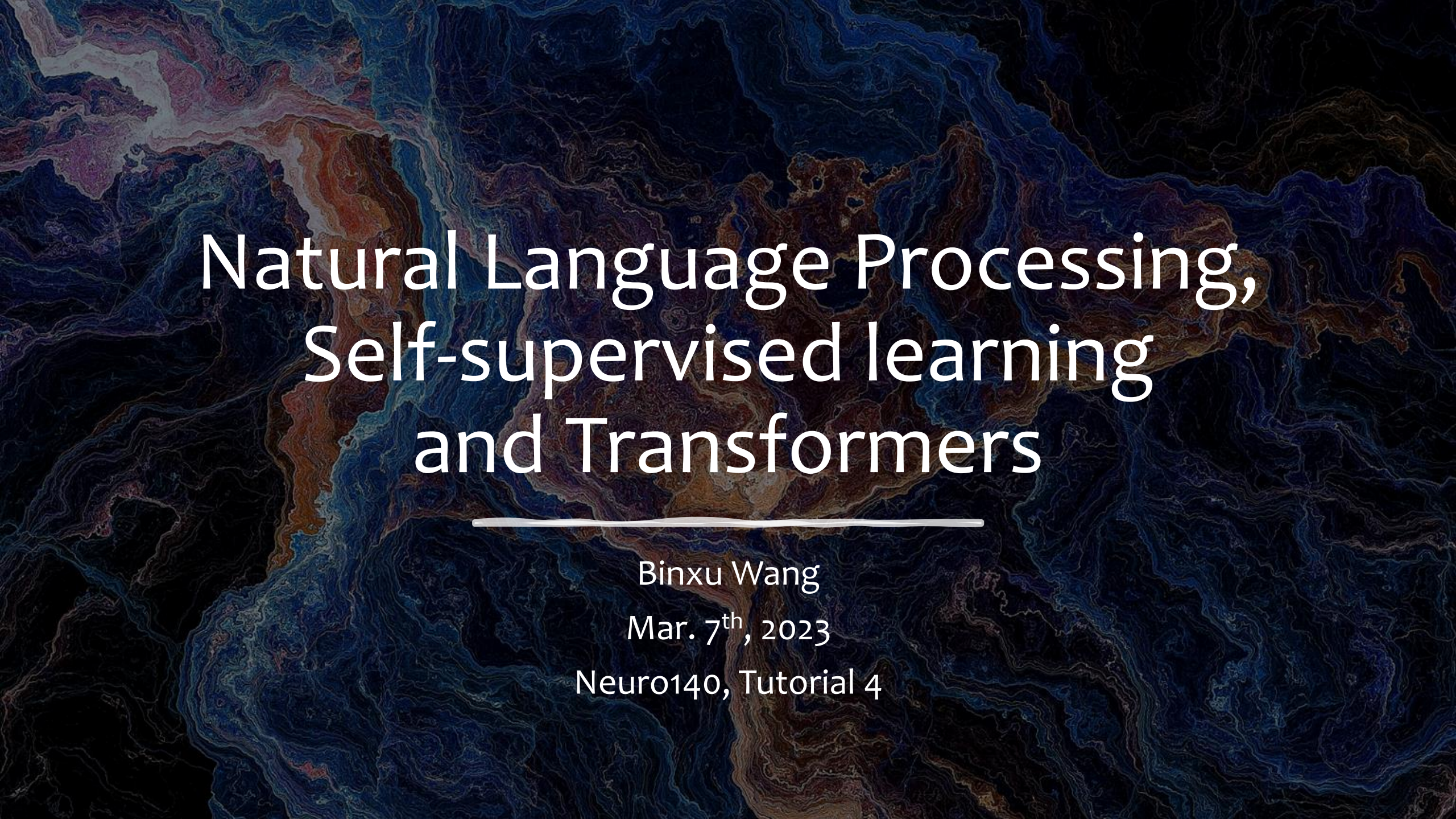


Natural Language Processing, Self-supervised learning and Transformers

Binxu Wang

Mar. 7th, 2023

Neuro140, Tutorial 4

Part I
Beyond Supervised Learning

How to learn without labelling

Paradigm of (supervised) machine learning

- Collect datasets of $\{X_i, y_i\}$

IMAGENET

- Fit a neural network f_θ to approximate y

$$f_\theta(X_i) \approx y_i$$

Apple, id=5

  - Minimize the loss over training set.

$$\min_\theta \sum_i \mathcal{L}(f_\theta(X_i), y_i)$$

$$f_\theta : X \to y$$

- Use f_θ to predict y for new samples.

Where does the supervision (labels) come from ?

- Supervision is distilled and denoised human perception.
- Labelling takes tremendous amount of human effort.
- For some tasks, human don't have good labels.
  - Depth estimation.

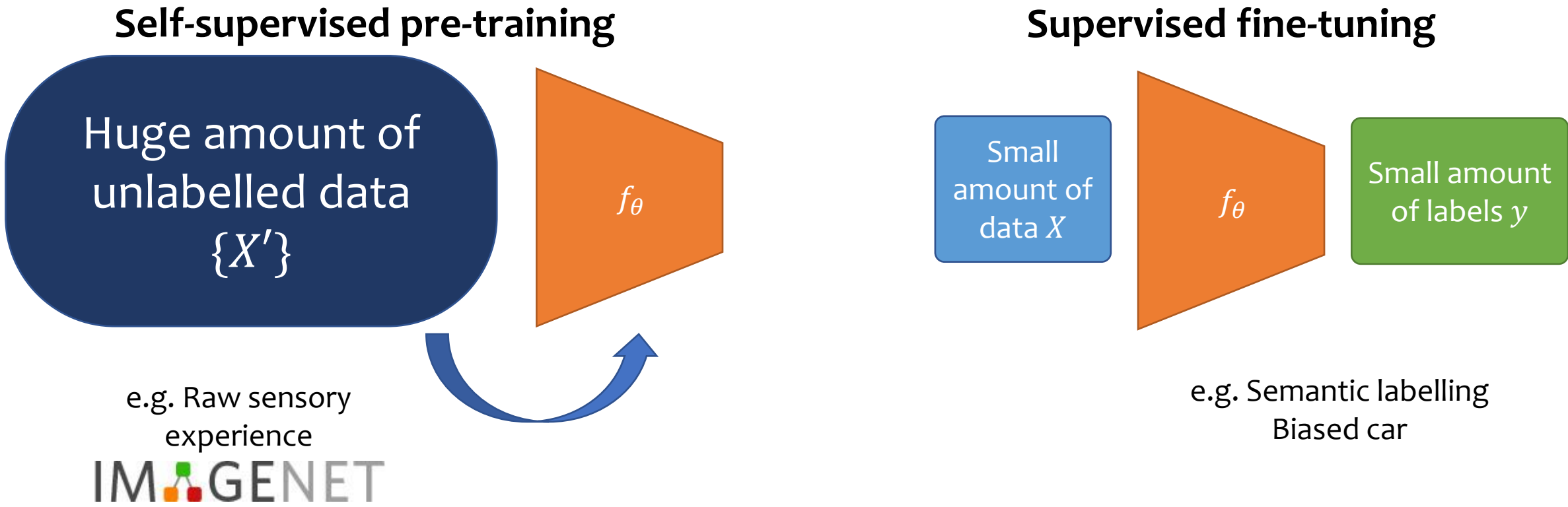
- 14 Million images annotated
- 10 annotations per images by Mturk workers

Infants learn without much ***external*** supervision…

- Infants' learning of language and vision doesn't require millions of labels.

- They learn from the statistics of the data itself.

Self-supervised learning paradigm



- Objective of pretraining
  - Learn ***good representation*** , *s.t.*
    - downstream tasks can be easily solved on these representations. (e.g. linear decodable)
    - with fewer labeled samples

No supervision?
Create supervised tasks and solve them.

- Self-supervised learning
  - Create labels from X_i
  - Learn to predict synthetic labels $X_i \rightarrow \tilde{y}_i$

Self-prediction
(Masked Autoencoder)

Contrastive learning
(SimCLR)

Part II
Natural Language Processing 101

Where self-supervised learning triumphed

Two Pillars of NLP

Representation

- How to represent language to machines?

Modelling

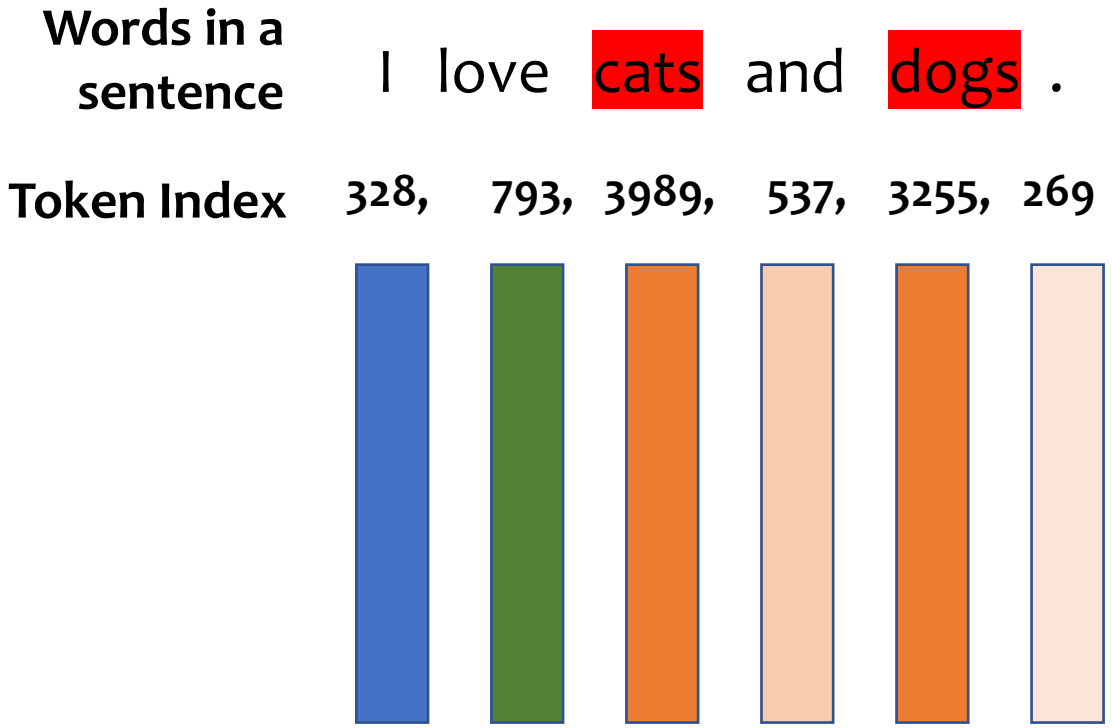
- How to model language?

Simultaneously solved by ML models,
e.g. transformers

Inspired by MIT 6.8610

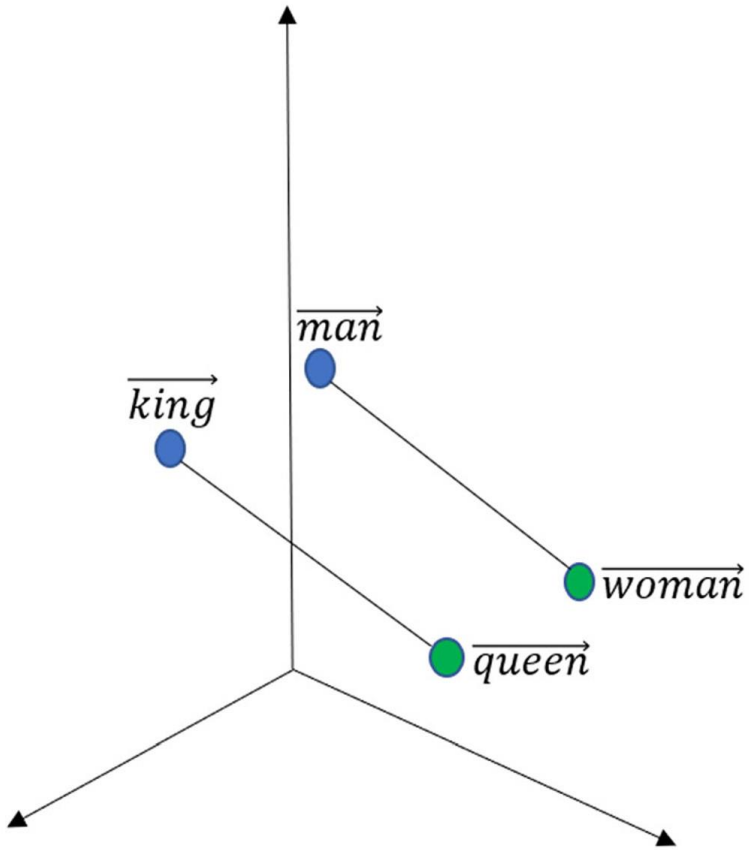
Representation: Word as Vectors

- Unlike pixel, meaning of word are not explicitly in the characters.
- Word can be represented as index in dictionary
  - But index is also meaningless.
- Represent words in a vector space
  - Vector geometry ⇒ semantic relation.
  - Vector clustering ⇒ semantic similarity

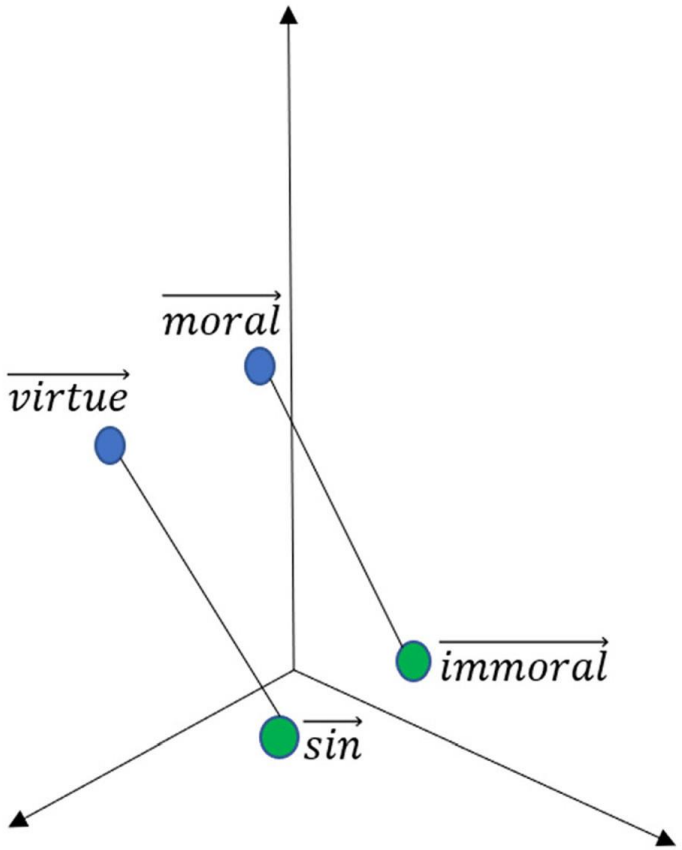


Word as Vectors:

Vector space geometry captures semantic relationship



a "gender" dimension

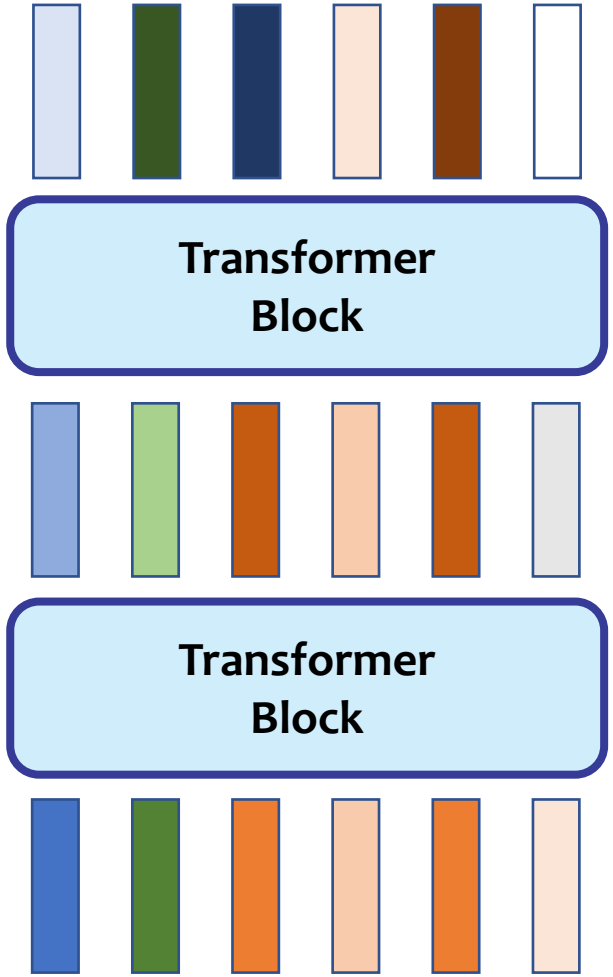


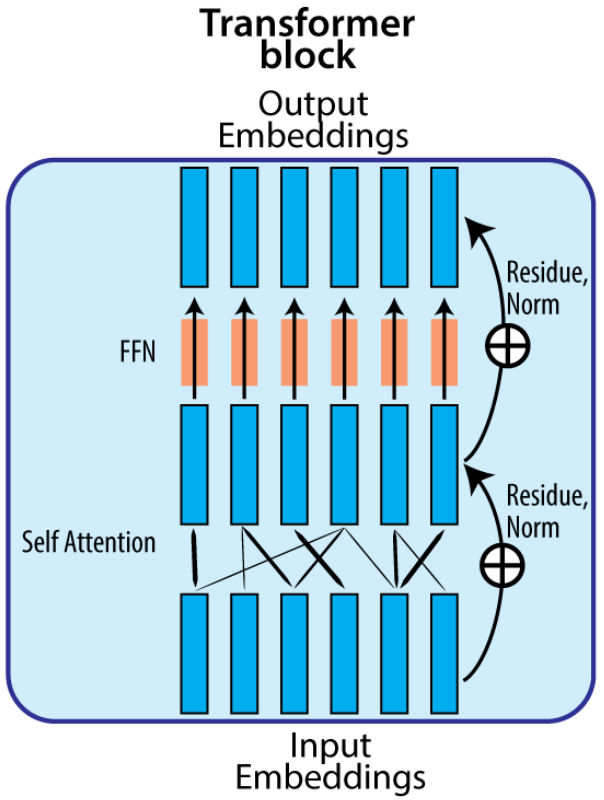
a "morality" dimension

Arseniev-Koehler, Alina. "Theoretical foundations and limits of word embeddings: what types of meaning can they capture?." *Sociological Methods & Research* (2021)

Word Vector in Context: RNN / Transformers

- Meaning of word depends on context.
  - "I can buy a can of fish."
- Word vectors should depend on context.
- Transformers / RNN let each word "absorb" influence from other words to be *contextualized*





Language modelling

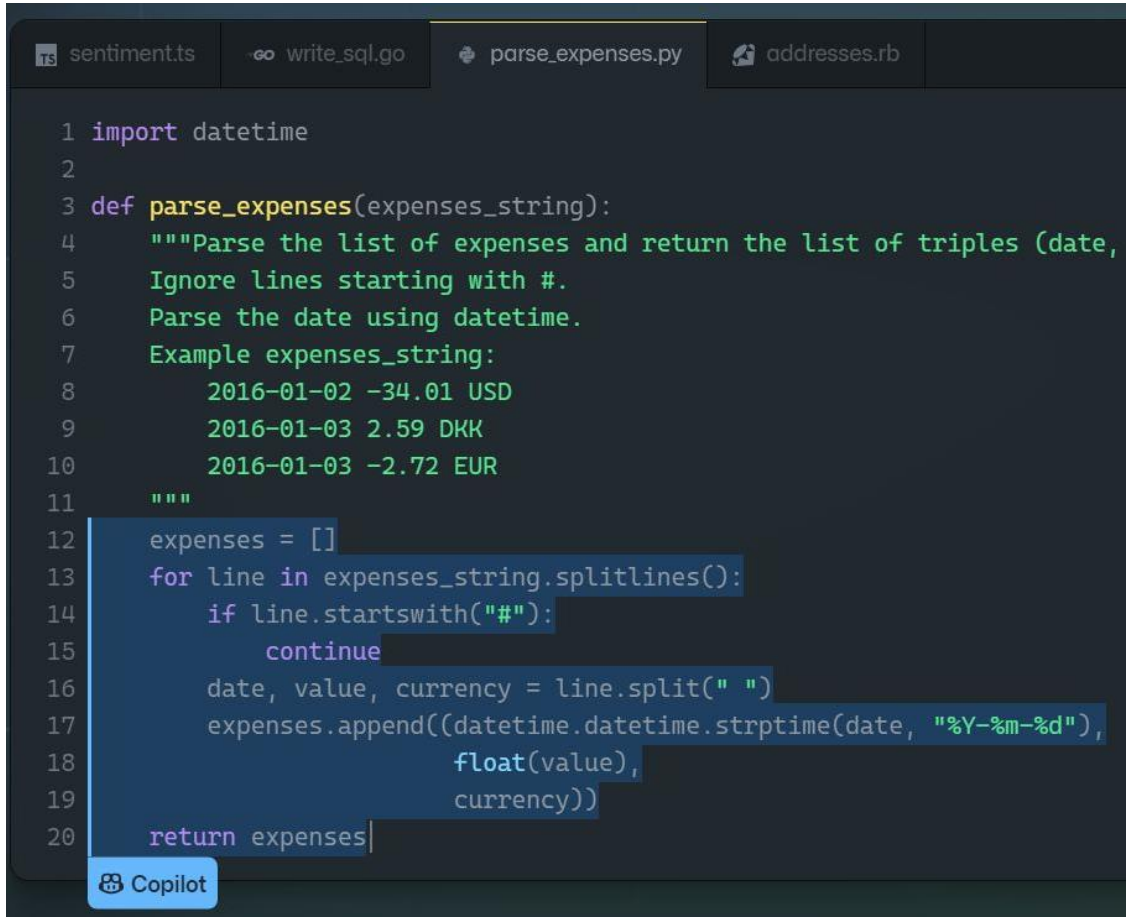
- To estimate probability of any sequence of words
  - $p(\text{"This is a fluffy dog."}) = p(w_1 w_2 w_3 w_4 w_5)$
- Given such model, we can
  - Guess missing words
    - "This _ a fluffy dog.“
    - $\arg\max p(w_2 | w_1 w_3 w_4 w_5) = \text{"}is\text{"}$
  - Predict next words
    - "This is a fluffy …
    - $\arg\max p(w_5 | w_1 w_2 w_3 w_4) = \text{"}cat\text{"}$

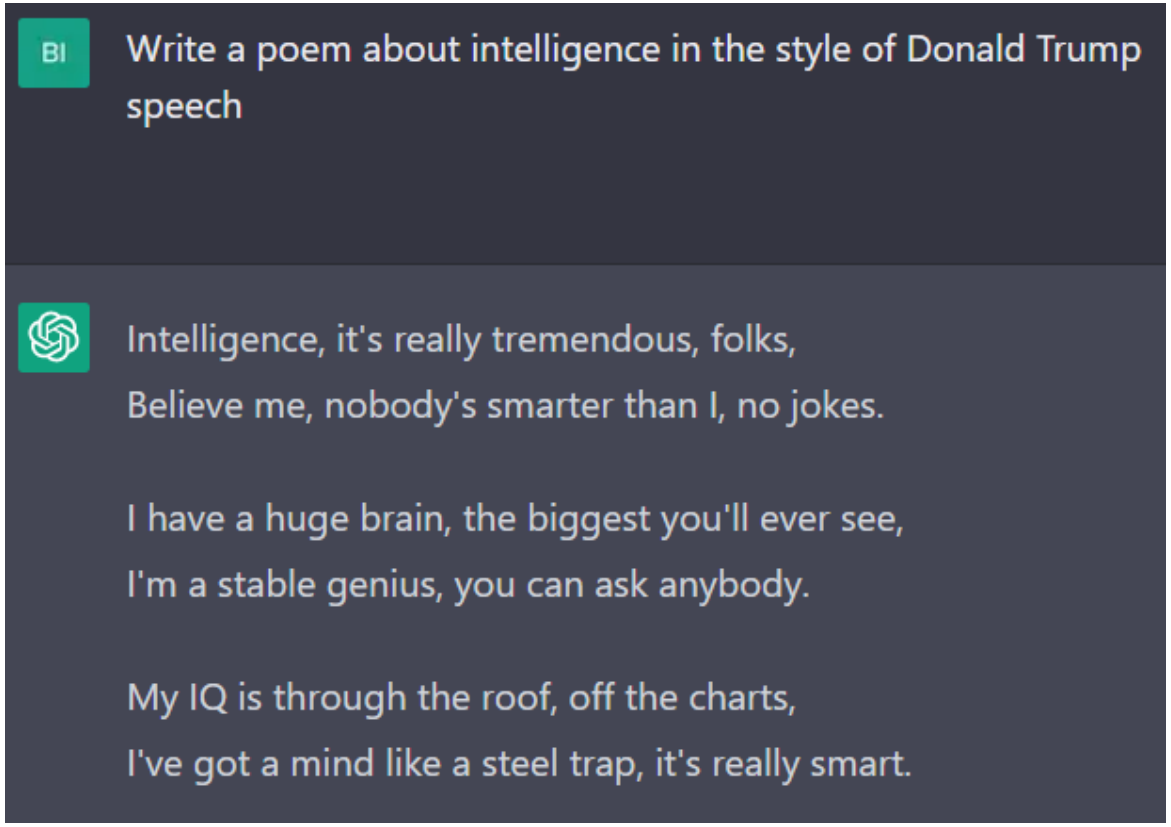
Language model can help …

- Text generation
  - Code completion
  - Text summary
  - Chat bot
  - Translation
  - Grammar correction
- Text Classification
  - Detect spam
  - Authorship attribution

Inspired by MIT 6.8610

E.g. Github Copilot, ChatGPT





How to learn language model?

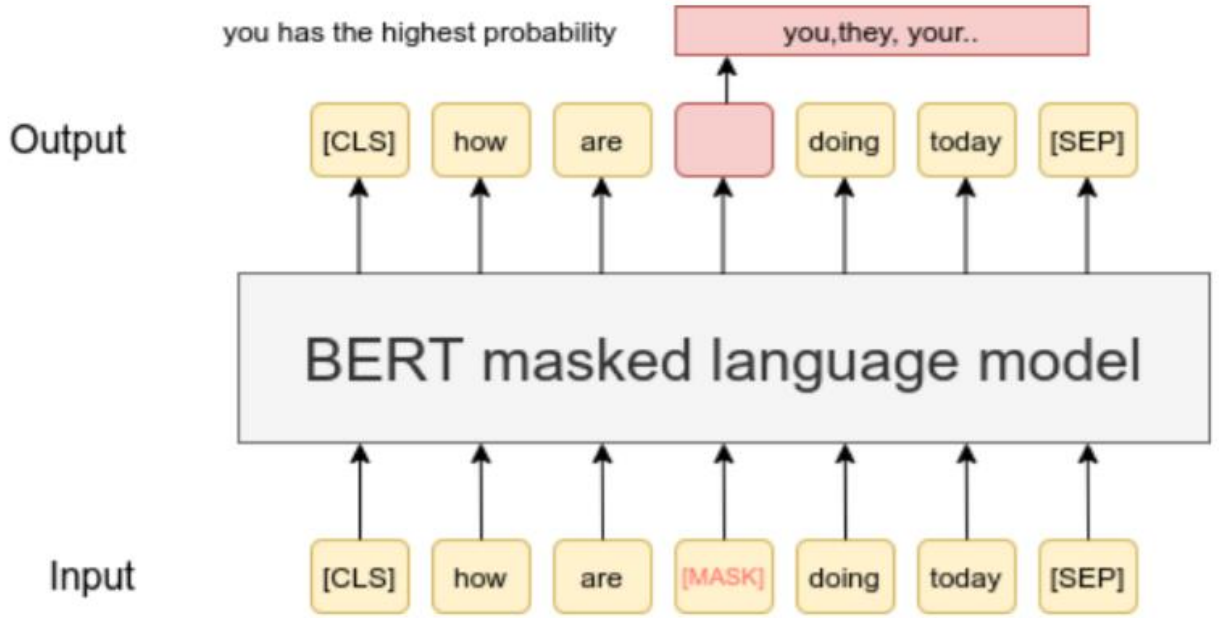
$$p(\text{"This is a fluffy dog."}) = p(w_1 w_2 w_3 w_4 w_5)$$

- Joint density is hard to model…
- Autoregressive LM: Sequential factorization
  $p(w_1 w_2 w_3 w_4 w_5) = p(w_1)p(w_2|w_1)p(w_3|w_1 w_2)p(w_4|w_1 w_2 w_3)p(w_5|w_1 w_2 w_3 w_4)$
  - Model next word probability given all previous words.
- Bi-gram model
  $p(w_1 w_2 w_3 w_4 w_5) \approx p(w_1)p(w_2|w_1)p(w_3|w_2)p(w_4|w_3)p(w_5|w_4)$
  - Assume next word only depends on the last word

Learning Word Vectors per *Language Modelling* BERT, GPT & CLIP

- *Self-supervised learning* of word representation
  - **Language modelling**: predict next word (GPT)
  - **Masked language modelling**: predict missing words (BERT)
  - **Contrastive Learning**: matching image and text. (CLIP)

Downstream Classifier can decode:
Part of speech, Sentiment, …



MLM — Sentence-Transformers documentation (sbert.net)

Why language modelling learns good word representation?

- The *meaning* of a word manifests in its *effect on its companion…*
  - If A and B are interchangeable in all context

$$p(w_1 w_2 A w_4 w_5) = p(w_1 w_2 B w_4 w_5) \quad \forall w_1, w_2, w_4, w_5$$

A and B have the same meaning.

  - Semantically similar words have similar effect on context, so they tend to have similar word vector.

Pretraining ⇒ fine-tuning paradigm

Vision Model Pretraining:

ImageNet classification

IMAGENET

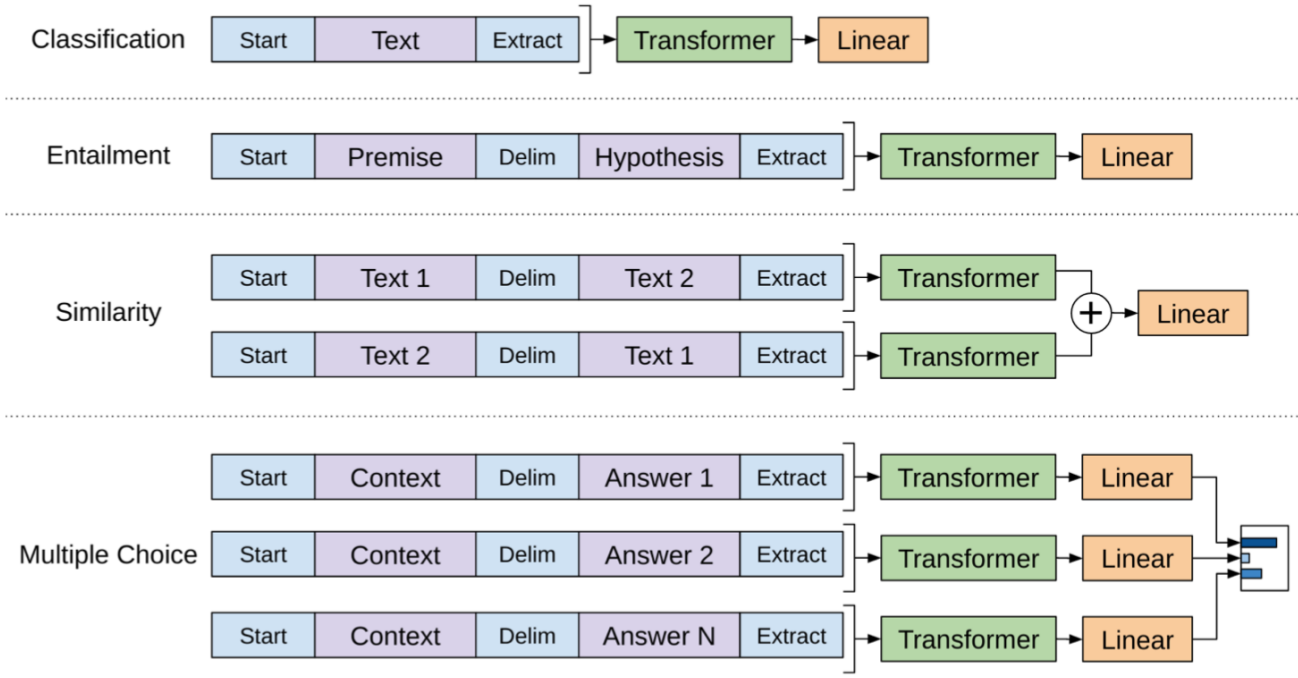
Downstream usage

Classification, segmentation etc.

Language Model Pretraining:

Wikipedia-text, Papers, Books…

Downstream usage



Interim Summary:

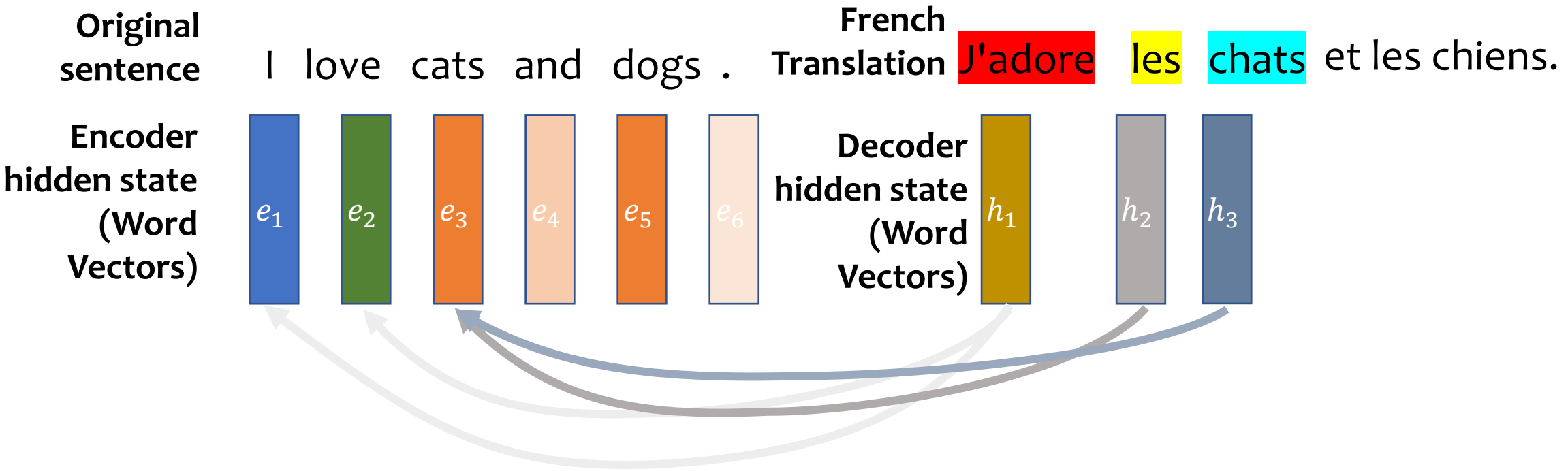
- **Final goal :**
  - A language model
  - A *good representation* of words, sentence, paragraph.

- **Model architecture**
  - RNN (LSTM, GRU), Transformer, etc.

- **Learning method**:
  - Self-supervised learning (SSL)
    - Language modelling: Predict missing words / next words
  - Multi-modal supervision.

Part III
Attention & Transformer

Origin of Attention:
Machine Translation (Seq2Seq)



- Use **Attention** to retrieve **relevant info** from a batch of vectors.

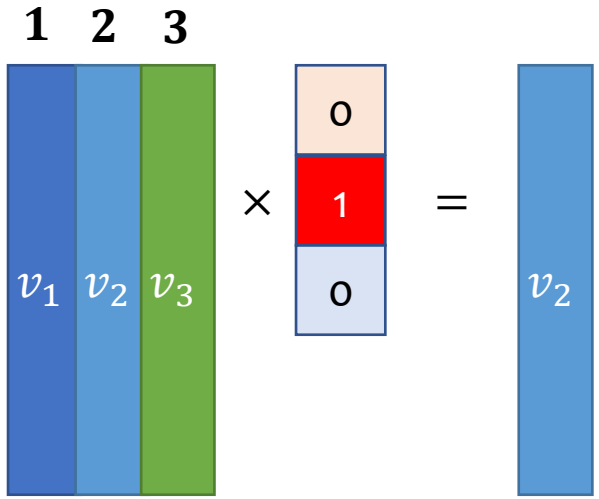
How to retrieve relevant information?

From dictionary to feature based attention.

From Dictionary to Attention

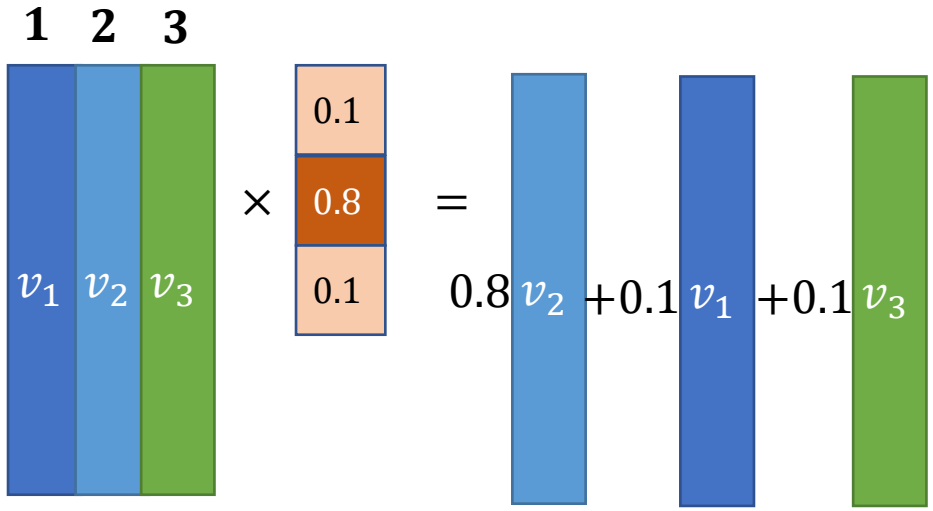
Dictionary: Hard-indexing

- Dictionary maps keys to values
  - `dic = $\{1 : v_1, 2 : v_2, 3 : v_3\}$`
  - Keys 1,2,3
  - Values v_1, v_2, v_3
- Query retrieves inforvation related to a key
  - `dic[2]` Query 2
  - Find 2 in keys
  - Get corresponding value.
- Retrieving values as matrix vector product
  - $\boldsymbol{a}$ is one hot vector over the keys
  - Weighted sum the value vectors.

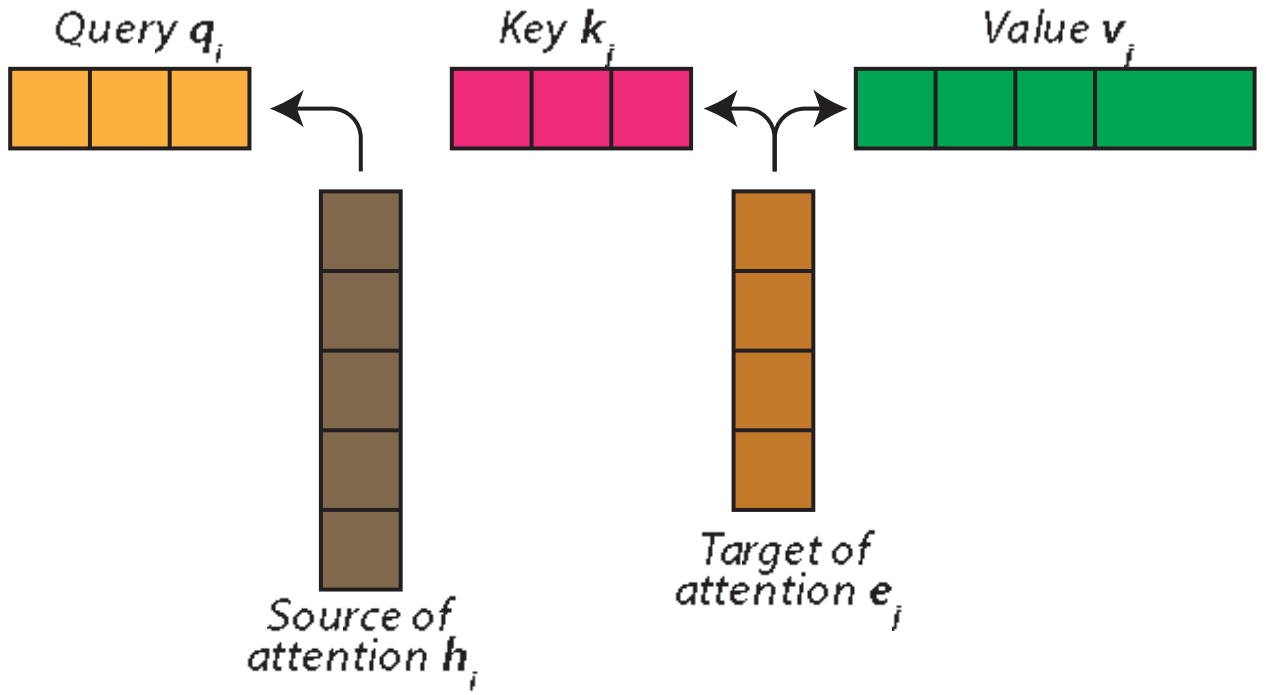


From Dictionary to Attention
Attention: Soft-indexing

- Soft indexing
  - $\boldsymbol{a}$ as a distribution over the keys, represents how much information I want from this key.
  - Matrix vector product, weighted combines the values.
- How to compute a?
  - *Feature based attention.*
  - based on similarity of query and key.



QKV attention

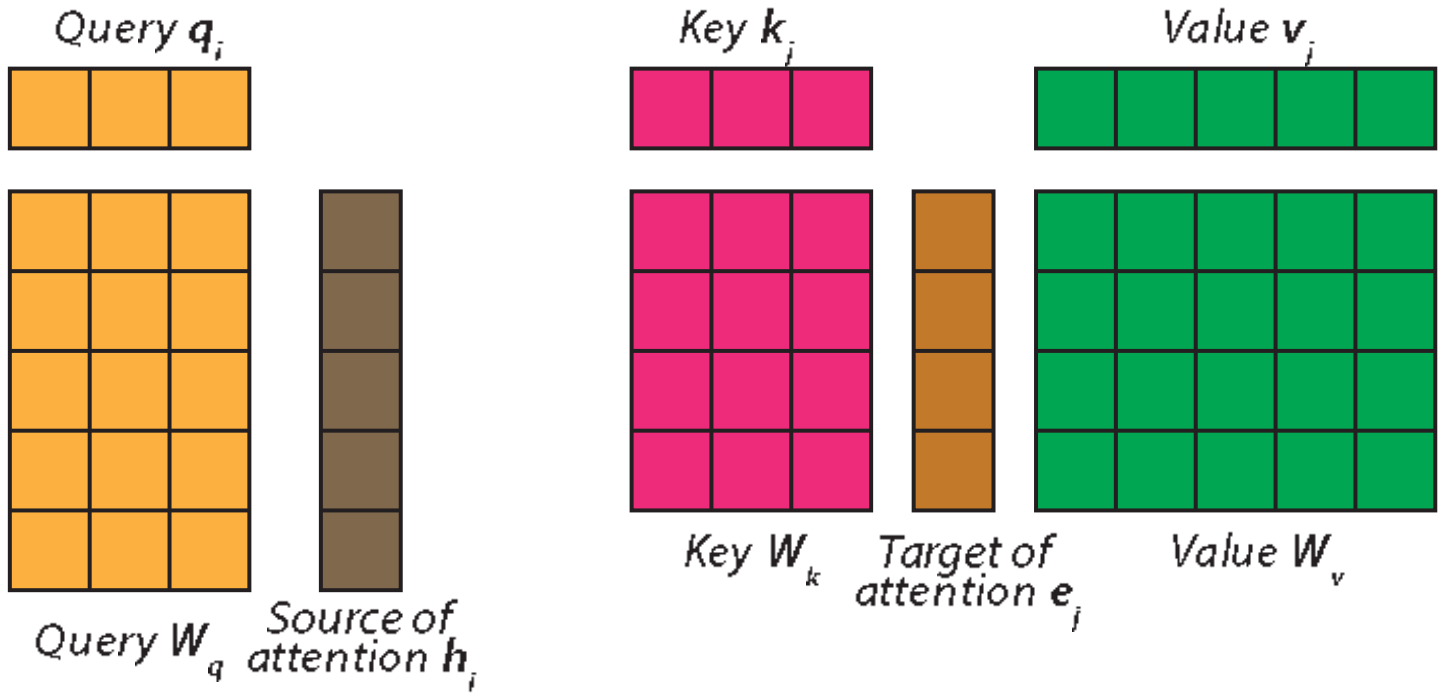


- $\boldsymbol{h}_i$ *vector of source of attention*
  - *(decoder hidden state, French)*
- $\boldsymbol{e}_j$ *vector of target of attention*
  - *(encoder hidden state, English)*

- **Q**uery : what source need
  - (*J'adore* : "I want subject pronoun & verb")
- **K**ey : what the target provide
  - (*I* : "Here is the subject")
- **V**alue : the information to be retrieved
  - (information related to *Je* or *J'*)

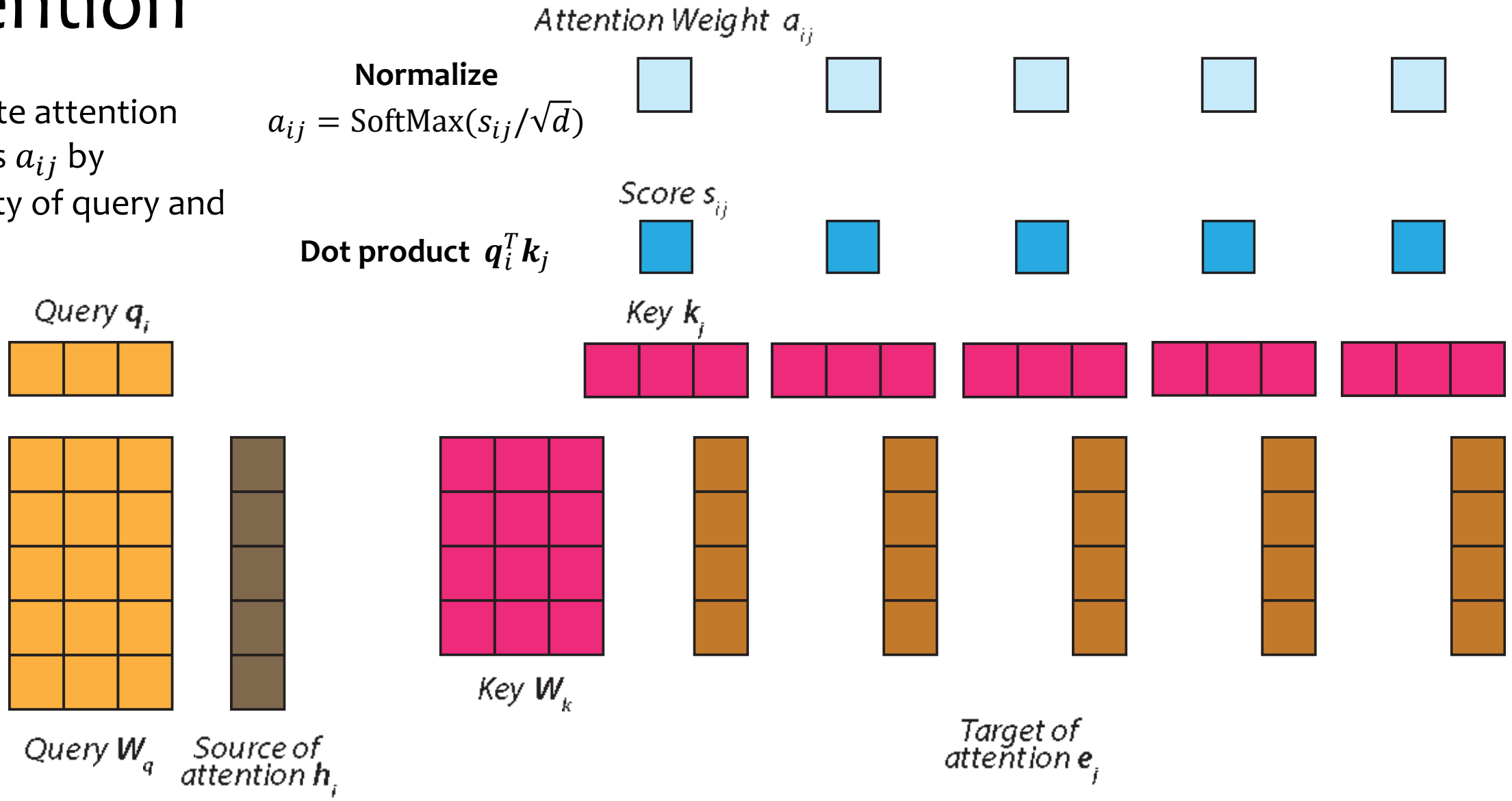
QKV attention

- QKV are linear projections of “word vector”
  - Query $q_i = W_q h_i$
  - Key $k_j = W_k e_j$
  - Value $v_j = W_v e_j$



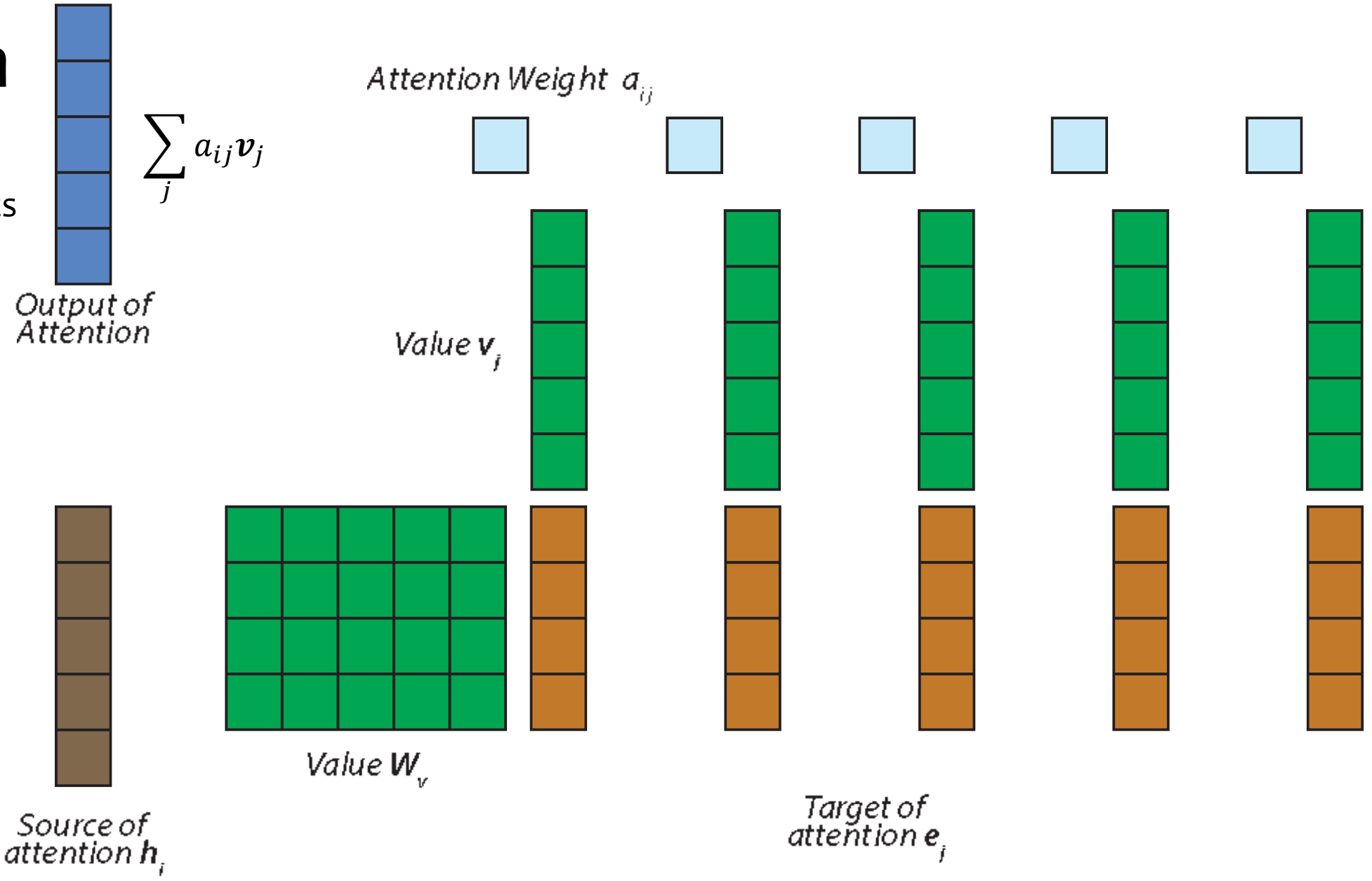
QKV attention

- Compute attention weights a_{ij} by similarity of query and key



QKV attention

- Use attention weights a_{ij} to weighted sum values $\boldsymbol{v}_j$



Summary: Attention mechanism

- Compute linear projection q, k, v
- Compute the inner product (similarity) of key k and query q
- SoftMax the normalized score as attention distribution.

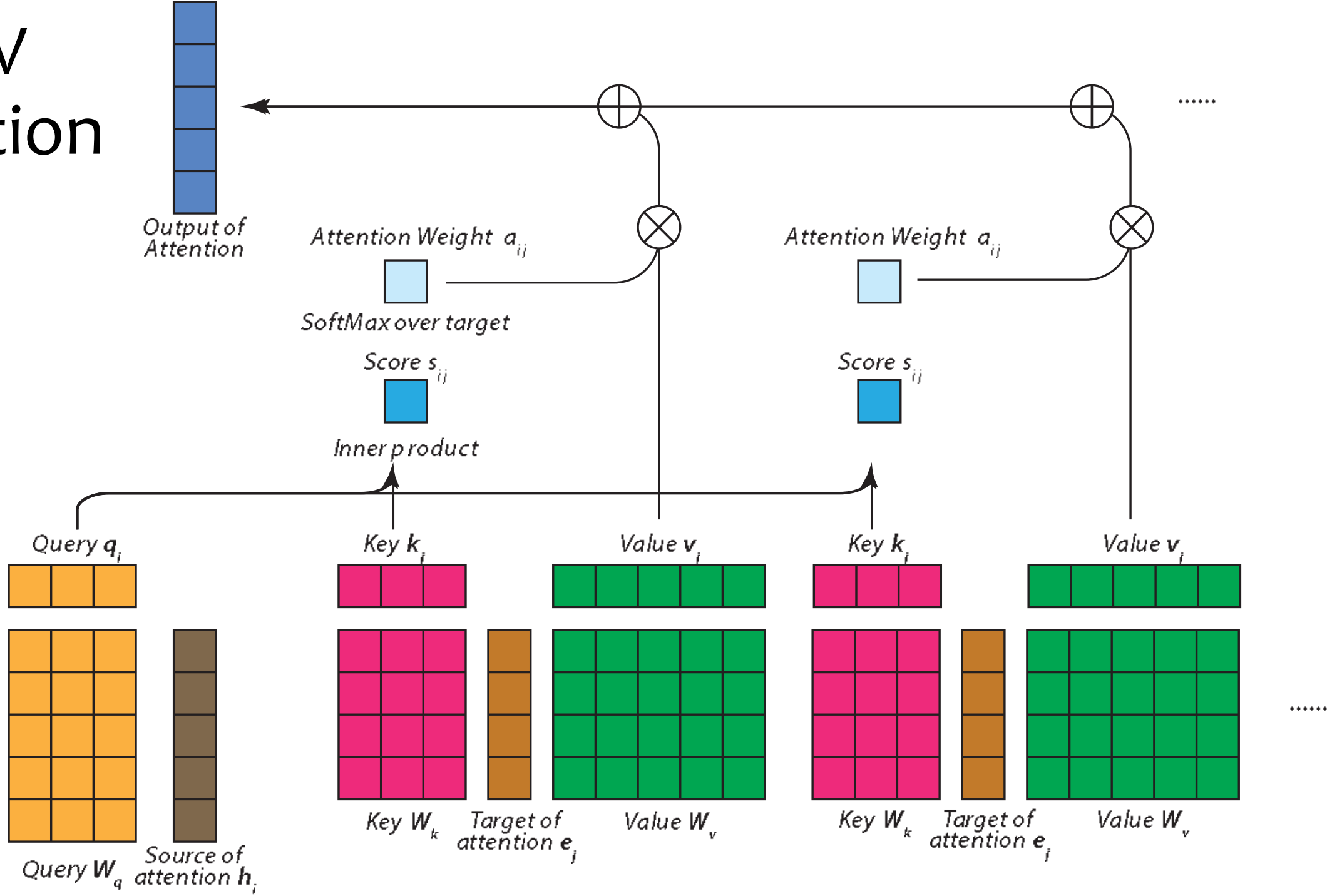
$$a_{ij} = \text{SoftMax}\left(\frac{k_j^T q_i}{\sqrt{len(q_i)}}\right), \sum_j a_{ij} = 1$$

- Use attention distribution to weighted average values v.

$$c_i = \sum_j a_{ij} v_j$$

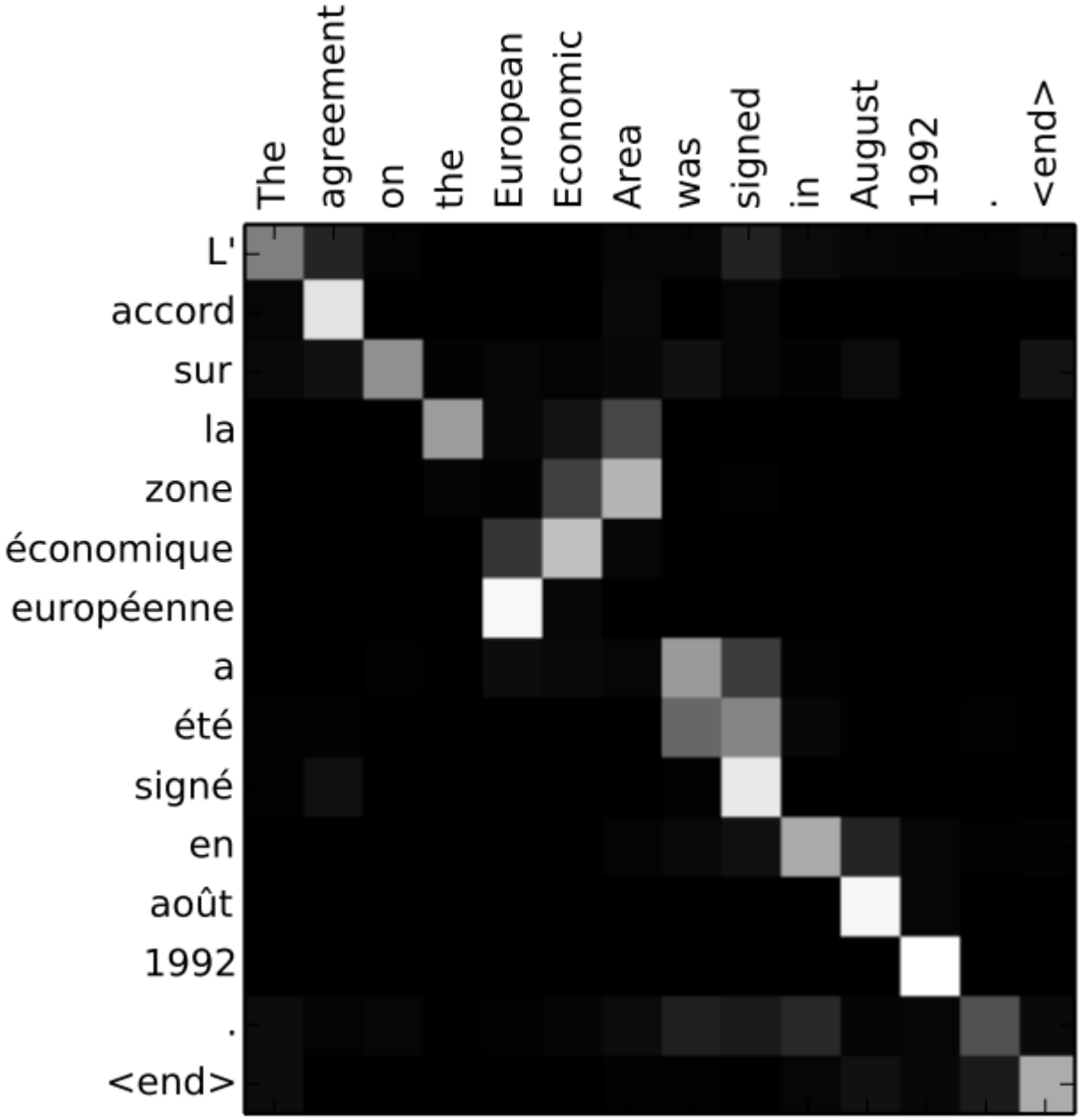
- Note:
  - Learnable weights: linear matrix W_q, W_k, W_v

QKV attention



Visualizing attention matrix $\boldsymbol{\alpha_{ij}}$ after training

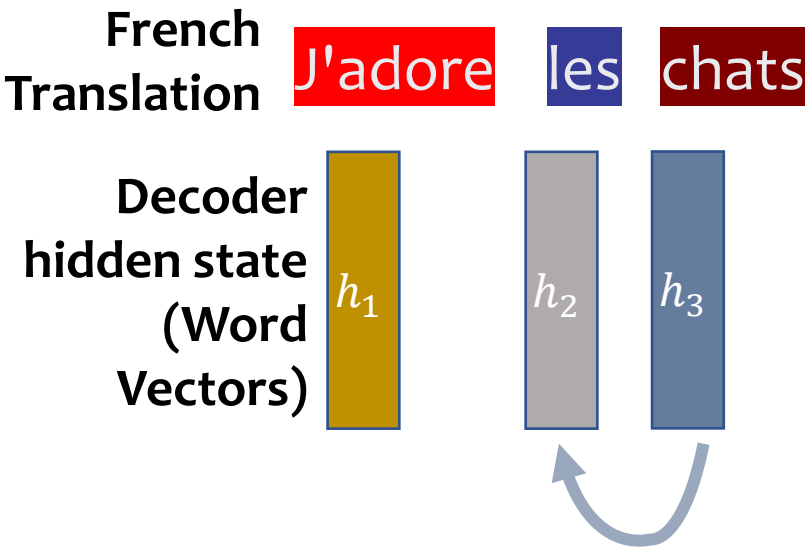
- *Learning to pay Attention*
- *French 2 English*
  - *"la zone economique europeenne" → "the European Economic Area"*
  - *"a ete signe" → "was signed"*



https://jalammar.github.io/visualizing-neural-machine-translation-mechanics-of-seq2seq-models-with-attention/

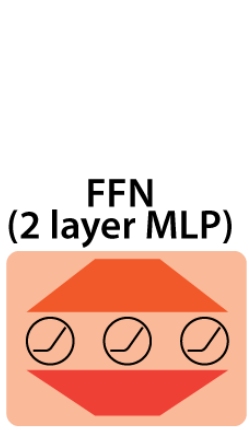
Cross & Self Attention

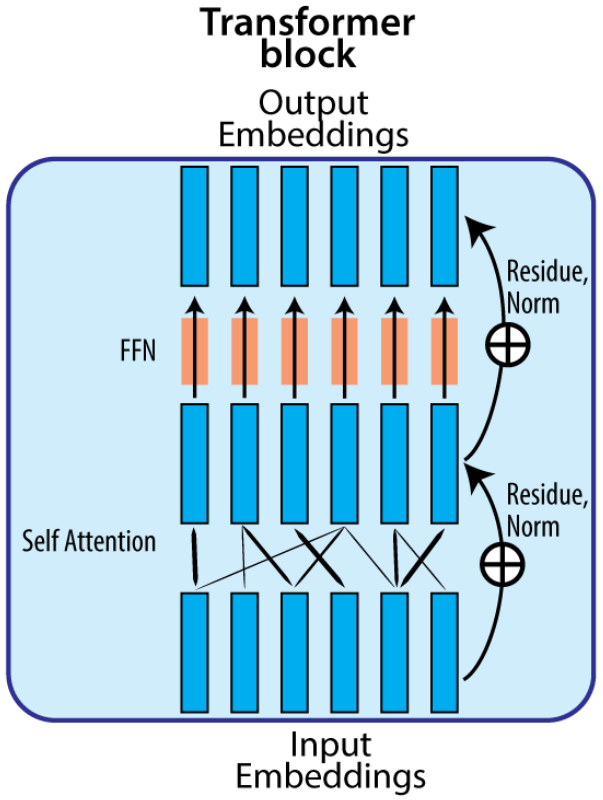
- Cross Attention
  - Source and target are different bunch of vectors (can have different length, number)
  - E.g. words in one language pay attention to words in **another**.
- Self Attention ($e_i = h_i$)
  - Source and target are the same bunch of vectors, with same length, number.



From Attention to Transformer

- A **transformer block** has
  - Self Attention
  - Feed forward network
    - E.g. a multi-layer perceptron with 1 hidden layer
  - Normalization (Layer normalization)
- A transformer model contains N x **transformer block**





Interim Summary: Attention

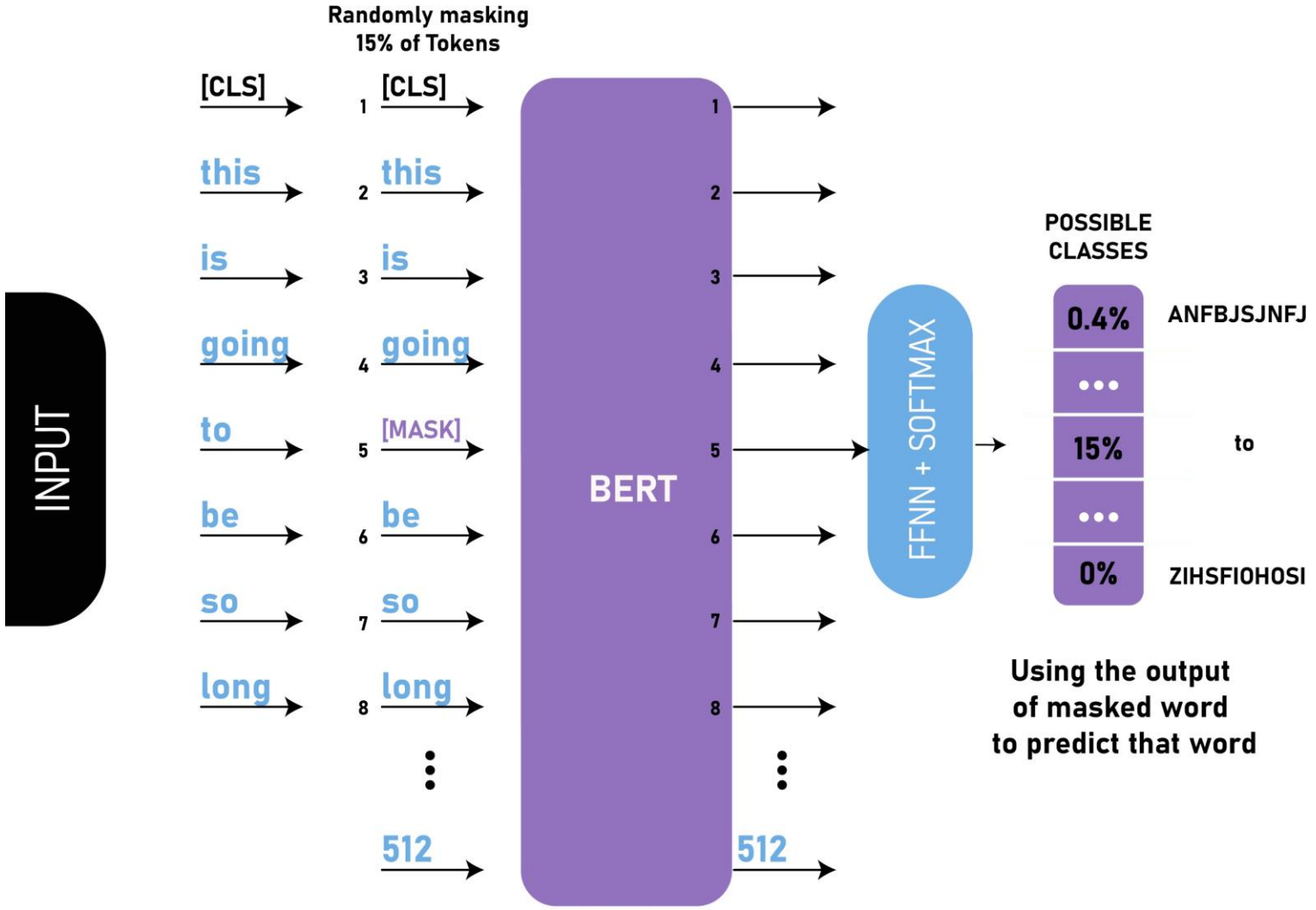
- Attention allows **dynamic routing** of information.
  - **Self attention** : Mixing information in one domain
  - **Cross attention** : Fetching information from one domain to the other
- Composition of Attention, MLP and Normalization forms Transformers

Part IV
Training language models

Given these models, how to train them?

Masked word prediction / Cloze task: BERT

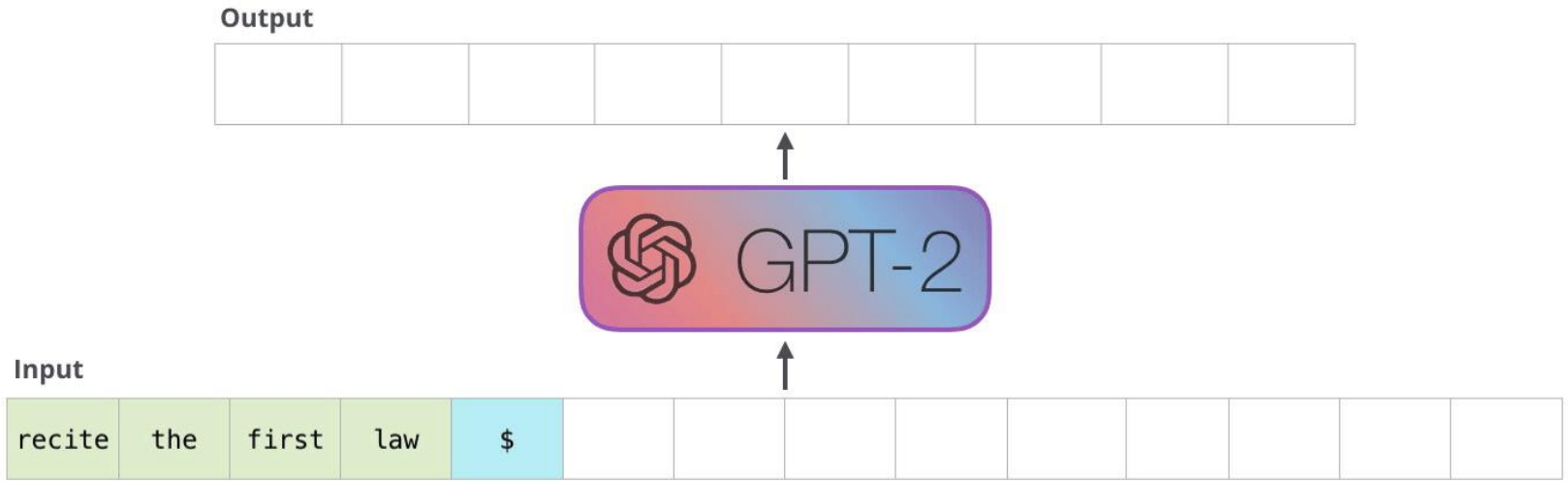
- Mask out 15% tokens
- Predict the mask token from context on both sides.



Next word prediction: GPT

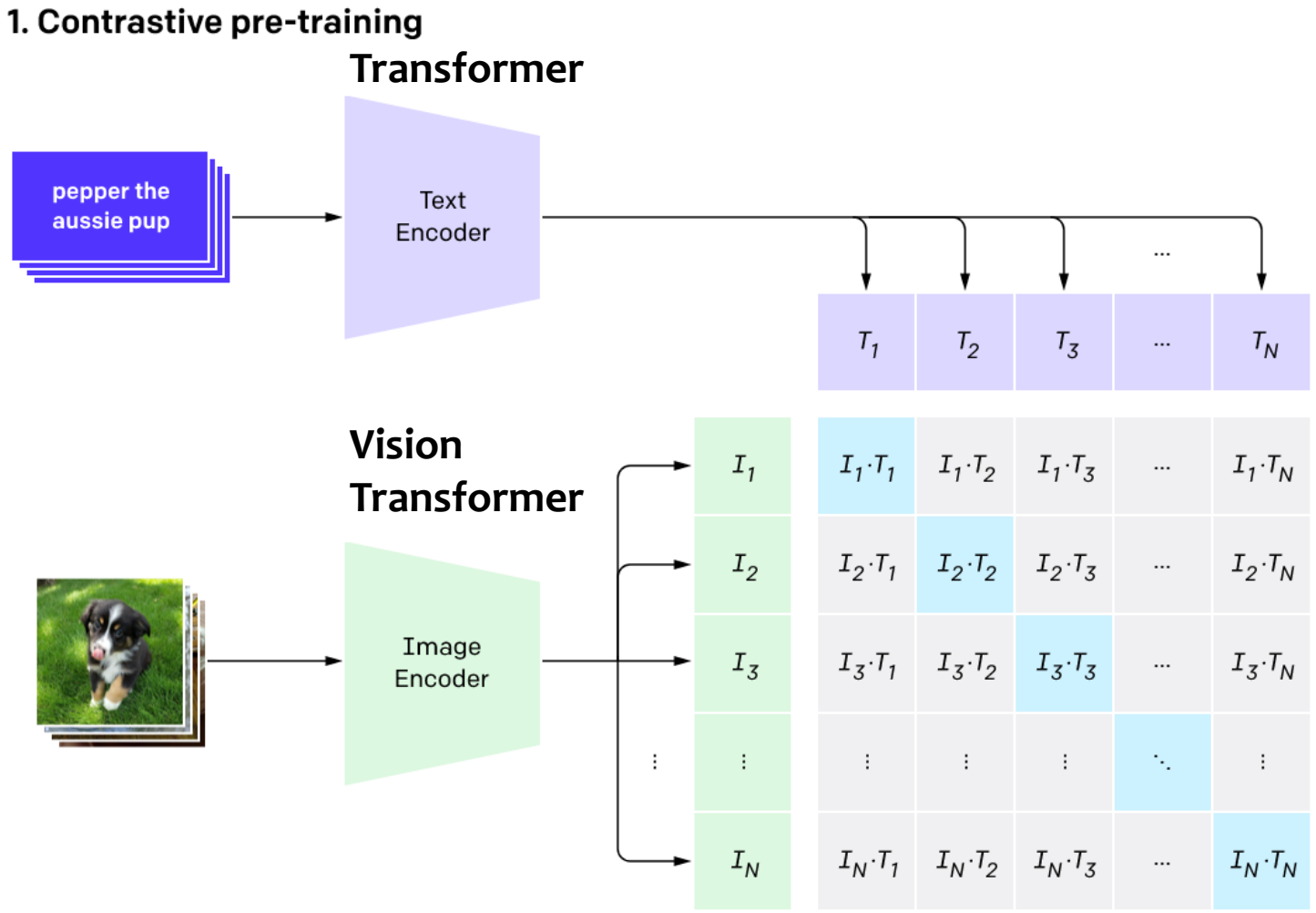
- Each word predict the next word considering all previous words
  - Enforcing causal attention.
  - Enabling text generation

- Autoregressive model, suitable for text generation.

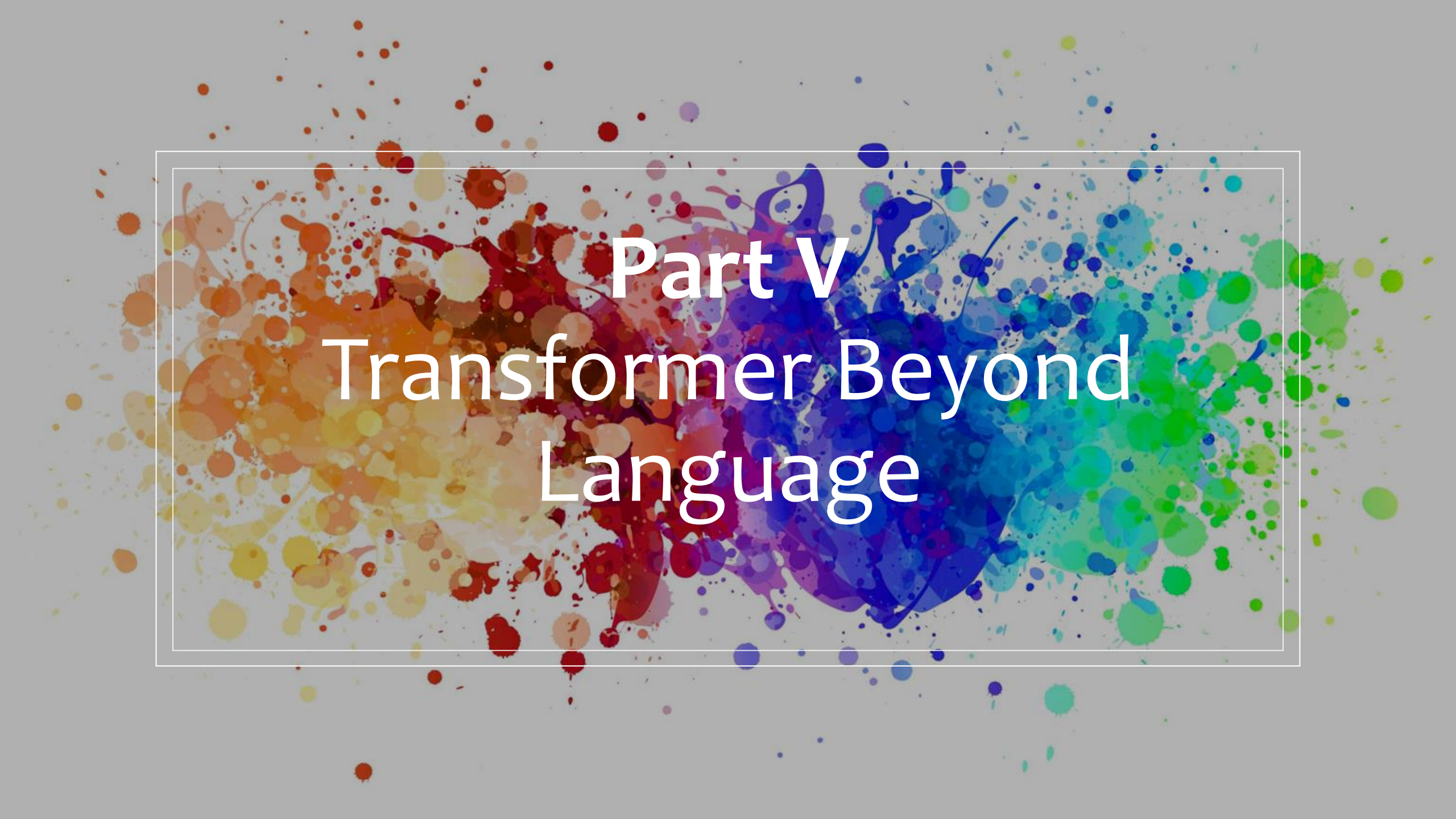


The Illustrated GPT-2

Multimodal representation learning: CLIP



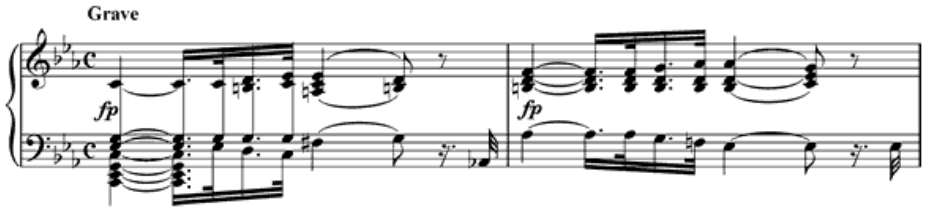
- Learn a joint encoding space for text caption and image
- Contrastive learning
  - Maximize representation similarity between paired image and caption.
  - Minimize representation similarity of other pairs


Part V
Transformer Beyond Language

Transformer is a powerful **sequence** processing tool…

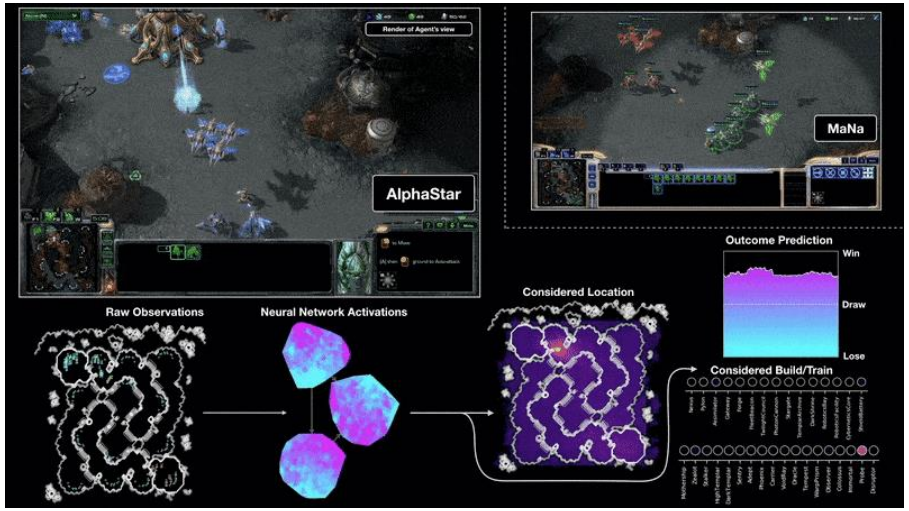
Everything can be formatted as sequence…

Audio Signal

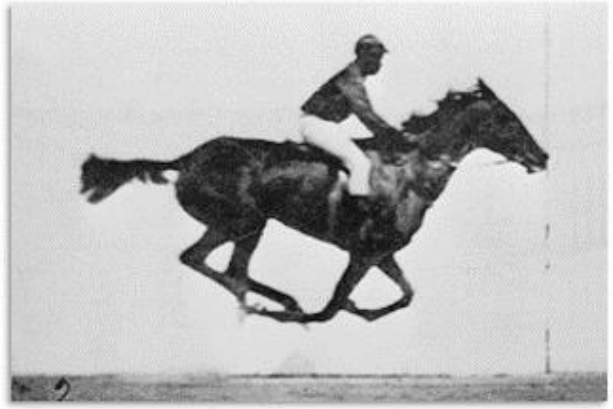


Music notes

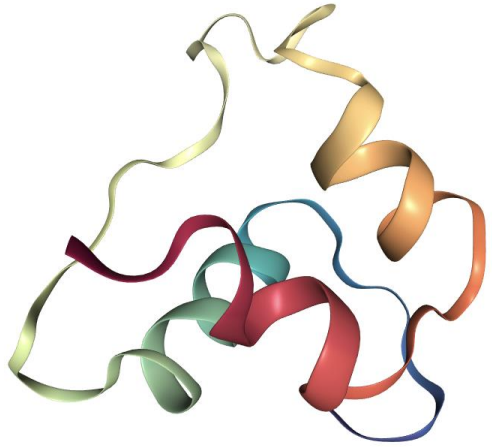
Image (as sequence of patches)



Action sequence in games



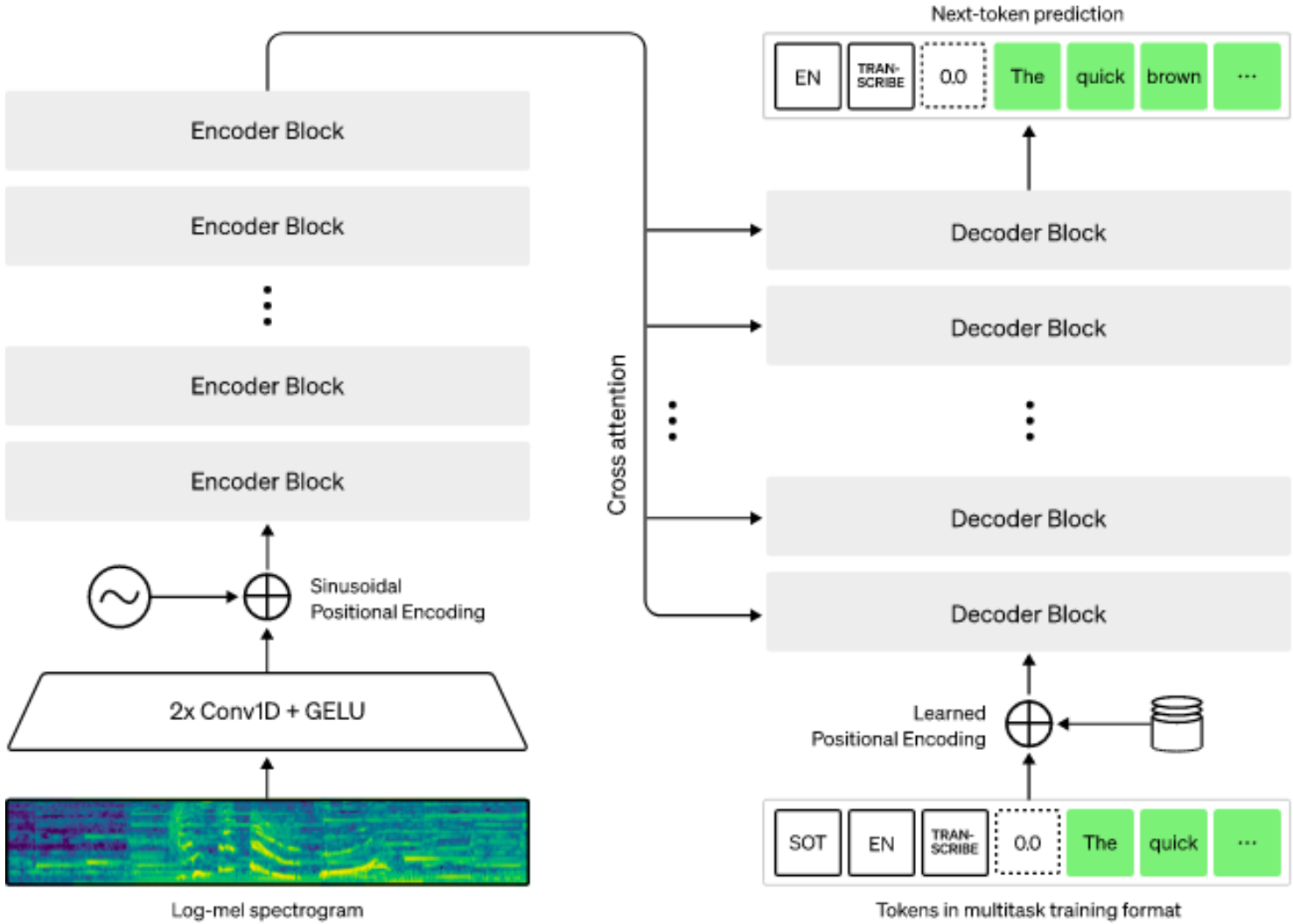
Video sequence



Protein sequence

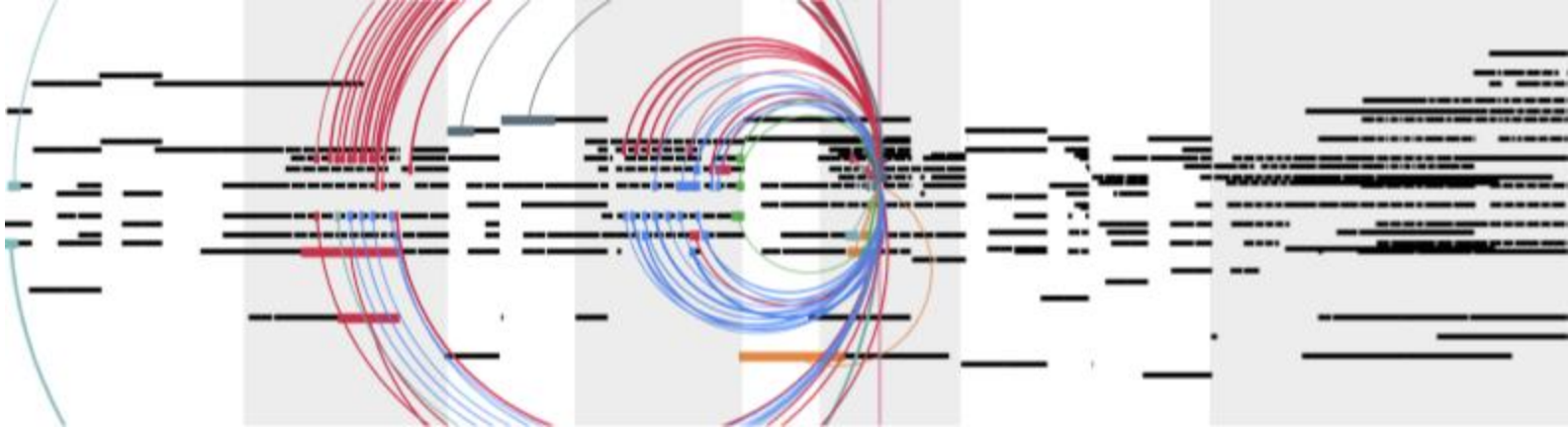
https://towardsdatascience.com/transformers-an-exciting-revolution-from-text-to-videos-dc70a15e617b

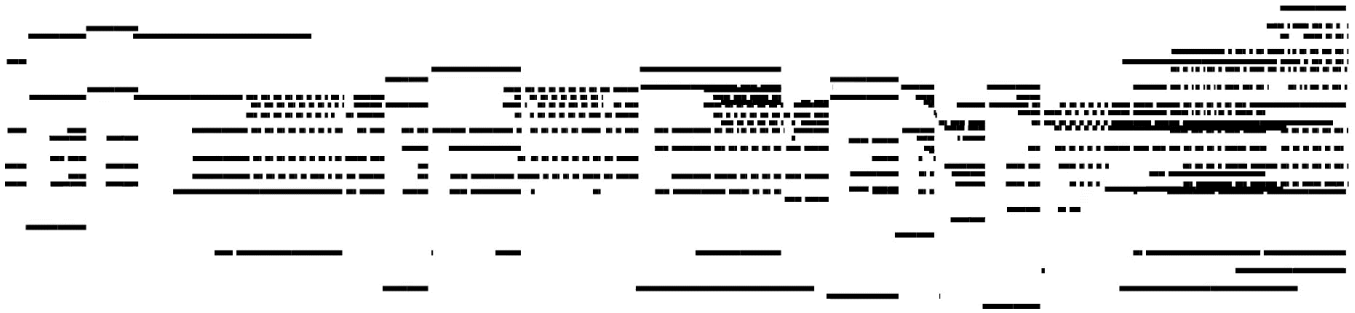
Audio transformer



OpenAI, Whisper, 2022
https://openai.com/research/whisper

Music transformer





Huang, Cheng-Zhi Anna, et al. "Music transformer." (2018).
https://magenta.tensorflow.org/music-transformer

Vision transformers

- Patches as tokens / words
- Self attention between patch tokens.
- Extra [CLASS] token used for classification.

https://towardsdatascience.com/transformers-an-exciting-revolution-from-text-to-videos-dc70a15e617b

Transformers for image generation

- Encoded patches discretized as tokens
- Transformer model the sequence of tokens
- Images decoded from the tokens.

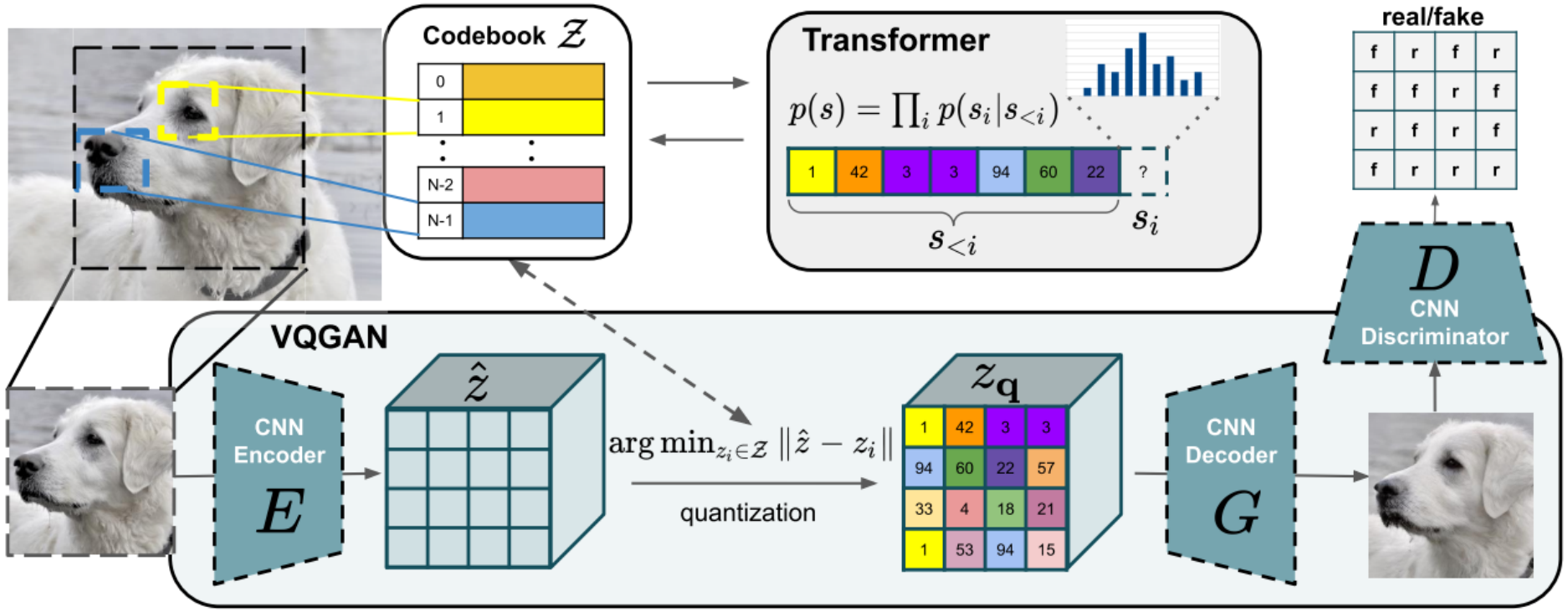
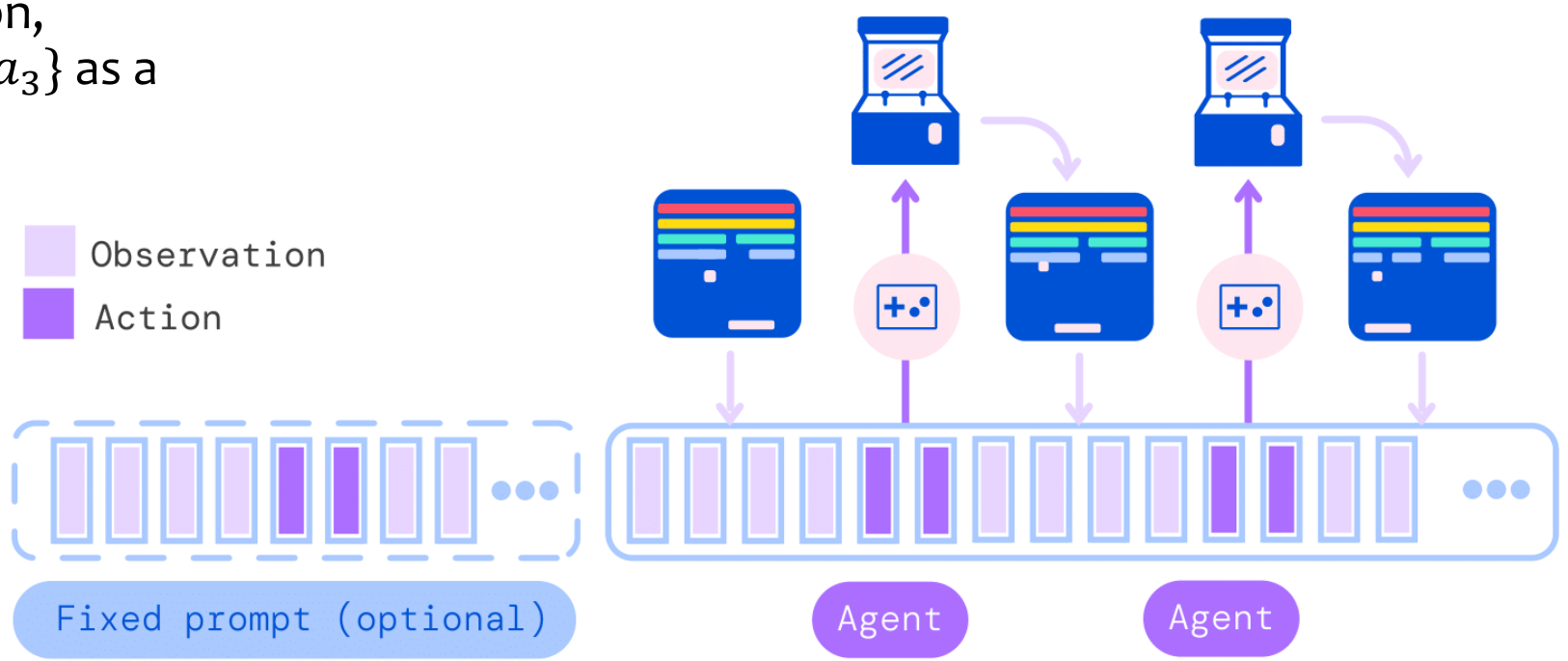


Figure 2. Our approach uses a convolutional *VQGAN* to learn a codebook of context-rich visual parts, whose composition is subsequently modeled with an autoregressive transformer architecture. A discrete codebook provides the interface between these architectures and a patch-based discriminator enables strong compression while retaining high perceptual quality. This method introduces the efficiency of convolutional approaches to transformer based high resolution image synthesis.

Esser, Patrick, Robin Rombach, and Bjorn Ommer. "Taming transformers for high-resolution image synthesis." *CVPR*. 2021.

Transformers for Control / Game

- Game playing / control as sequence completion
  - Interleaved sequence of action, observation $\{s_0, a_1, s_1, a_2, s_3, a_3\}$ as a sequence of tokens.
  - Predict next action from past observation and actions.



Gato, DeepMind, 2022

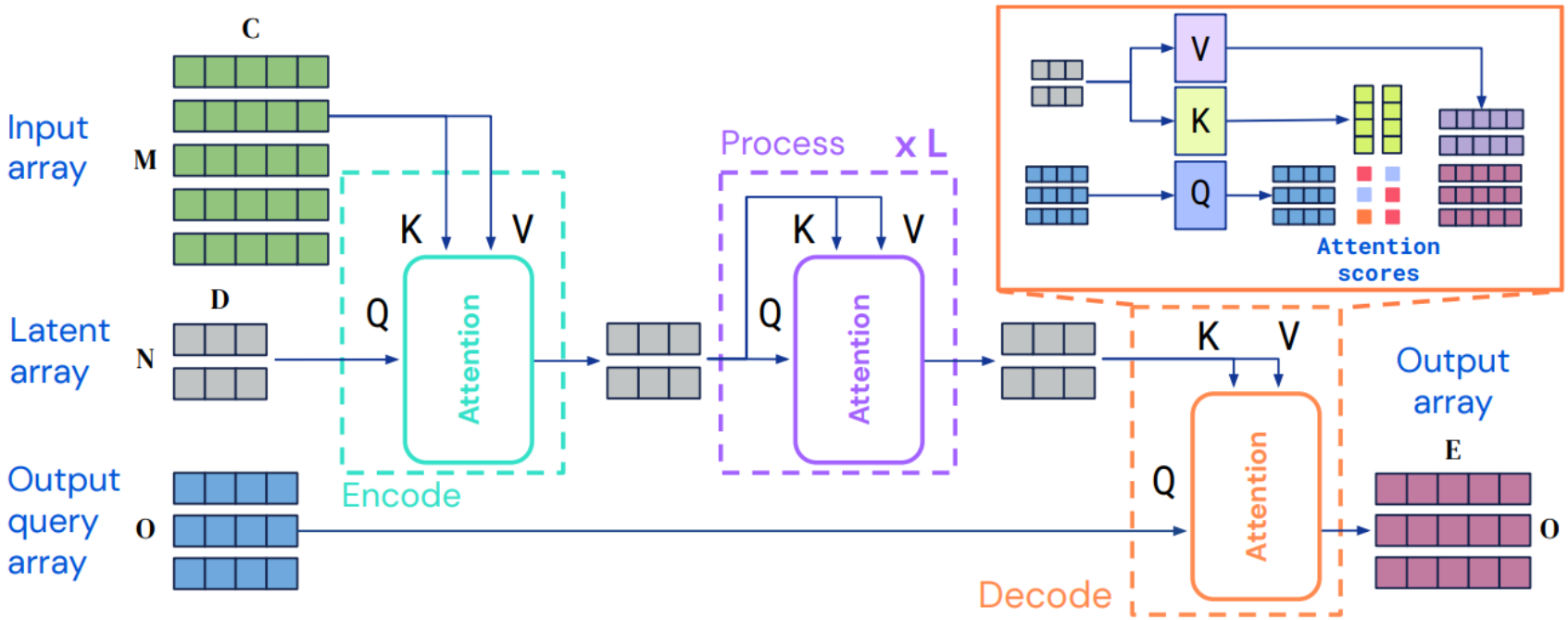
Attention provides a flexible mechanism to fuse between modalities.

Modality A & B can be

Image, audio, video, text, action sequence, protein sequence, neural activity time series, …

Perceiver / Perceiver IO:
Transformer for general data perception

- General data processing as long as data can be mapped into sequence of vectors
  - Use cross attention to fetch information from input
  - Self attention to process input.
  - Use cross attention to fetch relevant information and send to output.



Jaegle, Andrew, et al. "Perceiver: General perception with iterative attention." *ICML*, 2021.
Jaegle, Andrew, et al. "Perceiver io: A general architecture for structured inputs & outputs." *ICLR*, 2021

Transformers are powerful, but …

- Think about
  - How to represent your signal as a sequence of tokens (image -> sequence)
  - Use discrete tokens or continuous token vectors
  - $O(N^2)$ memory cost for long sequence
  - Special header / prompt for the sequence
  - Training objective.

Recap

- Goal of NLP, good representation.
- Attention mechanism & Transformers
- Self-supervised training objectives (e.g. BERT, GPT, CLIP)
- Pre-training and finetuning paradigm.
- Application of transformer beyond language. (e.g. Audio, Music, Vision, Game, Control etc.)

Thanks for your listening!