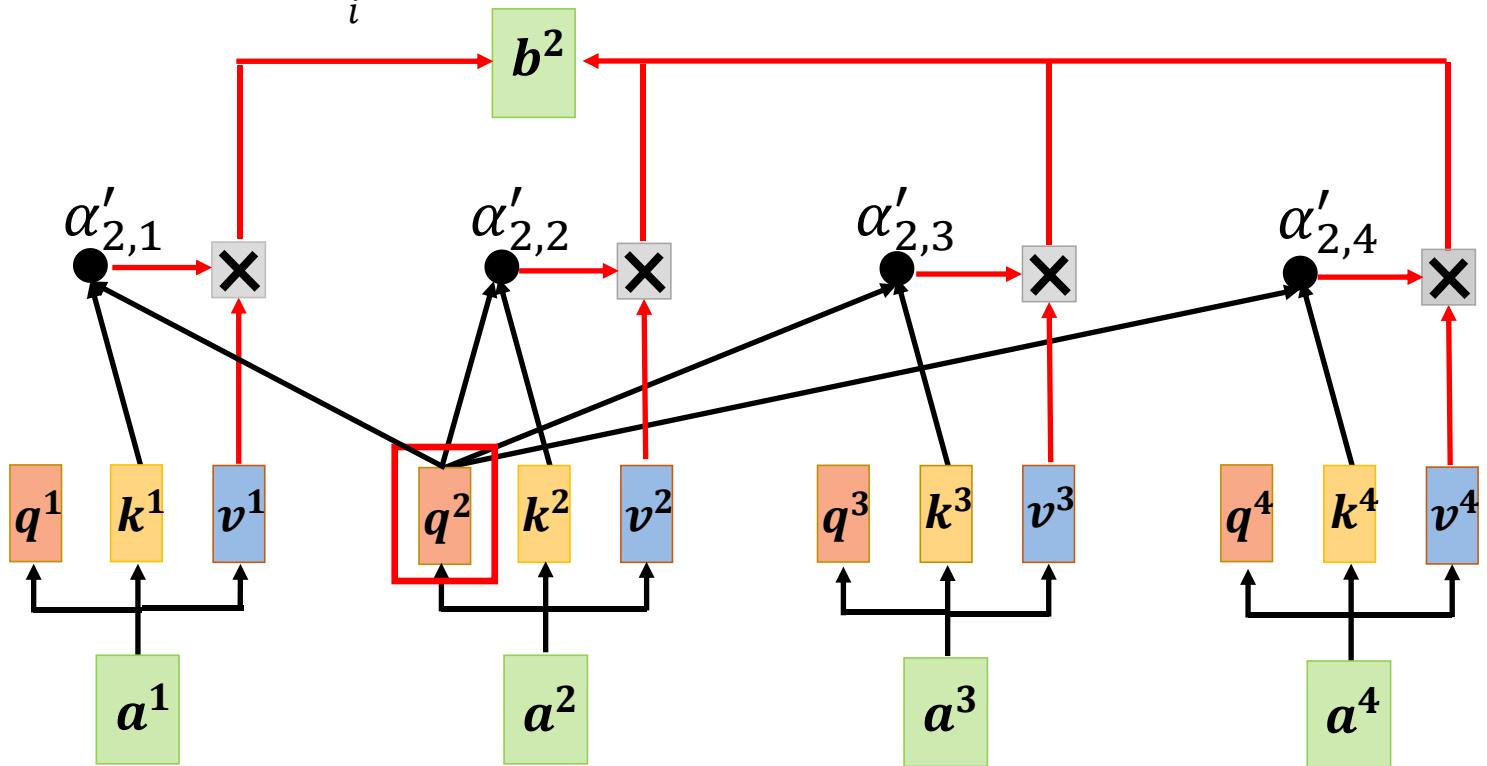




# Self-Attention

- Calculate self-attention using vectors

$$b^2 = \sum_i \alpha'_{2,i} v^i$$





# Self-Attention

- Matrix Calculation

$$\begin{matrix} q^1 & q^2 & q^3 & q^4 \\ \text{---} \\ Q \end{matrix} = \boxed{W^q} \begin{matrix} a^1 & a^2 & a^3 & a^4 \\ \text{---} \\ I \end{matrix}$$

$$\begin{matrix} k^1 & k^2 & k^3 & k^4 \\ \text{---} \\ K \end{matrix} = \boxed{W^k} \begin{matrix} a^1 & a^2 & a^3 & a^4 \\ \text{---} \\ I \end{matrix}$$

$$\begin{matrix} v^1 & v^2 & v^3 & v^4 \\ \text{---} \\ V \end{matrix} = \boxed{W^v} \begin{matrix} a^1 & a^2 & a^3 & a^4 \\ \text{---} \\ I \end{matrix}$$

Parameters to be learned

$$\begin{matrix} \alpha'_{1,1} & \alpha'_{2,1} & \alpha'_{3,1} & \alpha'_{4,1} \\ \alpha'_{1,2} & \alpha'_{2,2} & \alpha'_{3,2} & \alpha'_{4,2} \\ \alpha'_{1,3} & \alpha'_{2,3} & \alpha'_{3,3} & \alpha'_{4,3} \\ \alpha'_{1,4} & \alpha'_{2,4} & \alpha'_{3,4} & \alpha'_{4,4} \end{matrix}$$

$A'$

softmax

$$\begin{matrix} \alpha_{1,1} & \alpha_{2,1} & \alpha_{3,1} & \alpha_{4,1} \\ \alpha_{1,2} & \alpha_{2,2} & \alpha_{3,2} & \alpha_{4,2} \\ \alpha_{1,3} & \alpha_{2,3} & \alpha_{3,3} & \alpha_{4,3} \\ \alpha_{1,4} & \alpha_{2,4} & \alpha_{3,4} & \alpha_{4,4} \end{matrix}$$

$A$

$$\begin{matrix} k^1 \\ k^2 \\ k^3 \\ k^4 \end{matrix} = \begin{matrix} q^1 & q^2 & q^3 & q^4 \\ \text{---} \\ Q \end{matrix}$$
  

$$\begin{matrix} k^1 \\ k^2 \\ k^3 \\ k^4 \end{matrix} = \begin{matrix} K^T \\ \text{---} \\ K \end{matrix}$$

$$\begin{matrix} b^1 & b^2 & b^3 & b^4 \\ \text{---} \\ O \end{matrix} = \begin{matrix} v^1 & v^2 & v^3 & v^4 \\ \text{---} \\ V \end{matrix}$$

$$\begin{matrix} \alpha'_{1,1} & \alpha'_{2,1} & \alpha'_{3,1} & \alpha'_{4,1} \\ \alpha'_{1,2} & \alpha'_{2,2} & \alpha'_{3,2} & \alpha'_{4,2} \\ \alpha'_{1,3} & \alpha'_{2,3} & \alpha'_{3,3} & \alpha'_{4,3} \\ \alpha'_{1,4} & \alpha'_{2,4} & \alpha'_{3,4} & \alpha'_{4,4} \end{matrix}$$

$A'$



# Self-Attention

- Matrix Calculation

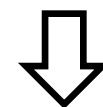
$$Q = W^q \begin{matrix} I \\ I \end{matrix}$$

$$K = W^k \begin{matrix} I \\ I \end{matrix}$$

$$V = W^v \begin{matrix} I \\ I \end{matrix}$$

$$A' \leftarrow \text{softmax} \quad A = \frac{Q \cdot K^T}{\sqrt{d_k}}$$

$$O = V \cdot A'$$

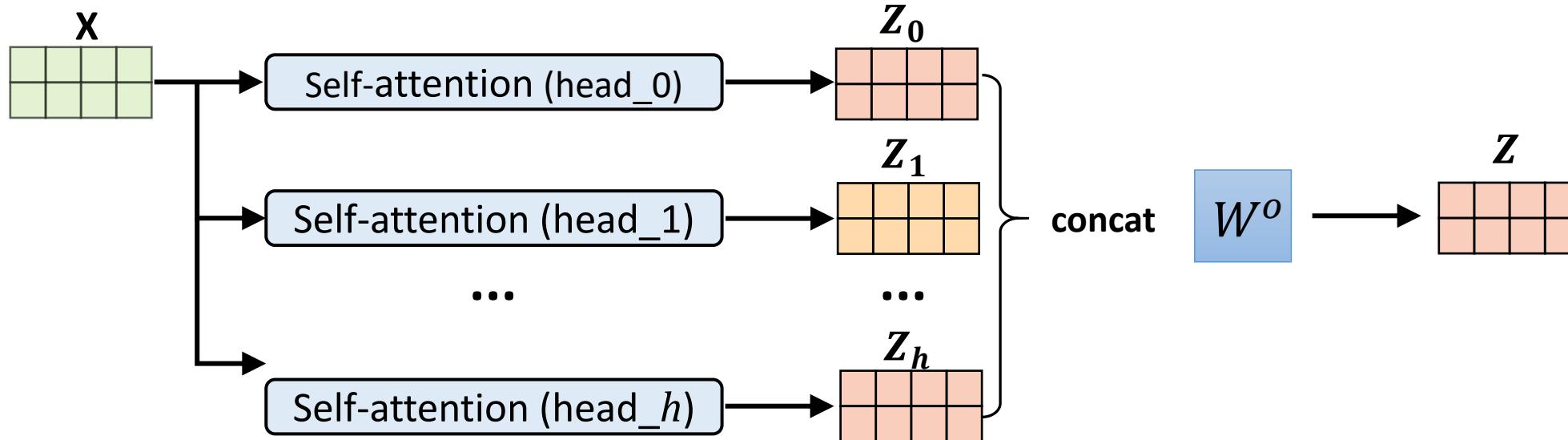


$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$



# Multi-head Self-Attention

- Allows the model to jointly attend to information from **different representation subspaces** at different positions.
- Capture **different types** of relevance.



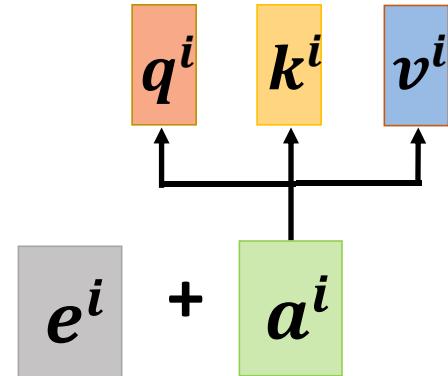
$$\text{MultiHead}(Q, K, V) = \text{Concat}(\text{head}_1, \dots, \text{head}_h)W^o$$

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



# Position Encoding

- No position information in self-attention.
- Each position has a unique positional vector  $e^i$



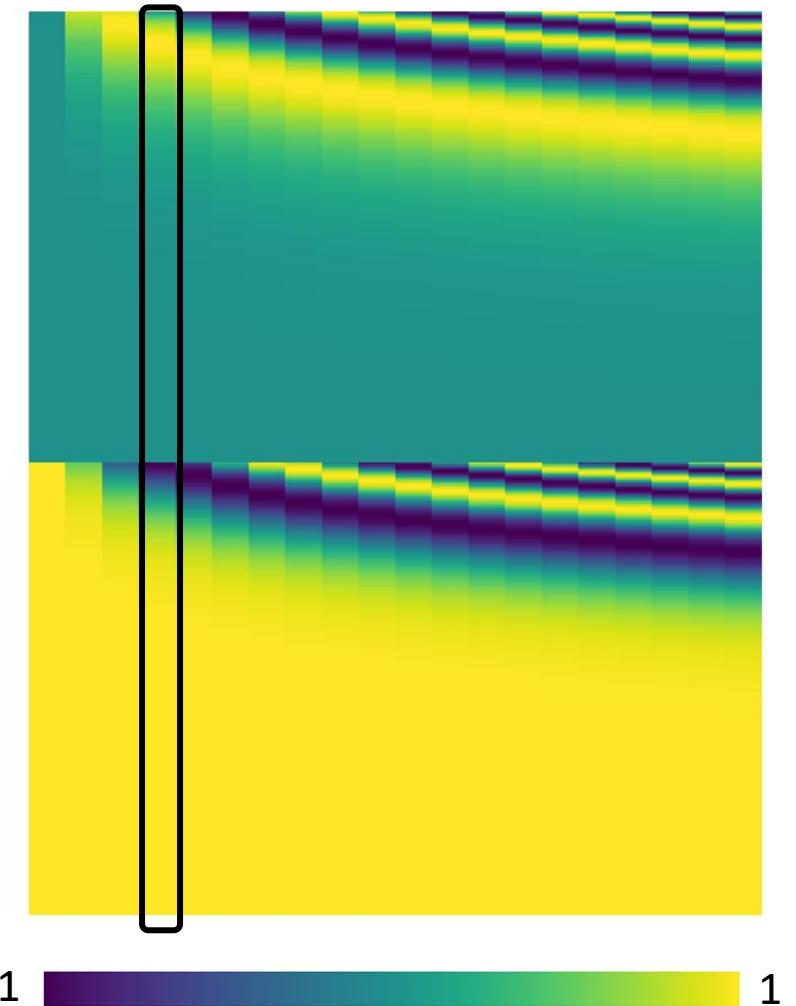
$$PE_{(pos, 2i)} = \sin(pos/1000^{2i/d_{model}})$$

$$PE_{(pos, 2i+1)} = \cos(pos/1000^{2i/d_{model}})$$

$d_{model}$ : Dimension of the output embedding space

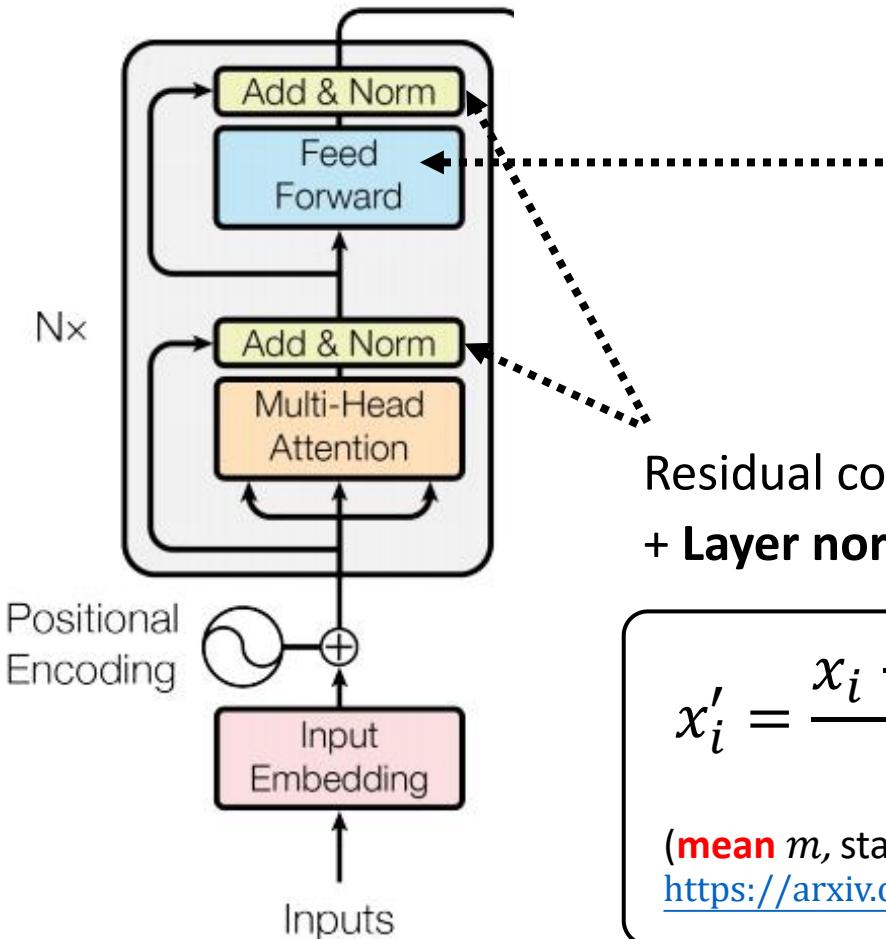
$pos$ : Position of an object in the input sequence

Each column represents a positional vector  $e^i$





# Transformer Encoder



**Fully connected feed-forward network**

$$FFN(x) = \max(0, xW_1 + b_1)W_2 + b_2$$

(two linear transformations with a ReLU activation)

Residual connection  
+ **Layer normalization**

$$x'_i = \frac{x_i - m}{\sigma}$$

(**mean**  $m$ , standard **deviation**  $\sigma$ )  
<https://arxiv.org/abs/1607.06450>

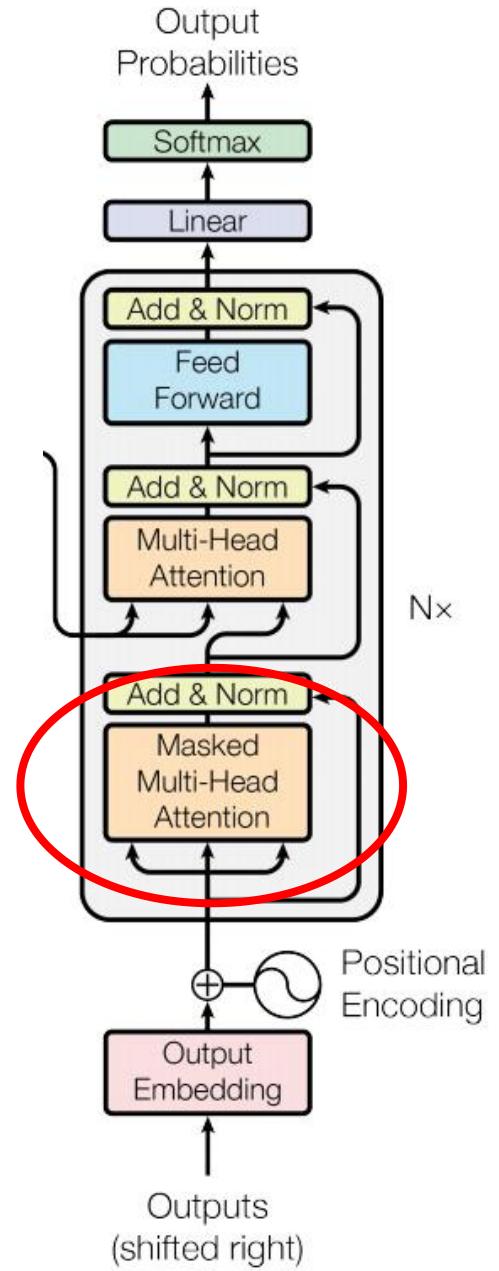


# Transformer Decoder

## Masked multi-head self-attention

- To prevent leftward information flow in the decoder to preserve the **auto-regressive** property.

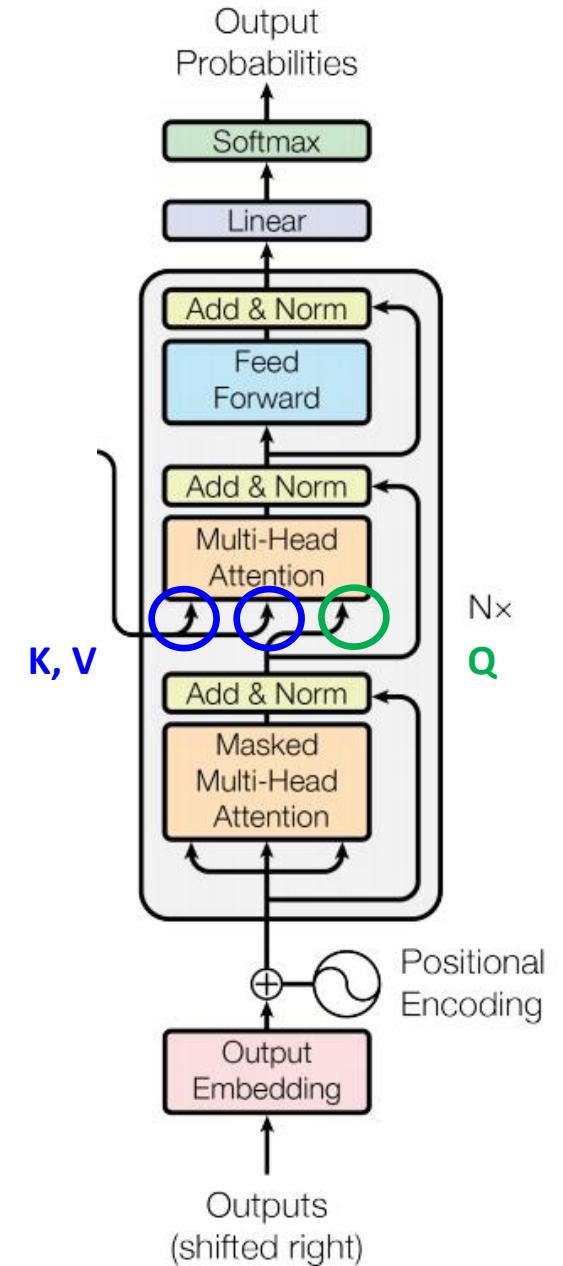
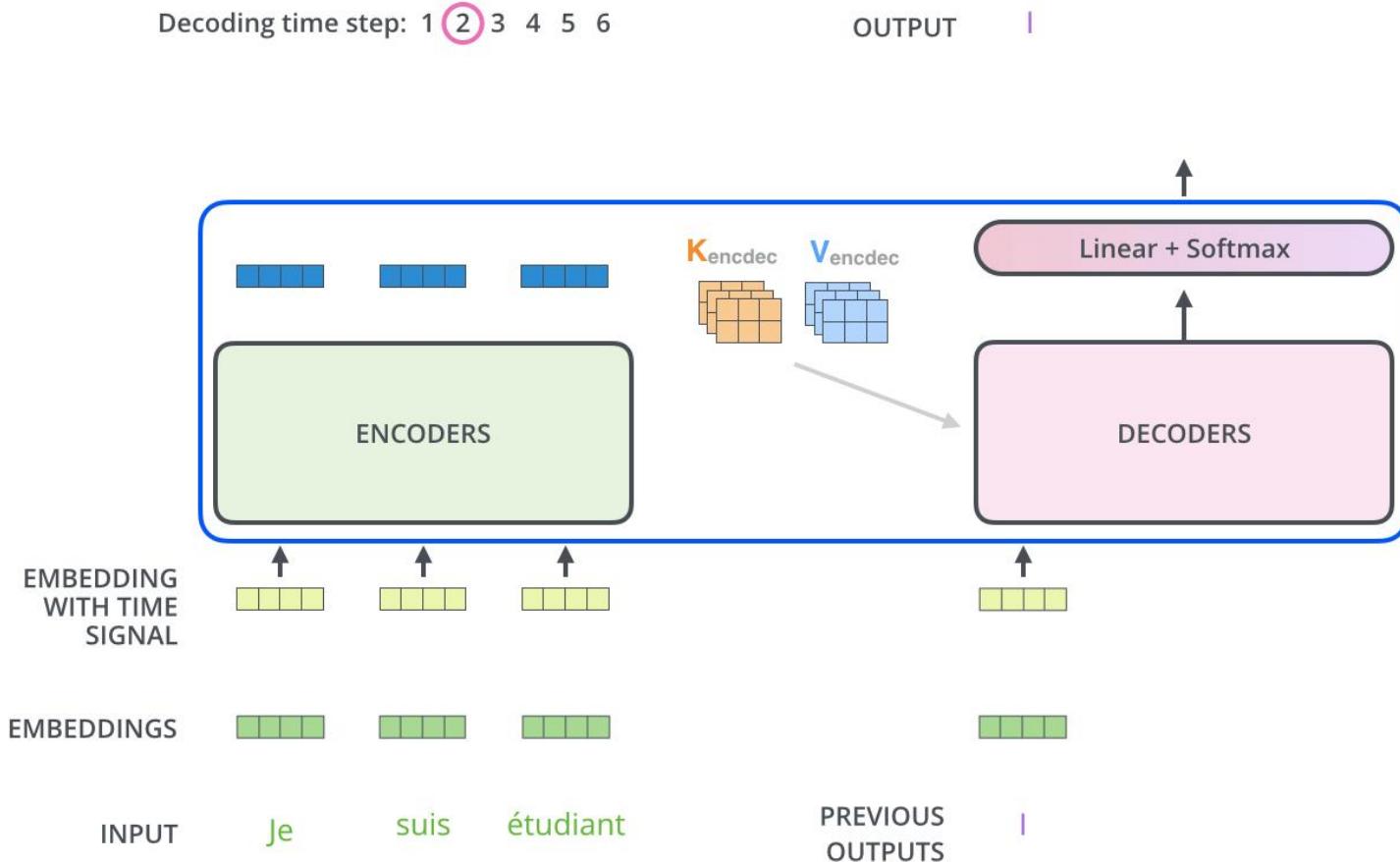
$$\begin{array}{cccc} \alpha_{1,1} & \alpha_{2,1} & \alpha_{3,1} & \alpha_{4,1} \\ \alpha_{1,2} & \alpha_{2,2} & \alpha_{3,2} & \alpha_{4,2} \\ \alpha_{1,3} & \alpha_{2,3} & \alpha_{3,3} & \alpha_{4,3} \\ \alpha_{1,4} & \alpha_{2,4} & \alpha_{3,4} & \alpha_{4,4} \end{array} \otimes \begin{array}{c} \text{Mask} \end{array} = \begin{array}{cccc} \alpha_{1,1} & \text{black} & \text{black} & \text{black} \\ \alpha_{1,2} & \alpha_{2,2} & \text{black} & \text{black} \\ \alpha_{1,3} & \alpha_{2,3} & \alpha_{3,3} & \text{black} \\ \alpha_{1,4} & \alpha_{2,4} & \alpha_{3,4} & \alpha_{4,4} \end{array}$$





# Transformer Decoder

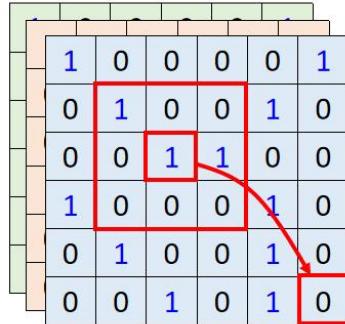
## Autoregressive





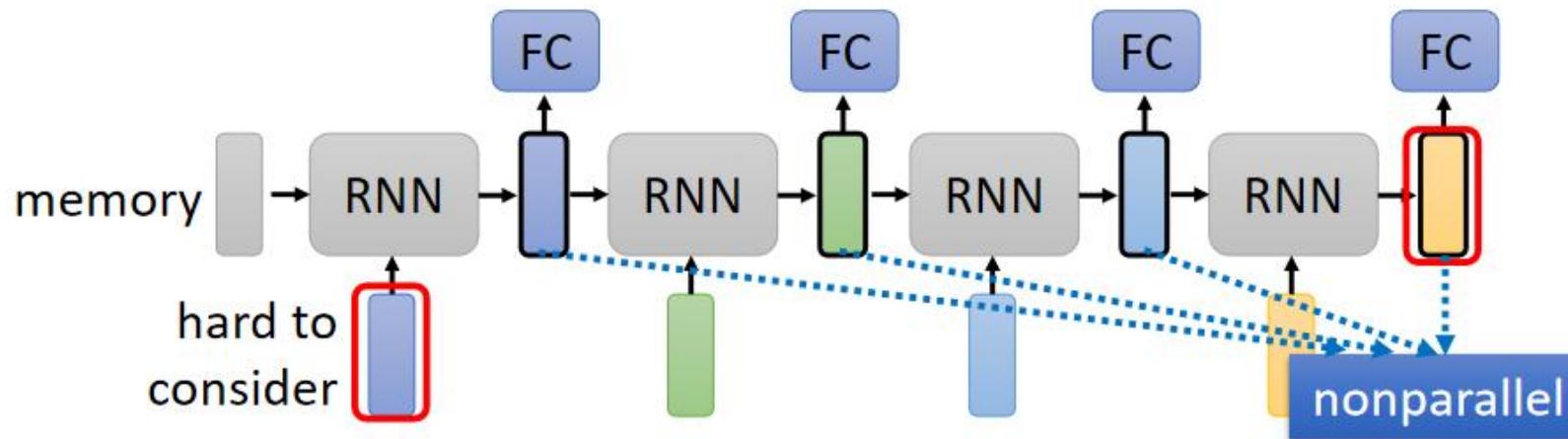
# Self-Attention vs. CNN/RNN

CNN



- CNN: self-attention that can only attend in a receptive field.
- RNN: hard to learn long-range dependencies;  
nonparallel, more computation time.

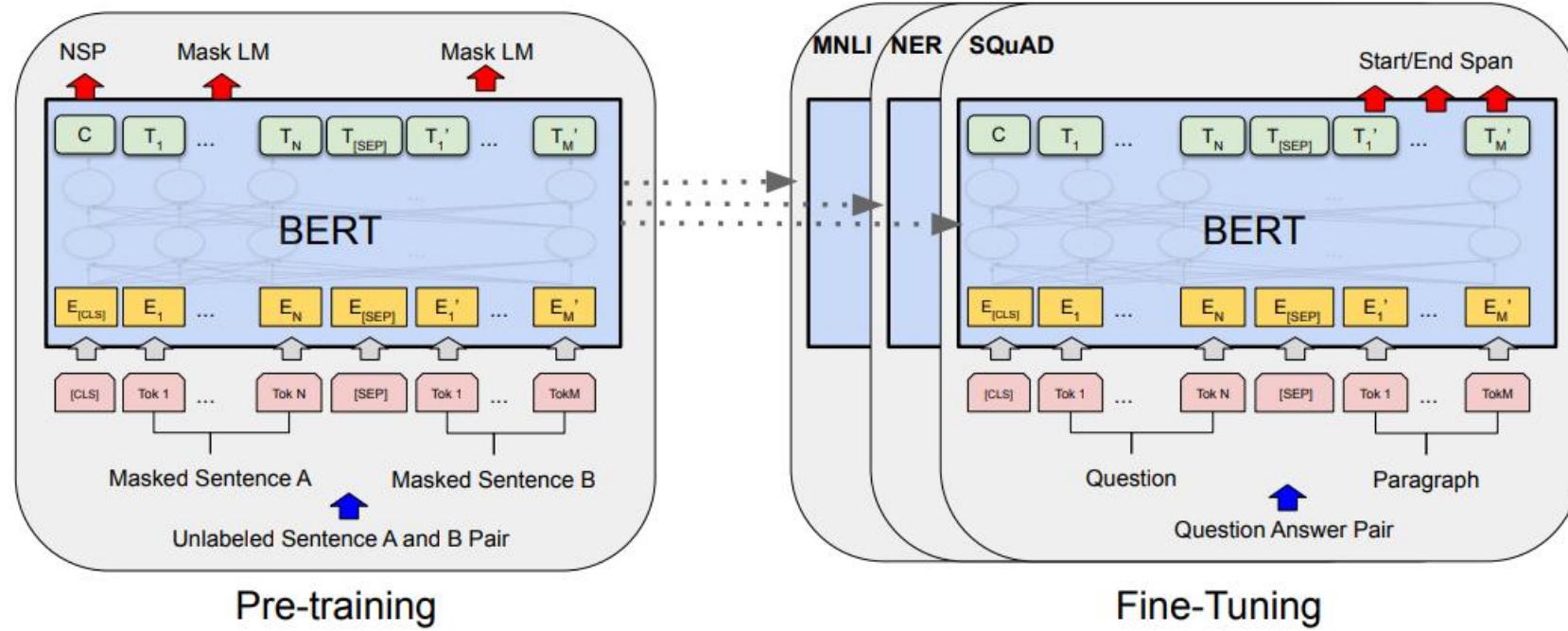
RNN





# Transformer in Neural Language Processing (NLP)

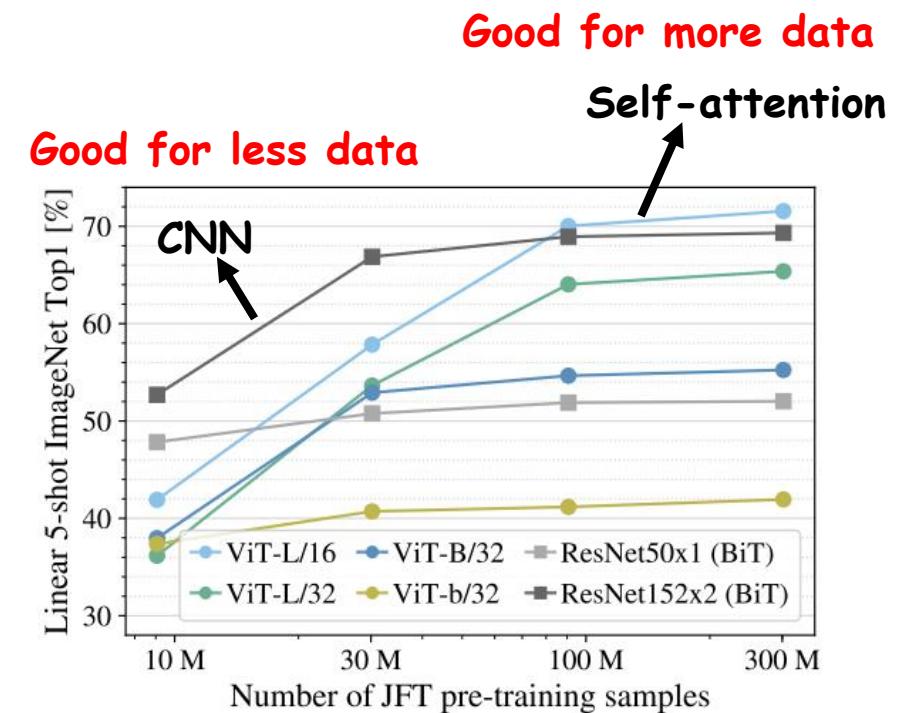
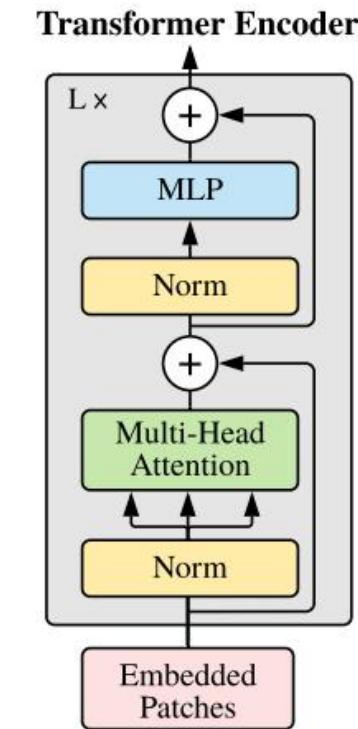
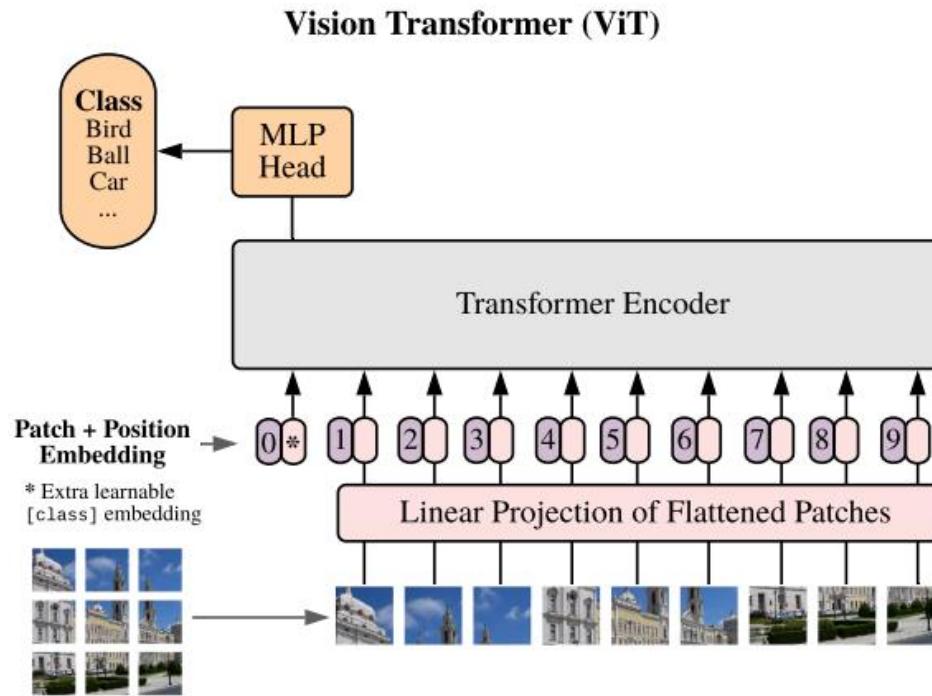
- BERT: same network architecture as **transformer encoder**.  
Masked language modeling (MLM) pre-training





# Transformer in Computer Vision (CV)

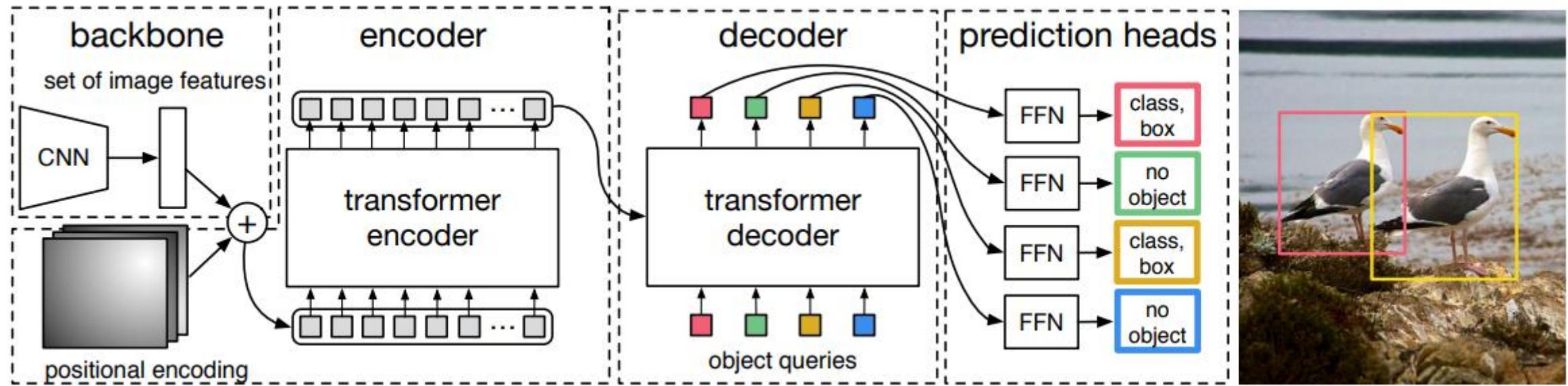
- Transformers for Image Recognition





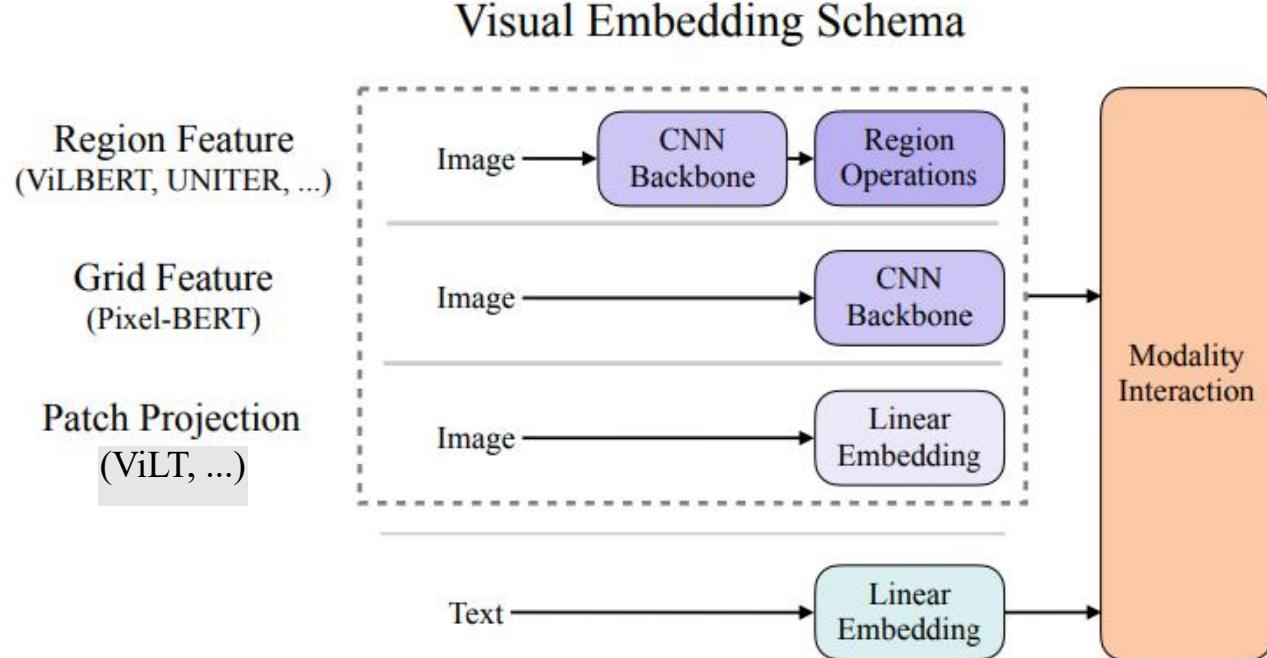
# Transformer in Computer Vision (CV)

- Transformers for Object detection





# Transformer for Vision-Language (VL)

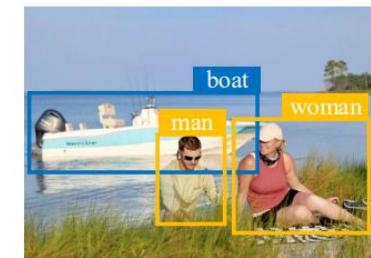


ViLBERT: Pretraining Task-Agnostic Visiolinguistic Representations for Vision-and-Language Tasks. NeurIPS 2019

UNITER: UNiversal Image-TExt Representation Learning. ECCV 2020

Pixel-BERT: Aligning Image Pixels with Text by Deep Multi-Modal Transformers. ArXiv:2004.00849

ViLT: Vision-and-Language Transformer Without Convolution or Region Supervision. ICML 2021



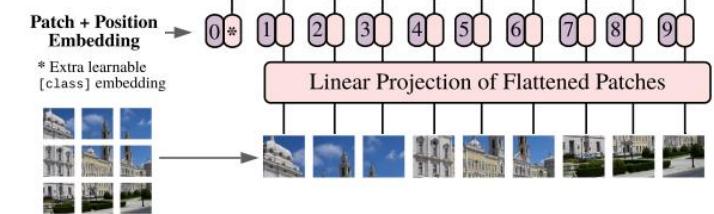
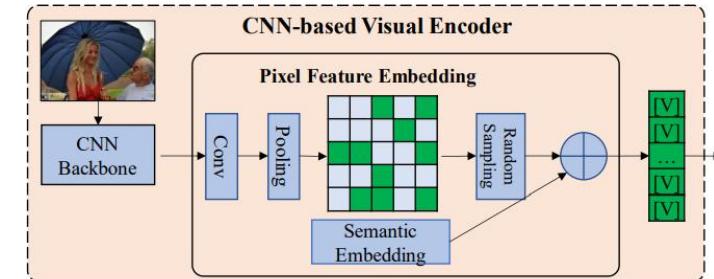
## Task I: TR

Baseline: A couple sit in a boat on the sea. ✗

Ours: A couple sit on the shore next to a boat on the sea. ✓

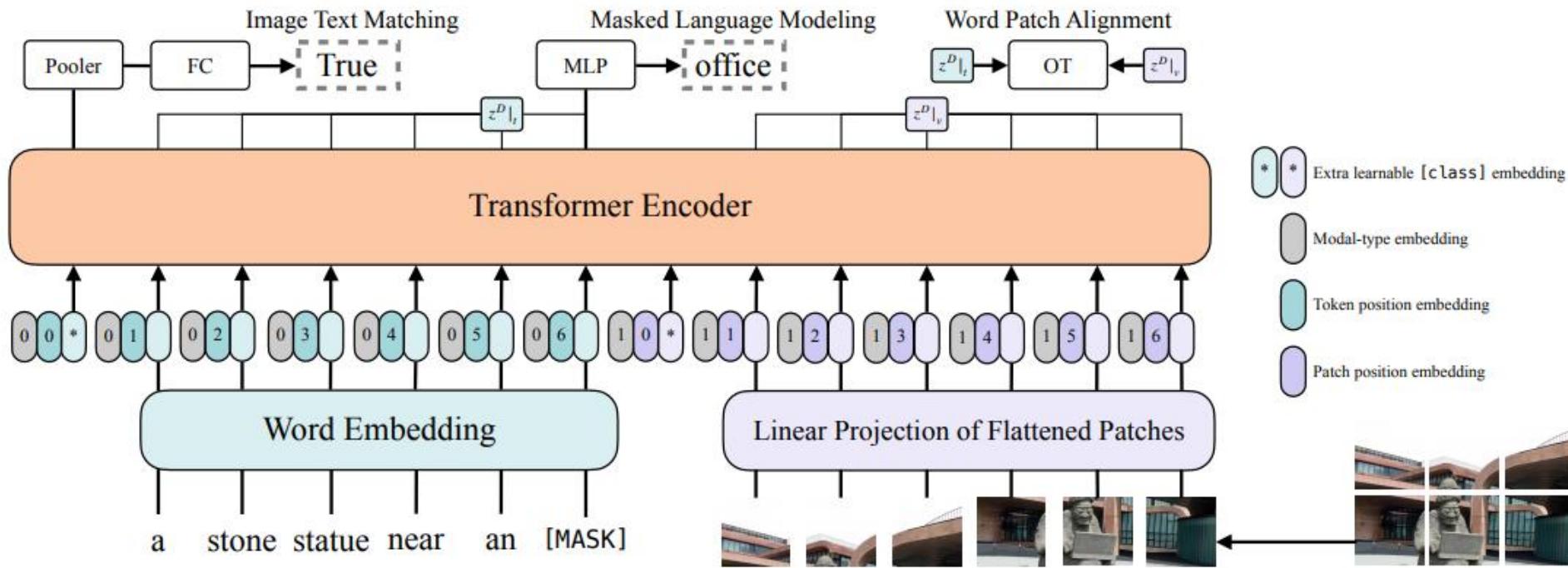
## Task II: VQA

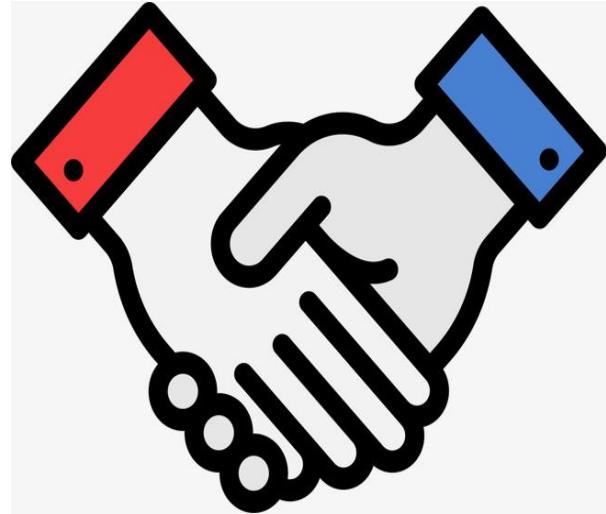
Q: What are the people doing?  
Baseline: Boating. ✗  
Ours: Chatting. ✓





# Transformer for Vison-Language (VL)





# Thanks



[zhengf@sustech.edu.cn](mailto:zhengf@sustech.edu.cn)