# Multimodal CL and NLP: Combining Language and Vision for (Computational) Semantics

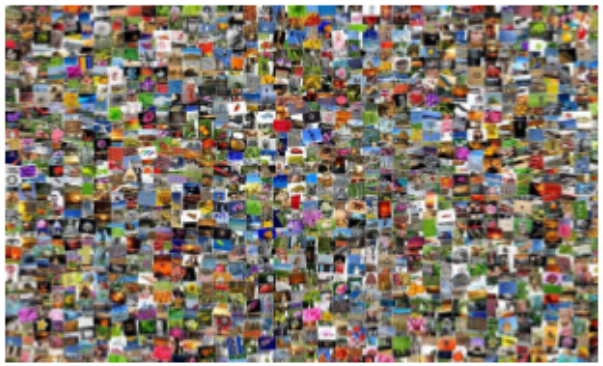
Carina Silberer

IMS, University of Stuttgart

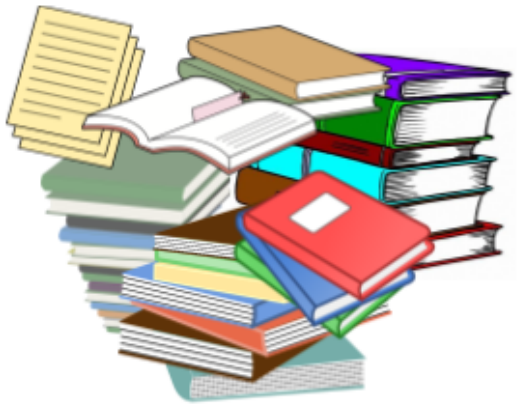
CL Fall School 2024, Passau
16 - 20 September 2024

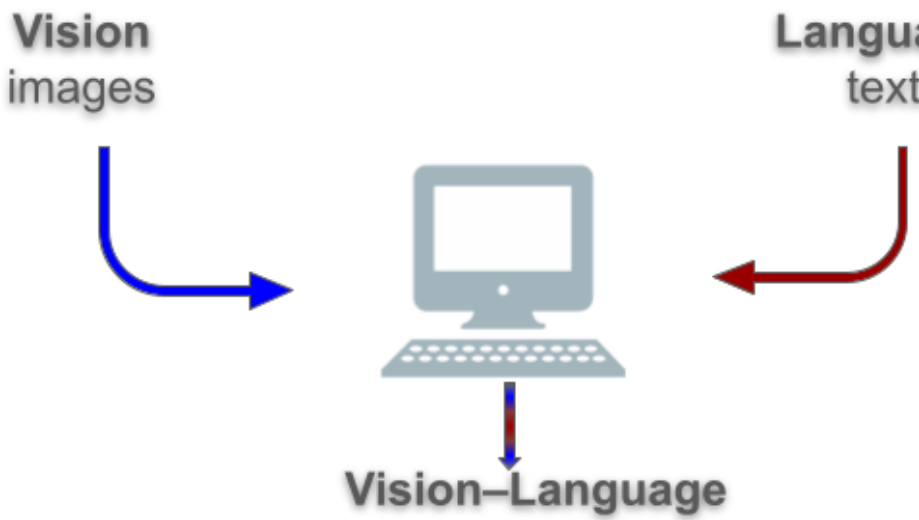
# Outline

# Multimodal (VL) Representations



https://pixabay.com/photos/photo-collection-pictures-photos-382018





# Outline

# Multimodal/Visually grounded representations

General idea: fuse visual and textual information about the same content/matter

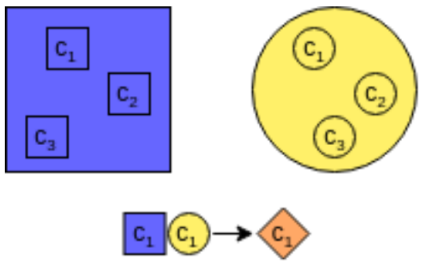
## Roadmap

1. For each input modality: Tokenise the input
   (e.g., (sub)words, image patches, regions-of-interests (objects))
2. Represent the input (tokens) by embedding it in an embedding space (vectors)
   (e.g., transformer vectors plus position embeddings, CNN features)
3. Combine the modalities

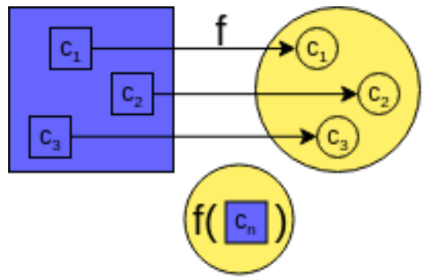
# Visual-Linguistic Semantics: Representation Models

## Visual-Linguistic Representations: Integration Strategies

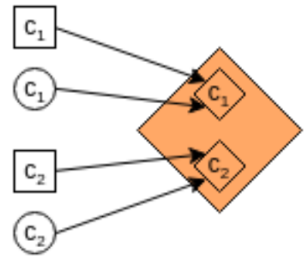
- Concatenation (a)
  Combine information; collapse shared information if dimensionality-reduction is applied
- Cross-modal Mapping (b)
  Exploit shared information
- Joint Learning (c)
  Exploit shared information, derive representations with shared and complementary features




(a) Multimodal fusion. Concatenate known representations from modality $A$ and $B$ and apply dimensionality reduction.




(b) Mapping. Learn a mapping function $f$ from modality $A$ to $B$ that can be applied on unknown concepts $c_n$.




(c) Joint learning. Optimize two objectives simultaneously: quality of unimodal representations and cross-modal alignment.

# Multimodal/Visually grounded representations

General idea: fuse visual and textual information about the same content/matter

## Modality Integration Methods

1. Early: concatenation, canonical correlation analysis, autoencoders, skipgram, etc.
2. Transformer-based, i.e., attention-based cross-modal interaction modeling between visual and textual representations from transformer and CNN feature embeddings
   – Fusion, e.g., VilBERT, LXMERT (fusion-encoder)
   – Alignment, e.g., CLIP (dual encoder / alignment)

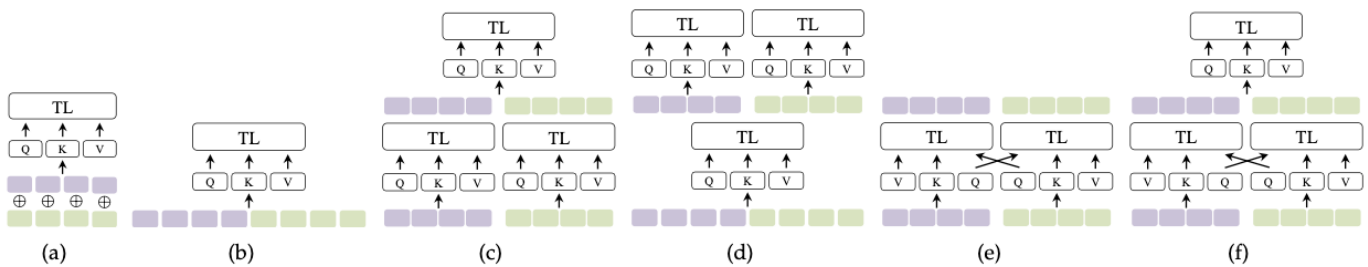



Fig. 2. Transformer-based cross-modal interactions: (a) Early Summation, (b) Early Concatenation, (c) Hierarchical Attention (multi-stream to one-stream), (d) Hierarchical Attention (one-stream to multi-stream), (e) Cross-Attention, and (f) Cross-Attention to Concatenation. "Q": Query embedding; "K": Key embedding; "V": Value embedding. "TL": Transformer Layer. Best viewed in colour.

# Multimodal (VL) Representations

## Early models

Visual representations
+
Textual representations

**Fusion techniques:**

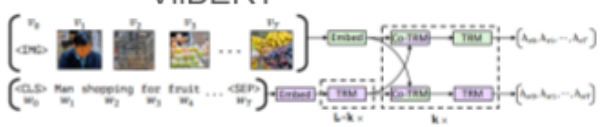
word2vec
autoencoders
CCA
concatenation
...

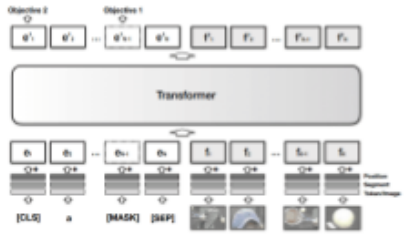
## Transformers

### BERT-like models

Visual encoder
+
Textual encoder

ViLBERT






VisualBERT

### Contrastive learning models

Visual encoder
+
Textual encoder

CLIP

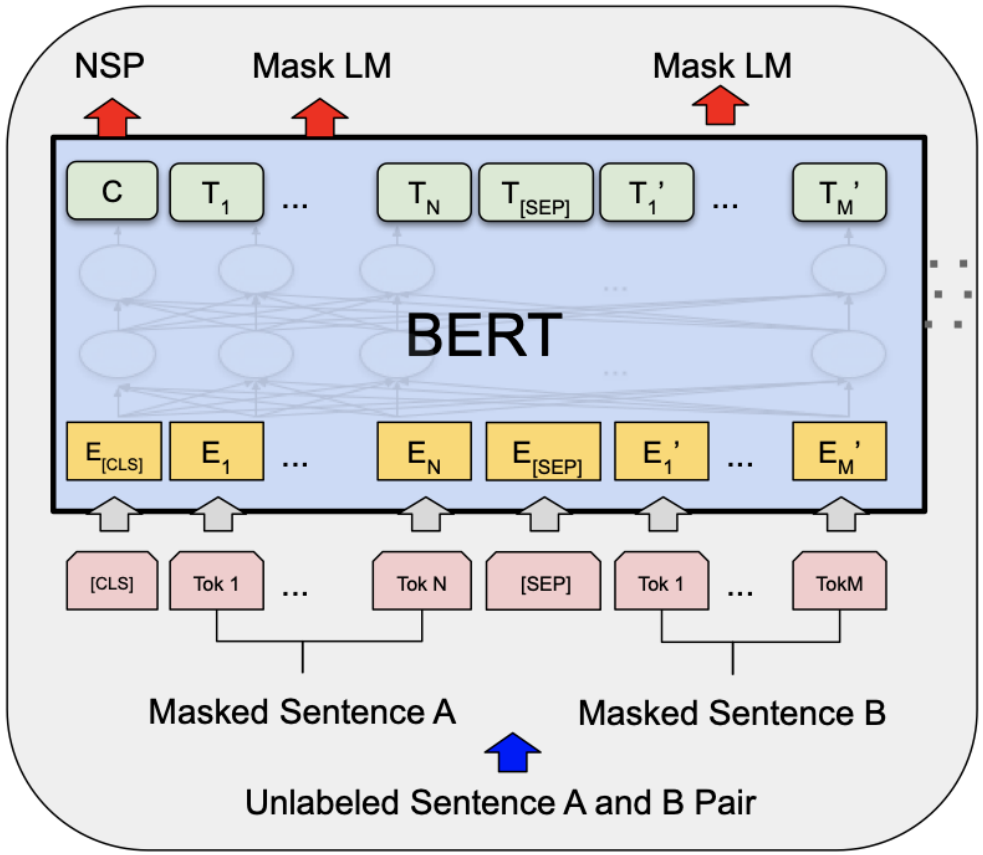
### Instruction-tuned and few-shot learning models

Flamingo
BLIP-2
BLIPInstruct
LLaVA

# Recall: Pretrained Transformer Models for Language

## BERT [Devlin et al., 2019]

Proxy Tasks: Next-Sentence Prediction (NSP) and Masked Language Modeling (MLM)




Pre-training

# Grounded Pretrained Transformer Models: Overview

## Single-Stream Architecture Models

- UNITER [Chen et al., 2020]
- VL-BERT [Su et al., 2020]
- VisualBERT [Li et al., 2019, Li et al., 2020]
- ViLT [Kim et al., 2021]

## Double-Stream Architecture Models

- ViLBERT [Lu et al., 2019, Lu et al., 2020]
- LXMERT [Tan and Bansal, 2019]

# Grounded Pre-Trained Transformer Models: Overview

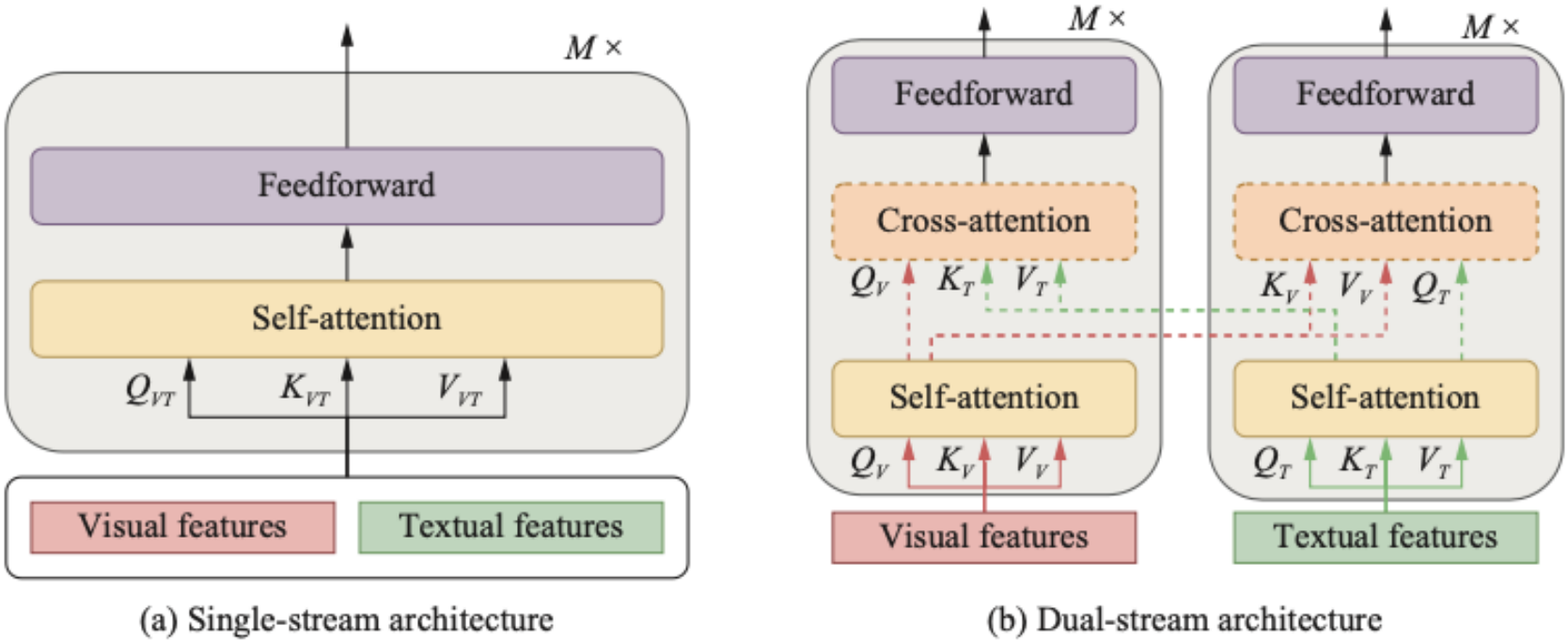
## Table from [Du et al., 2022]

| VL-PTM | Text encoder | Vision encoder | Fusion scheme | Pre-training tasks |
|---|---|---|---|---|
| **Fusion Encoder** | | | | |
| VisualBERT [2019] | BERT | Faster R-CNN | Single stream | MLM+ITM |
| Uniter [2020] | BERT | Faster R-CNN | Single stream | MLM+ITM+WRA+MRFR+MRC |
| OSCAR [2020c] | BERT | Faster R-CNN | Single stream | MLM+ITM |
| InterBert [2020] | BERT | Faster R-CNN | Single stream | MLM+MRC+ITM |
| ViLBERT [2019] | BERT | Faster R-CNN | Dual stream | MLM+MRC+ITM |
| LXMERT [2019] | BERT | Faster R-CNN | Dual stream | MLM+ITM+MRC+MRFR+VQA |
| VL-BERT[2019] | BERT | Faster R-CNN+ ResNet | Single stream | MLM+MRC |
| Pixel-BERT[2020] | BERT | ResNet | Single stream | MLM+ITM |
| Unified VLP[2020] | UniLM | Faster R-CNN | Single stream | MLM+seq2seq LM |
| UNIMO[2020b] | BERT, RoBERTa | Faster R-CNN | Single stream | MLM+seq2seqLM+MRC+MRFR+CMCL |
| SOHO[2021] | BERT | ResNet + Visual Dictionary | Single stream | MLM+MVM+ITM |
| VL-T5[2021] | T5, BART | Faster R-CNN | Single stream | MLM+VQA+ITM+Visual Grounding+Grounded Captioning |
| XGPT [2021] | Transformer | Faster R-CNN | Single stream | IC+MLM+DAE+MRFR |
| Visual Parsing[2021] | BERT | Faster R-CNN + Swin Transformer | Dual stream | MLM+ITM+MFR |
| ALBEF[2021a] | BERT | ViT | Dual stream | MLM+ITM+CMCL |
| SimVLM[2021b] | ViT | ViT | Single stream | PrefixLM |
| WenLan[2021] | RoBERTa | Faster R-CNN + EffcientNet | Dual stream | CMCL |
| ViLT[2021] | ViT | Linear Projection | Single stream | MLM+ITM |
| **Dual Encoder** | | | | |
| CLIP[2021] | GPT2 | ViT | | CMCL |
| ALIGN[2021] | BERT | EffcientNet | | CMCL |
| **Fusion Encoder+ Dual Encoder** | | | | |
| VLMo[2021a] | BERT | ViT | Single stream | MLM+ITM+CMCL |
| FLAVA[2021] | ViT | ViT | Single stream | MMM+ITM+CMCL |

Table 1: Glossary of Representative VL-PTMs. MLM/MVM: (Cross-Modal) Masked Language/Vision Modeling. ITM: Image-Text Matching. MRC: Masked Region Classification. MRFR: Masked Region Feature Regression. WRA: Word-Region Alignment. CMCL: Cross-Modal Contrastive Learning. DAE: Denoising AutoEncoding

# Grounded Pre-Trained Transformer Models: Overview

Table from [Du et al., 2022]




(a) Single-stream architecture (b) Dual-stream architecture

Fig. 1 Illustration of two types of model architectures for VLP

source: [Chen et al., 2023]

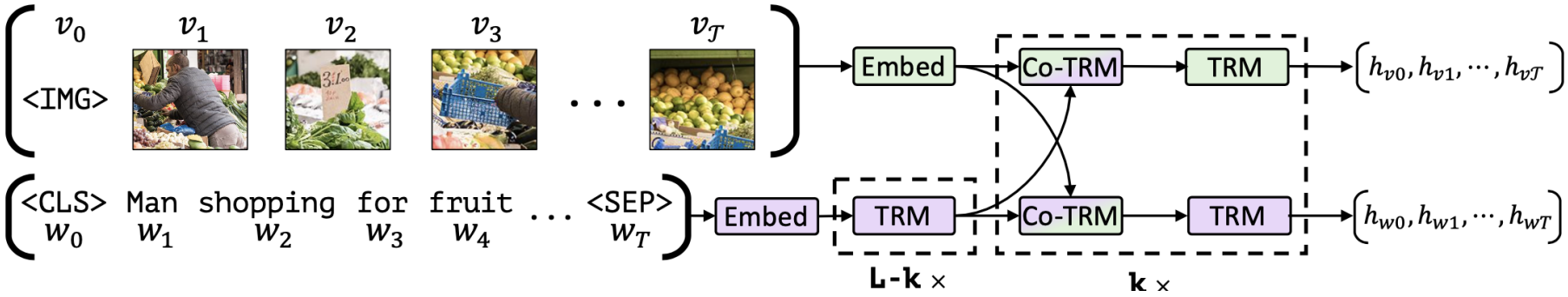
# Grounded Transformers: Two-Stream Models

# Grounded Transformers: Two-Stream Models

## ViLBERT [Lu et al., 2019, 2020]

### ViLBERT [Lu et al., 2019, Lu et al., 2020]

- Learn visual-linguistic representations from paired data
- Based on extending BERT to visual + textual input
- Separate streams for visual and textual input
- Modalities interact via **co-attentional transformer layers**



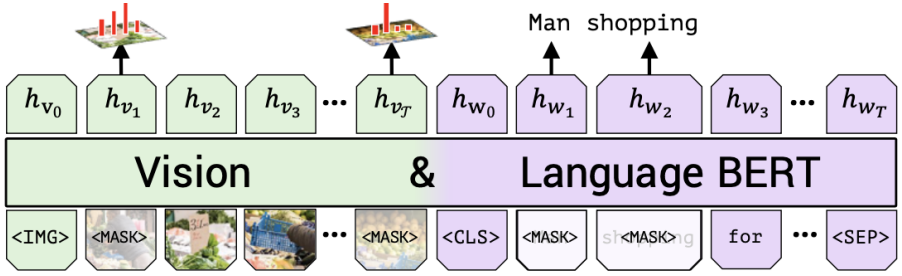
# Grounded Transformers: Two-Stream Models

## ViLBERT [Lu et al., 2019, 2020]

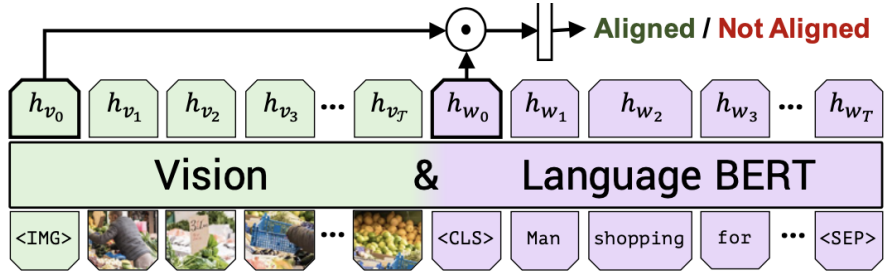
### Proxy Tasks for Pre-training

Given: Paired data (image–caption pairs)

- Masked Multi-modal Learning:
  Masked LM (as in standard BERT) and Masked Image Regions (distribution over semantic classes)
- Multi-modal Alignment:
  Does text describe visual content?




(a) Masked multi-modal learning




(b) Multi-modal alignment prediction

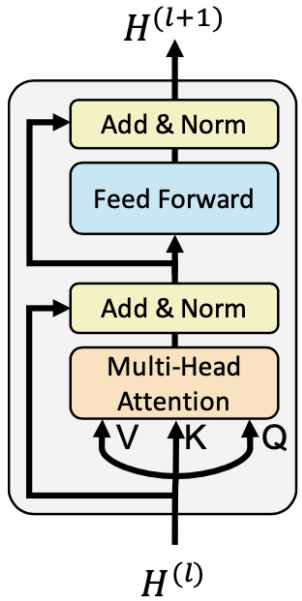
# Grounded Transformers: Two-Stream Models

## ViLBERT [Lu et al., 2019, 2020]

### Recall Textual Encoding: Query, Key, Value [Vaswani et al., 2017]

- Input (hidden embeddings) mapped to Query, Key, Value vectors
- **Query** (target): current focus
  **Keys** (context elements): to compute relevance to target (attention)
- **Values** (target and context elements): to compute contextualised output for the target

⇒ Output: Attention-weighted mean of value embeddings + input embedding (*residual connection*)




Q: *Which words to consider to contextually represent target "with"?*

"I saw the elephant with a telescope."

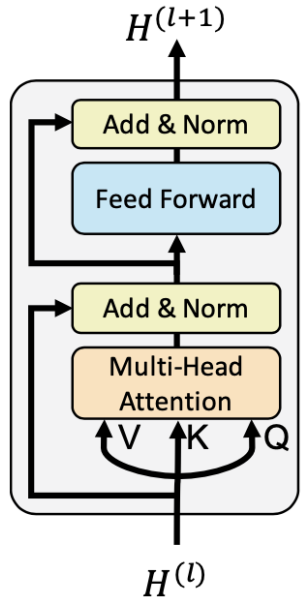
# Grounded Transformers: Two-Stream Models

## ViLBERT [Lu et al., 2019, 2020]

### Recall Textual Encoding: Query, Key, Value [Vaswani et al., 2017]

- Input (hidden embeddings) mapped to Query, Key, Value vectors
- **Query** (target): current focus
  **Keys** (context elements): to compute relevance to target (attention)
- **Values** (target and context elements): to compute contextualised output for the target

⇒ Output: Attention-weighted mean of value embeddings + input embedding (*residual connection*)




Q: *Which words to consider to contextually represent target "with"?*

*"I saw the elephant with a telescope."

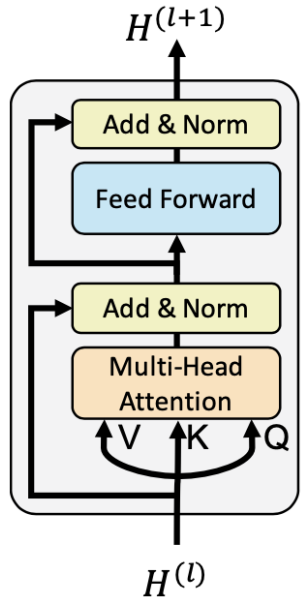
# Grounded Transformers: Two-Stream Models

## ViLBERT [Lu et al., 2019, 2020]

### Recall Textual Encoding: Query, Key, Value [Vaswani et al., 2017]

- Input (hidden embeddings) mapped to Query, Key, Value vectors
- **Query** (target): current focus
  **Keys** (context elements): to compute relevance to target (attention)
- **Values** (target and context elements): to compute contextualised output for the target

⇒ Output: Attention-weighted mean of value embeddings + input embedding (*residual connection*)




Q: *Which words to consider to contextually represent target "with"?*

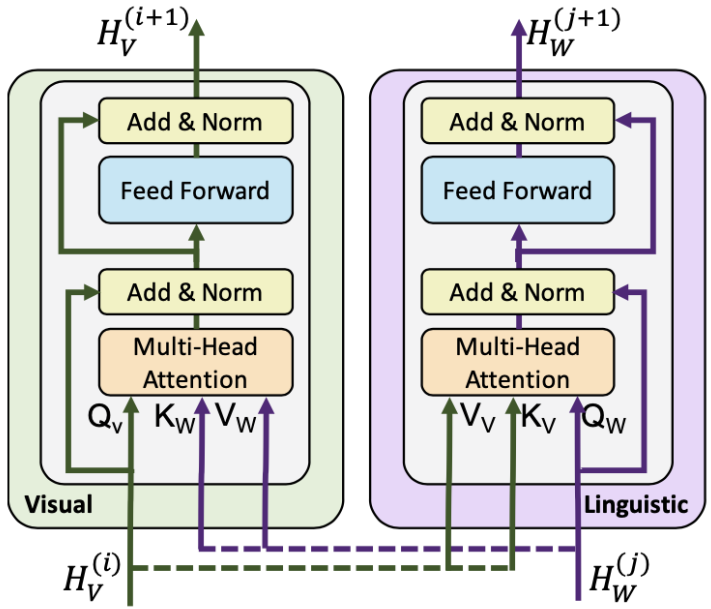
"I saw the elephant with a telescope."

# Grounded Transformers: Two-Stream Models

## ViLBERT [Lu et al., 2019, 2020]

### Modality Fusion: Co-Attention

- Query, Key and Value embeddings computed for visual input, $Q_V$, $K_V$, $V_V$, and for textual input, $Q_W$, $K_W$, $V_W$, resp.

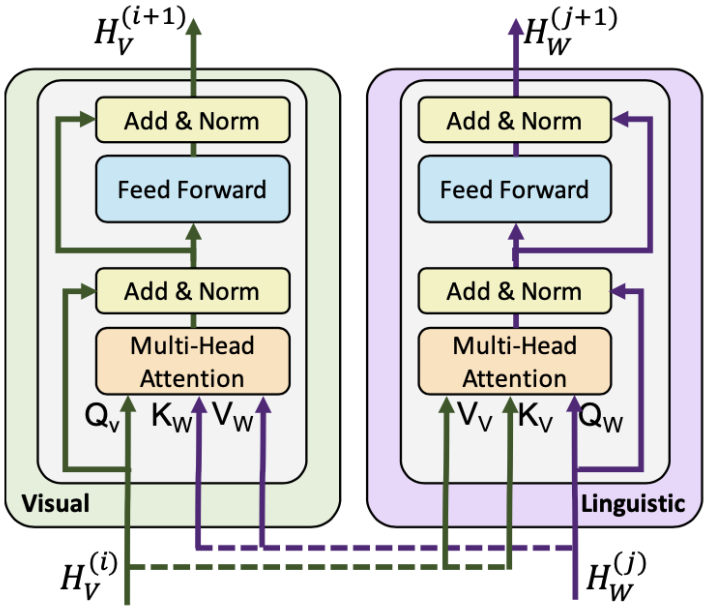


# Grounded Transformers: Two-Stream Models

## ViLBERT [Lu et al., 2019, 2020]

### Modality Fusion: Co-Attention

- Query, Key and Value embeddings computed for visual input, $Q_V$, $K_V$, $V_V$, and for textual input, $Q_W$, $K_W$, $V_W$, resp.
- Textual attention head: *Given word, to which image regions does it relate, and how strong is relation?*
  Computes attention-weighted mean of *visual* values using $Q_W$, $K_V$, $V_V$.




dreamstime.com

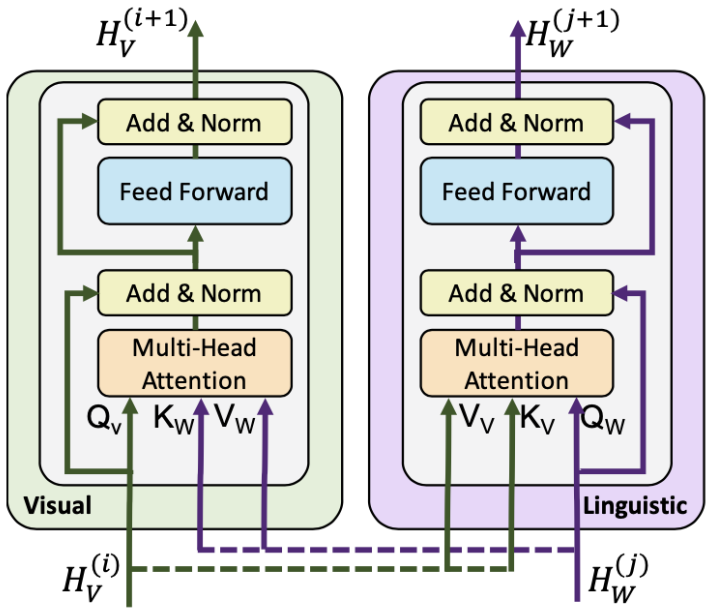
"I saw the elephant with a telescope."

# Grounded Transformers: Two-Stream Models

## ViLBERT [Lu et al., 2019, 2020]

### Modality Fusion: Co-Attention

- Query, Key and Value embeddings computed for visual input, $Q_V$, $K_V$, $V_V$, and for textual input, $Q_W$, $K_W$, $V_W$, resp.
- Textual attention head: Uses $Q_W$, $K_V$, $V_V$




dreamstime.com

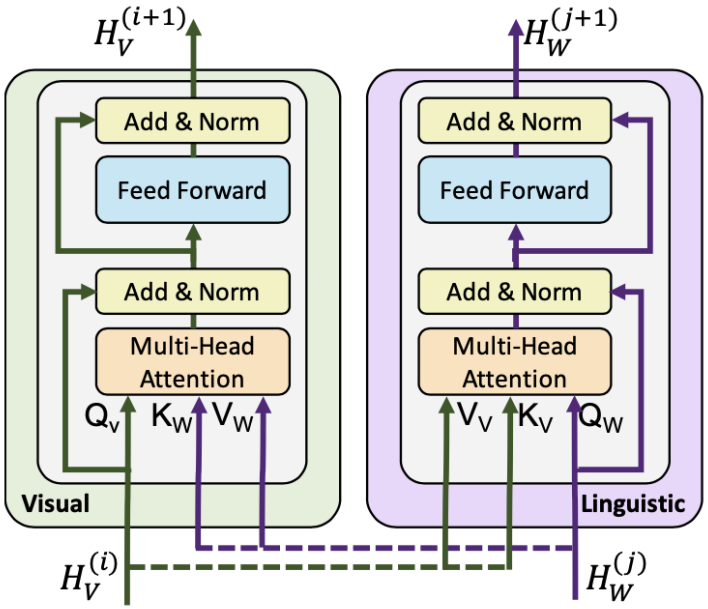
"I saw the elephant with a telescope."

# Grounded Transformers: Two-Stream Models

## ViLBERT [Lu et al., 2019, 2020]

### Modality Fusion: Co-Attention

- Query, Key and Value embeddings computed for visual input, $Q_V$, $K_V$, $V_V$, and for textual input, $Q_W$, $K_W$, $V_W$, resp.
- Textual attention head: Uses $Q_W$, $K_V$, $V_V$
- Visual attention head: Uses $Q_V$, $K_W$, $V_W$ *Given image region, what are the words that relate to the shown content, and how strong is relation?*




dreamstime.com

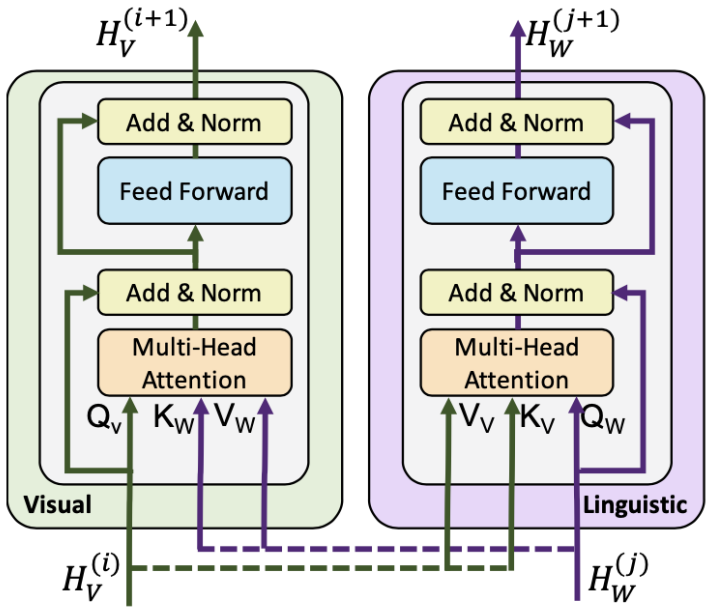
"I saw the elephant with a telescope."

# Grounded Transformers: Two-Stream Models

## ViLBERT [Lu et al., 2019, 2020]

### Modality Fusion: Co-Attention

- Query, Key and Value embeddings computed for visual input, $Q_V$, $K_V$, $V_V$, and for textual input, $Q_W$, $K_W$, $V_W$, resp.
- Textual attention head: Uses $Q_W$, $K_V$, $V_V$
- Visual attention head: Uses $Q_V$, $K_W$, $V_W$
  $\Rightarrow$ attention-weighted mean of *textual* values: $Q_V$, $K_W$, $V_W$.




dreamstime.com

"I saw the elephant with a telescope."

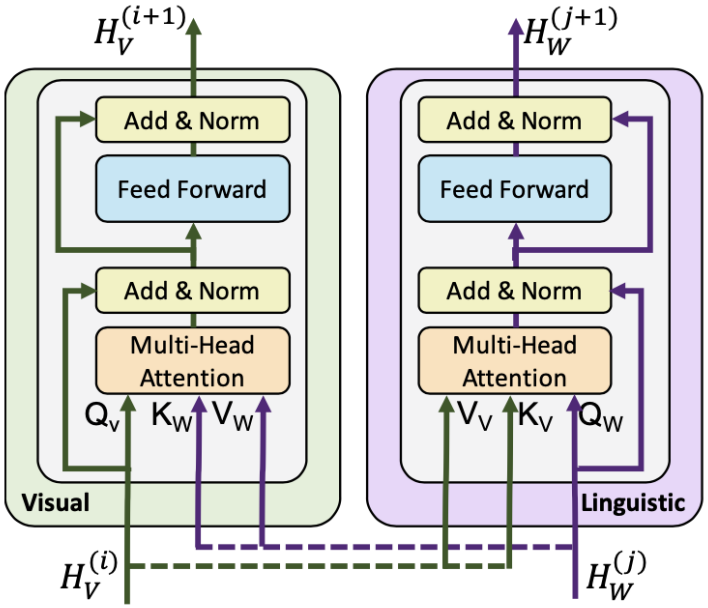
# Grounded Transformers: Two-Stream Models

## ViLBERT [Lu et al., 2019, 2020]

### Modality Fusion: Co-Attention

- Query, Key and Value embeddings computed for visual input, $Q_V$, $K_V$, $V_V$, and for textual input, $Q_W$, $K_W$, $V_W$, resp.
- Textual attention head: Uses $Q_W$, $K_V$, $V_V$
- Visual attention head: Uses $Q_V$, $K_W$, $V_W$

$\Rightarrow$ Output: Analogously, weighted mean of values + target




"I saw the elephant with a telescope."

# Grounded Transformers: Datasets

## Input Representations

- Textual: As in standard BERT (word, position, segment embeddings)
- Visual: Each image region (rectangular *boundig box*) is represented through sum of (i) and (ii)
  - (i) feature vector, extracted from pre-trained object detector (CNN-based)
  - (ii) position and size encoding of bounding box

# Grounded Transformers: Datasets

## Datasets used to pre-train models on proxy tasks: ViLBERT

- **Conceptual Captions** [Sharma et al., 2018, Ng et al., 2020]
  Automatically processed alt-text into clean image captions

**Alt-text**: A Pakistani worker helps to clear the debris from the Taj Mahal Hotel November 7, 2005 in Balakot, Pakistan.

**Conceptual Captions**: a worker helps to clear the debris.

**Alt-text**: Musician Justin Timberlake performs at the 2017 Pilgrimage Music & Cultural Festival on September 23, 2017 in Franklin, Tennessee.

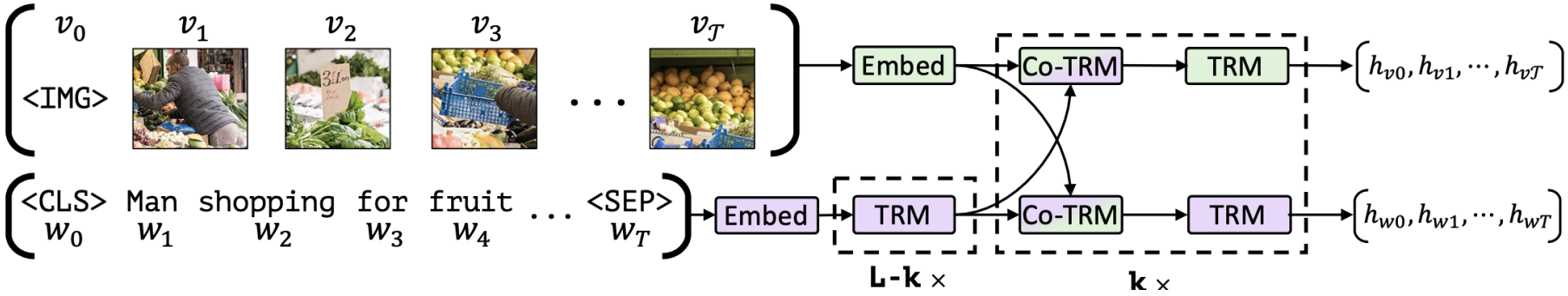
**Conceptual Captions**: pop artist performs at the festival in a city.

# Grounded Transformers: Two-Stream Models

## ViLBERT [Lu et al., 2019, 2020]

### ViLBERT: Summary [Lu et al., 2019, Lu et al., 2020]

- Separate streams for visual and textual input
- Modalities interact via **co-attentional transformer layers**
- Proxy tasks for pre-training:
  (i) Masked Language/Region Modeling
  (ii) Image-Text Matching
- Pre-trained on Conceptual Captions



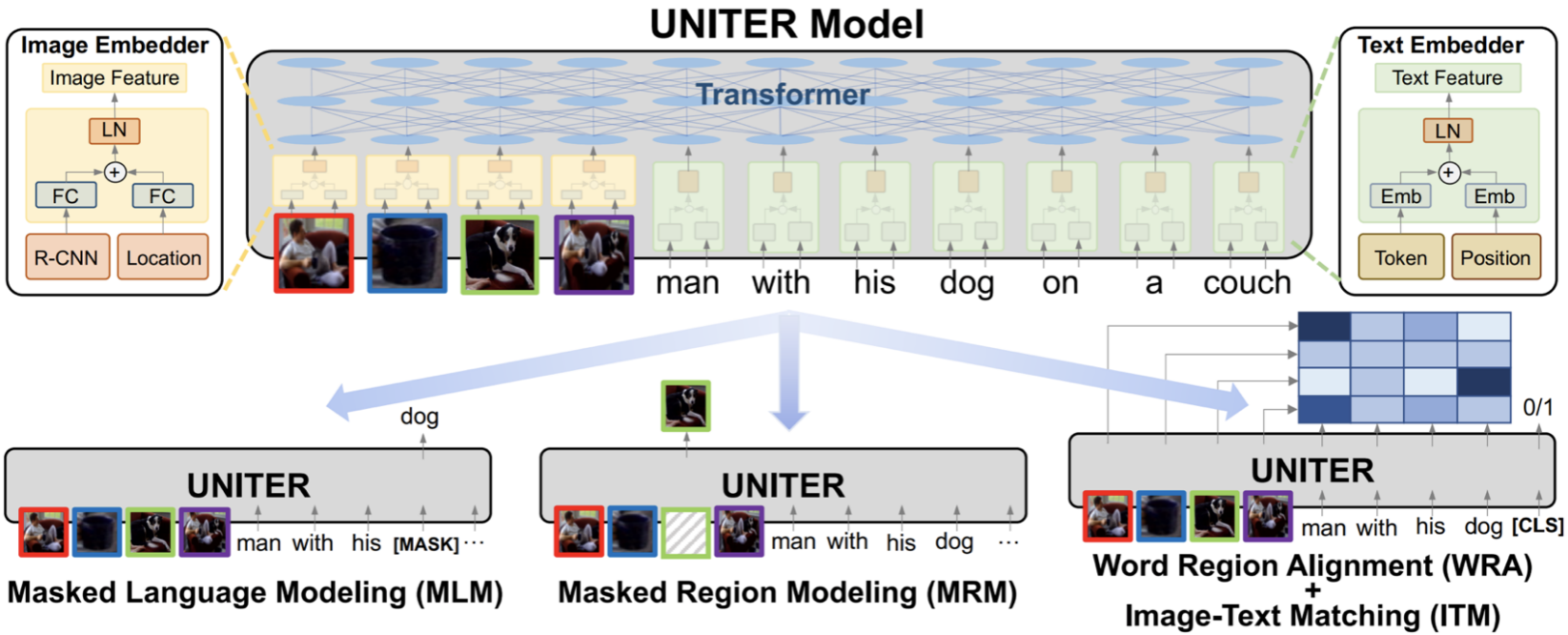
# Grounded Transformers: Single-Stream Models

# Grounded Transformers: Single-Stream Models

## UNITER [Chen et al., 2020]

### UNiversal Image TExt Representation [Chen et al., 2020]

- **Input** image regions and textual words **mapped into common embedding space**
- **Transformer** module operates on common space

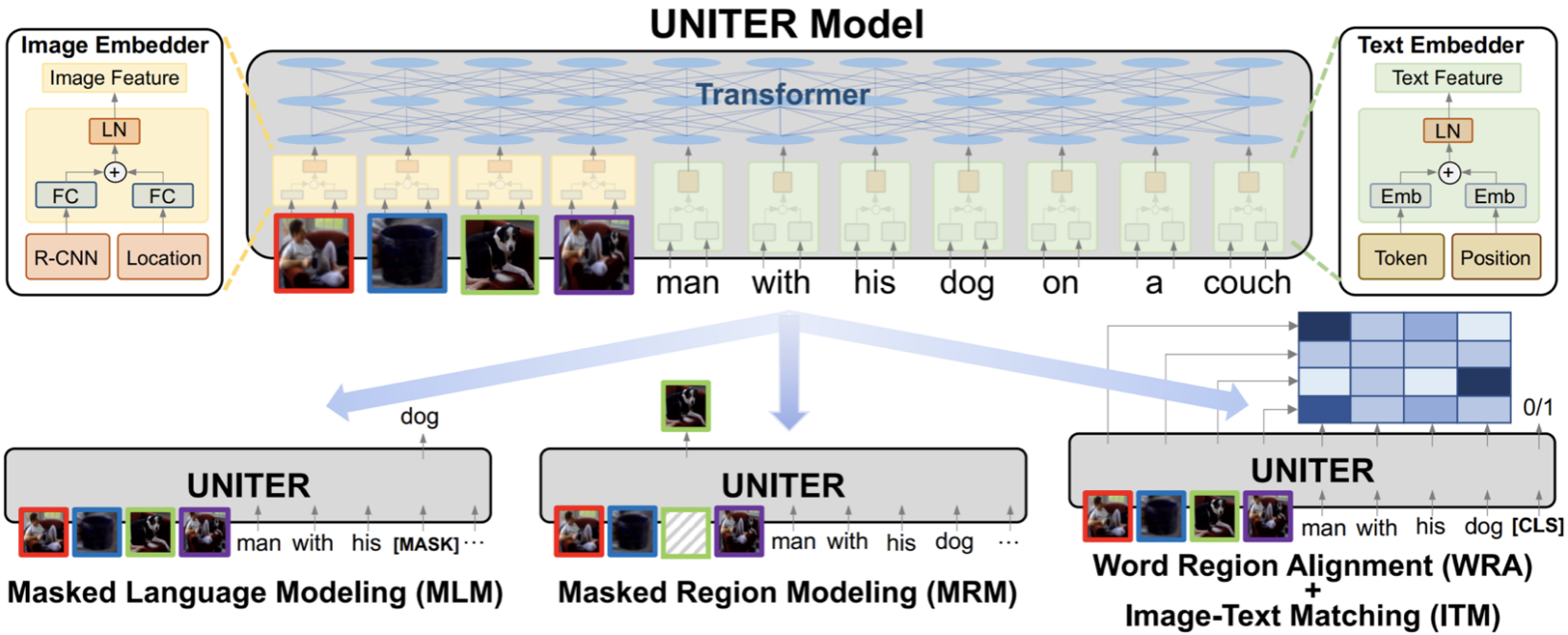


# Grounded Transformers: Single-Stream Models

## UNITER [Chen et al., 2020]

### UNiversal Image TExt Representation [Chen et al., 2020]

- **Input** image regions and textual words **mapped into common embedding space**
- **Transformer** module operates on common space

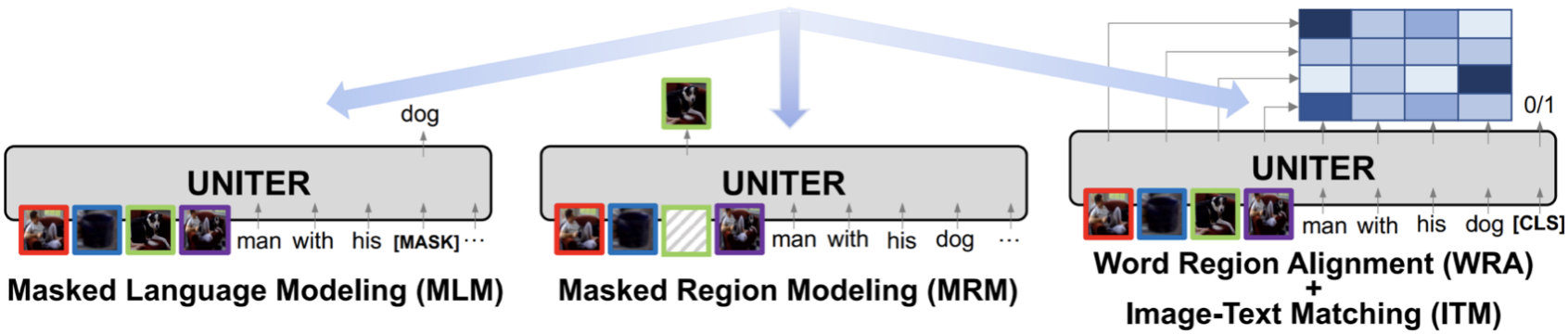


# Grounded Transformers: Single-Stream Models

## UNITER [Chen et al., 2020]

### UNiversal Image TExt Representation [Chen et al., 2020]

- **Input** image regions and textual words **mapped into common embedding space**
- **Transformer** module operates on common space
- Pre-training (proxy) tasks:
  (i) Masked Language/Region Modeling **conditioned on image/text** (i.e., mask only one modality at a time)
  (ii) Image-Text Matching
  (iii) **Word-Region Alignment**

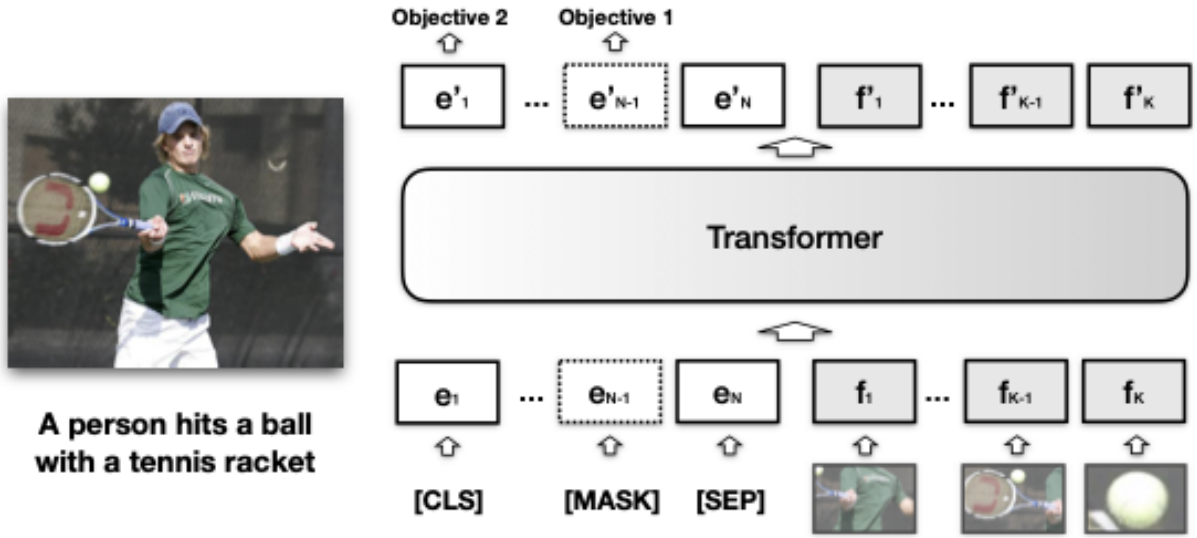


# Grounded Transformers: Single-Stream Models

## VisualBERT [Li et al., 2019, Li et al., 2020]

### VisualBERT [Li et al., 2019, Li et al., 2020]

- **Input** image regions and textual words mapped into common embedding space
- **Transformer** module operates on common space



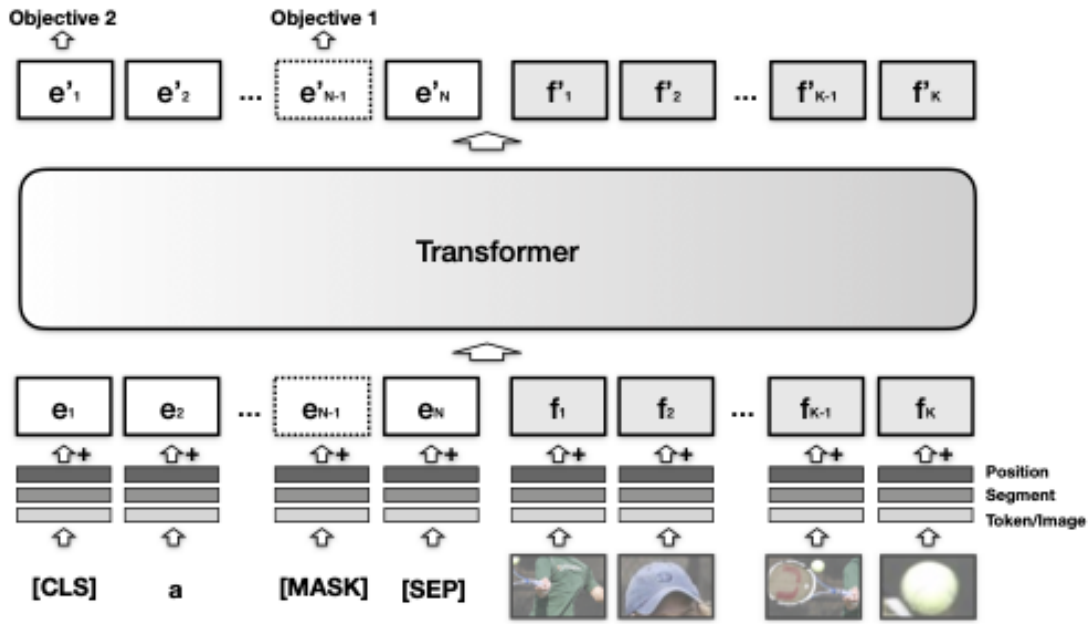
# Grounded Transformers: Single-Stream Models

## VisualBERT [Li et al., 2019, Li et al., 2020]

### VisualBERT [Li et al., 2019, Li et al., 2020]

- **Input** image regions and textual words mapped into common embedding space
- **Transformer** module operates on common space

A person hits a ball with a tennis racket



# Grounded Transformers: Single-Stream Models

## VisualBERT [Li et al., 2019, Li et al., 2020]

### VisualBERT [Li et al., 2019, Li et al., 2020]

- **Input** image regions and textual words mapped into common embedding space
- **Transformer** module operates on common space
- Pre-training (proxy) tasks:
  (i) Masked Language Modeling conditioned on image
  (ii) Image-Text Matching
- Pretraining dataset: MS COCO

# Grounded Transformers: Datasets

## Datasets used to pre-train models on proxy tasks: UNITER

- **Conceptual Captions** [Sharma et al., 2018, Ng et al., 2020]
  Automatically processed alt-text into clean image captions

**Alt-text**: A Pakistani worker helps to clear the debris from the Taj Mahal Hotel November 7, 2005 in Balakot, Pakistan.

**Conceptual Captions**: a worker helps to clear the debris.

**Alt-text**: Musician Justin Timberlake performs at the 2017 Pilgrimage Music & Cultural Festival on September 23, 2017 in Franklin, Tennessee.

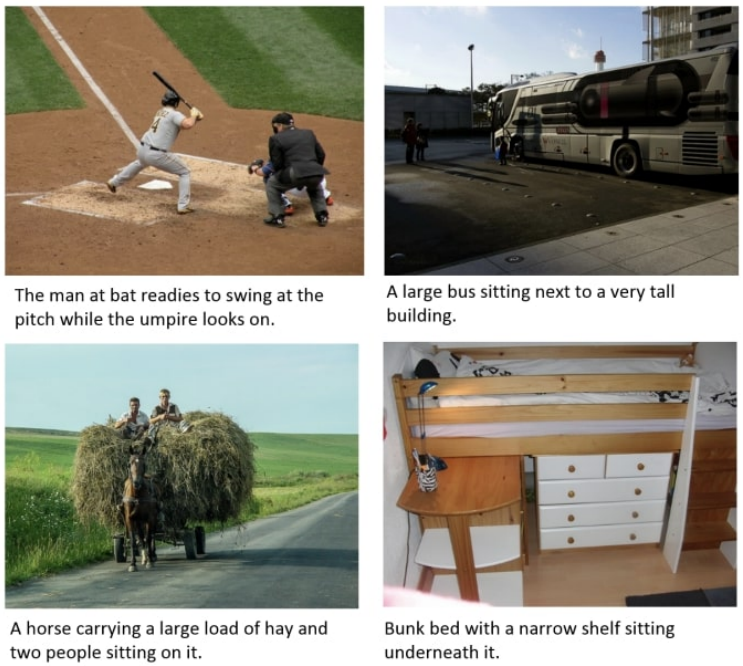
**Conceptual Captions**: pop artist performs at the festival in a city.

$\Rightarrow$ Text and image are semantically weakly associated!

# Grounded Transformers: Datasets

## Datasets used to pre-train models on proxy tasks: UNITER

- **Conceptual Captions** [Sharma et al., 2018, Ng et al., 2020]
  Automatically processed alt-text into clean image captions
- **MS COCO** [Chen et al., 2015]
  Human-elicited captions for "Common Objects in Context"




Five captions per image.

# Grounded Transformers: Datasets

## Datasets used to pre-train models on proxy tasks: UNITER

- **Conceptual Captions** [Sharma et al., 2018, Ng et al., 2020]
  Automatically processed alt-text into clean image captions
- **MS COCO** [Chen et al., 2015]
  Human-elicited captions for "Common Objects in Context"
- **Visual Genome** [Krishna et al., 2016]
  Human-elicited descriptions of image regions
- **SBU Captions** [Ordonez et al., 2011]

# Visual Representations

## Development of Visual Feature Representations

| | |
|---|---|
| early | *handcrafted* feature vectors |
| 2012+ | Convolutional Neural Networks |
| 2019+ | Visual Transformers |
| Dataset | Huge manually labeled datasets required |
| 2021+ | Visual-Linguistic Models (transformer-based)<br>learn in unsupervised way from image-text pairs |

# Visual Representations

## Development of Visual Feature Representations

| | |
|---|---|
| early | *handcrafted* feature vectors |
| 2012+ | Convolutional Neural Networks |
| 2019+ | Visual Transformers |
| Dataset | Huge manually labeled datasets required |
| 2021+ | **Visual-Linguistic Models (transformer-based) learn in unsupervised way from image-text pairs** |

# Pretrained Visual(-Linguistic) Models

## Multimodal Models: Language for Visual Pretraining

# Multimodal (VL) Representations

## Early models

Visual representations
+
Textual representations

**Fusion techniques:**

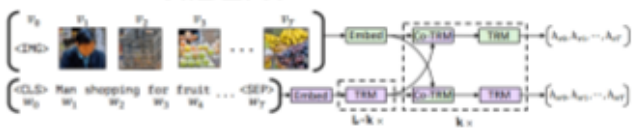
word2vec
autoencoders
CCA
concatenation
…

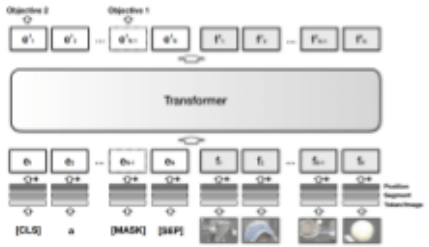
## Transformers

### BERT-like models

Visual encoder
+
Textual encoder

VilBERT






VisualBERT

### Contrastive learning models

Visual encoder
+
Textual encoder

CLIP

### Instruction-tuned and few-shot learning models

Flamingo
BLIP-2
BLIPInstruct
LLaVA

# Pretrained Visual(-Linguistic) Models

Language for Visual Pretraining

- Supervised pretraining paradigm with object recognition limits visual recognition systems to closed-set of visual concepts
- ? How do we **recognise** linguistic variants and **unseen objects**?
  e.g., "sandwich", "BLT", ...
- ⇒ Learn to project visual and text embeddings into common space, s.th.
  - associated embeddings are close
  - disassociated embeddings are away from each other
- ⇒ Enables *zero-shot* visual recognition (no fine-tuning on task!)

# Pretrained Visual(-Linguistic) Models: Overview

- CLIP (dual encoder ) [Radford et al., 2021a, Radford et al., 2021b]
- ALIGN (dual encoder) [Jia et al., 2021]
- Flava (fusion encoder + dual decoder) [Singh et al., 2022]
- Flamingo [Alayrac et al., 2022a]

**Further references:**
Vision-Language Models for Vision Tasks: A Survey
Zhang et al. (2023) `https://arxiv.org/pdf/2304.00685.pdf`

# Pretrained Visual(-Linguistic) Models: Overview

| Method | Datasets | Objective | |
|---|---|---|---|
| CLIP | CLIP (loosely associated V+T) | Contrastive | zero-shot classification |
| ALIGN | ALIGN (noisy V+T pairs) | Contrastive | |
| Flava | 9 diff. datasets | Contrastive + Generative | universal, foundational VLM uni- & multimodal |
| Flamingo | diff. datasets | Generative (V-conditioned text generation) | images & videos few-shot learning |
| SimVLM | ALIGN, C4 | Prefix Language Modelling | Image captioning, Visual QA |

**Further references:**

- Vision-Language Models for Vision Tasks: A Survey
  Zhang et al. (2023)
  `https://arxiv.org/pdf/2304.00685.pdf`
- HuggingFace blog entry on VL-pretraining:
  `huggingface.co/blog/vision_language_pretraining`

# Pretrained Visual(-Linguistic) Models

## CLIP

### CLIP [Radford et al., 2021a, Radford et al., 2021b]

- Motivation: Learn visual features from natural language supervision
  Goal: Zero-shot visual classification
- Approach:
  - Combination of language and visual transformers [Dosovitskiy et al., 2021]
  - Contrastive learning objective: maximise embedding similarity of aligned image–text pairs
- Dataset: Image–text pairs crawled from internet
  cf. linguistic representation models

# CLIP [Radford et al., 2021a]

## Why Using Natural Language Supervision for Vision?

- Issues of current CV approaches using vision datasets:
  - datasets are labour intensive and expensive
  - contain limited set of concepts
  - standard vision models generalise badly to new tasks
  - benchmark results not reliable with respect to the models true visual recognition ability

# CLIP: Contrastive Language-Image Pre-Training

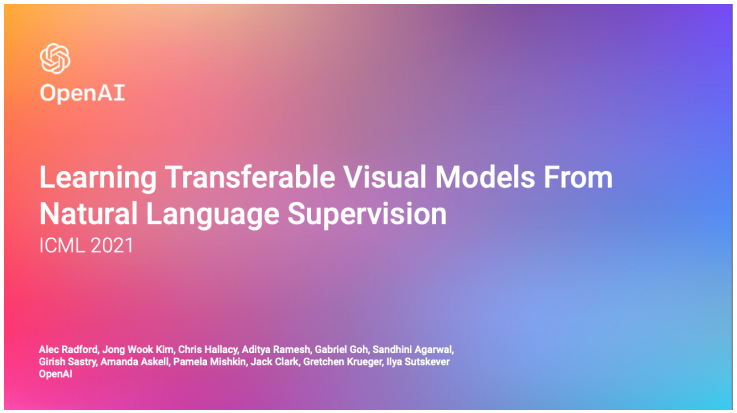
## CLIP [Radford et al., 2021a]

- Goal: Obtain more robust visual representations, by pre-training model on visual-linguistic web data
- Data has a large variety of images and natural language
- Objective: contrastive language supervision: given an image, predict which out of a set random text snippets is the correct one
- Pre-trained model allows zero-shot transfer to existing computer vision classification datasets

# CLIP: Contrastive Language-Image Pre-Training

[Radford et al., 2021a]

**The following slides are from the presentation of the authors of CLIP at the conference where the paper was published (ICML 2021)**



# Contrastive learning

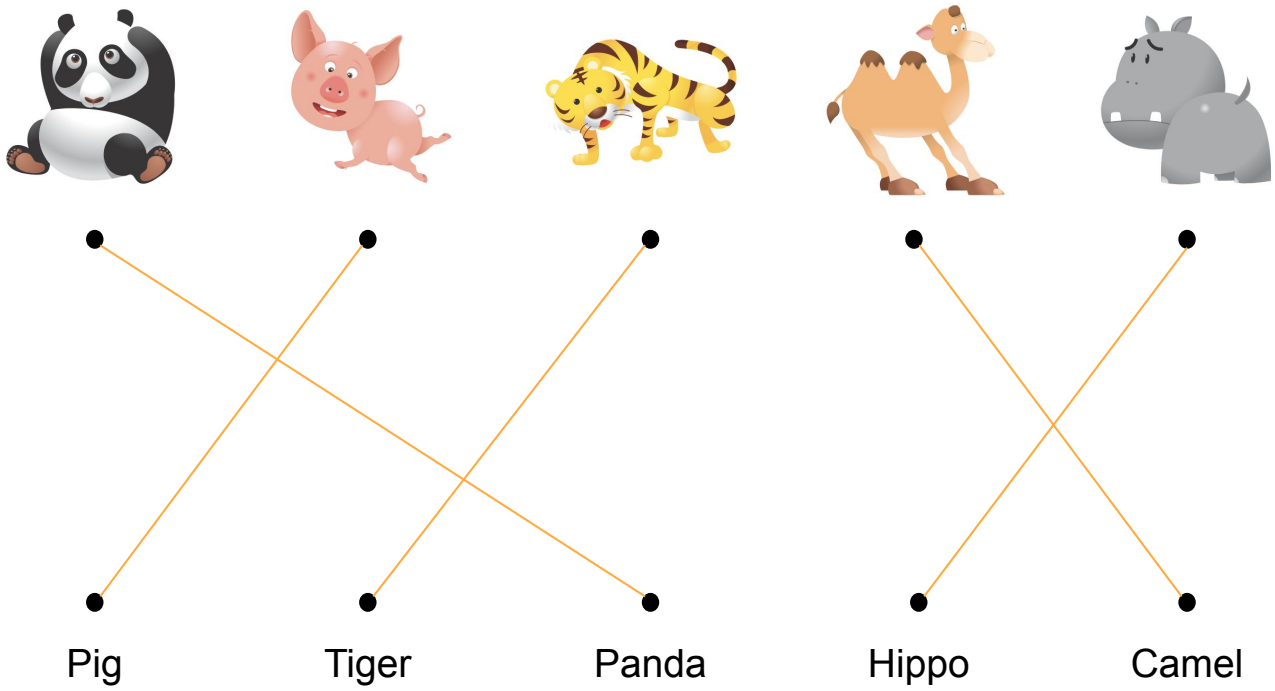
# Contrastive learning





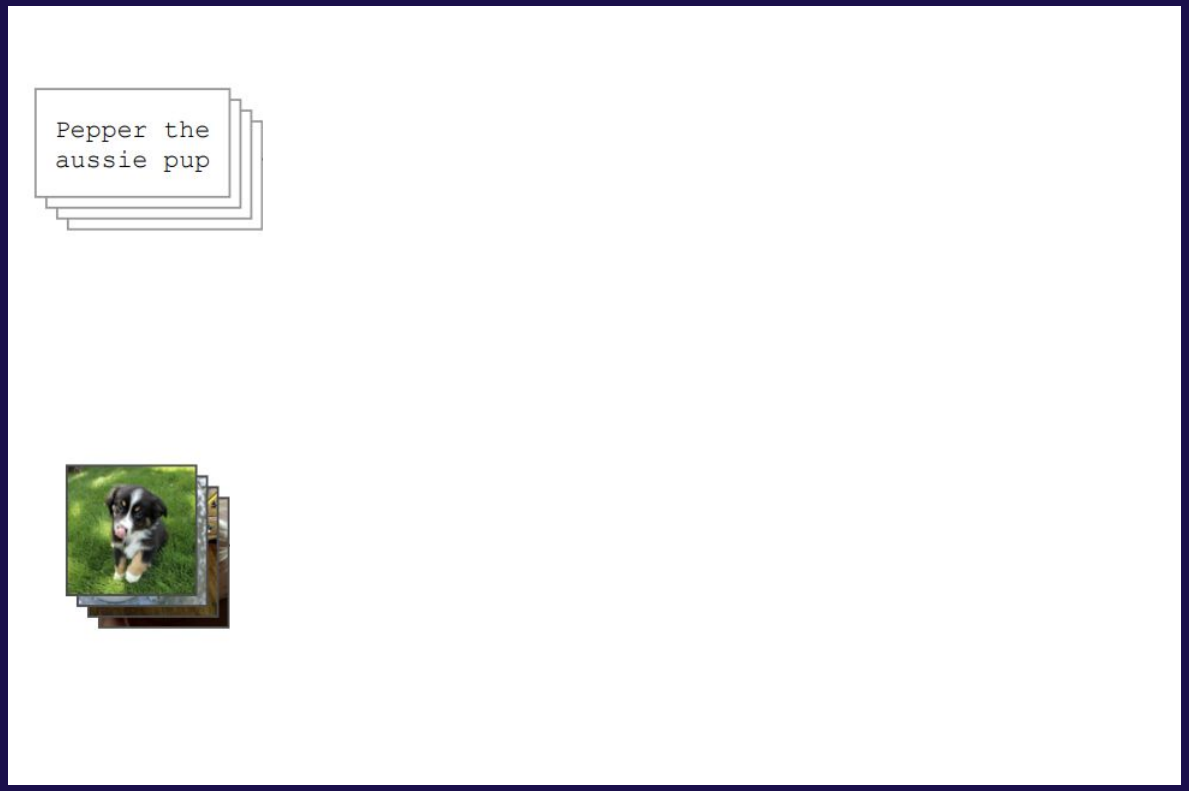
Pig Tiger Panda Hippo Camel

# Contrastive learning

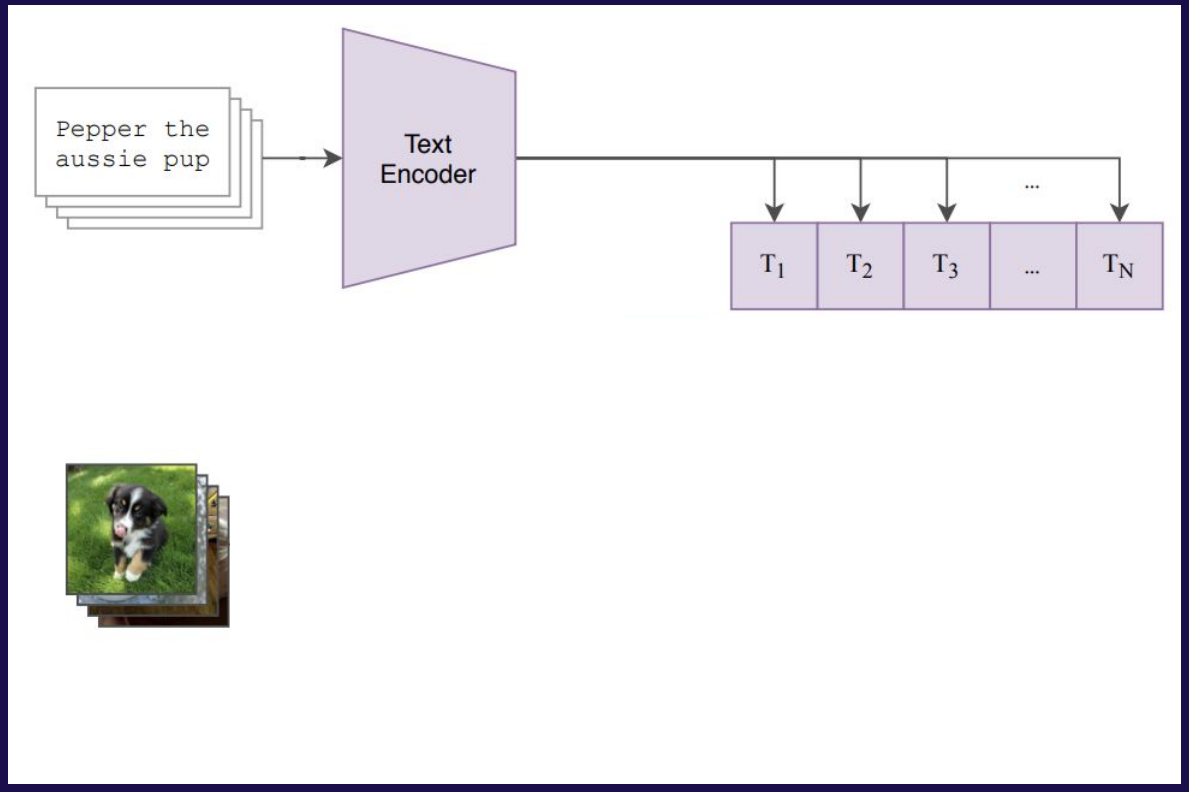


# CLIP: Contrastive Language-Image Pre-training

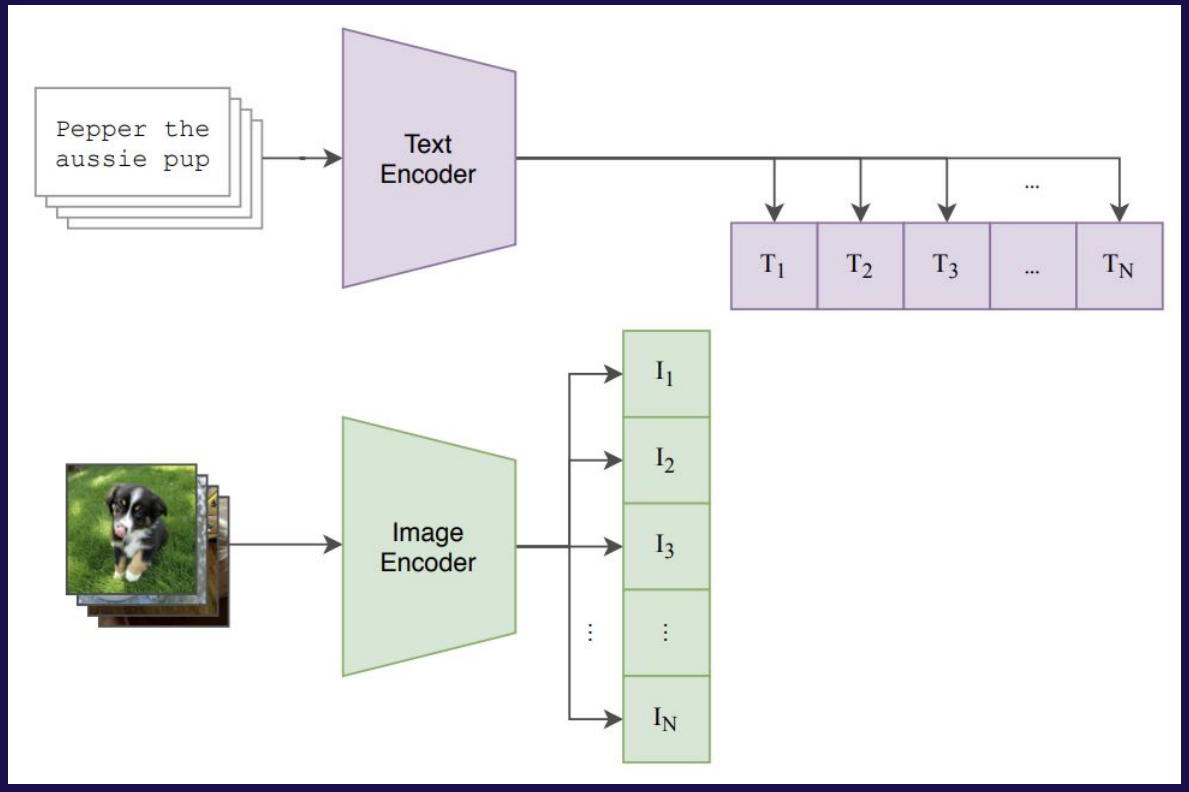
# CLIP: Contrastive Language-Image Pre-training



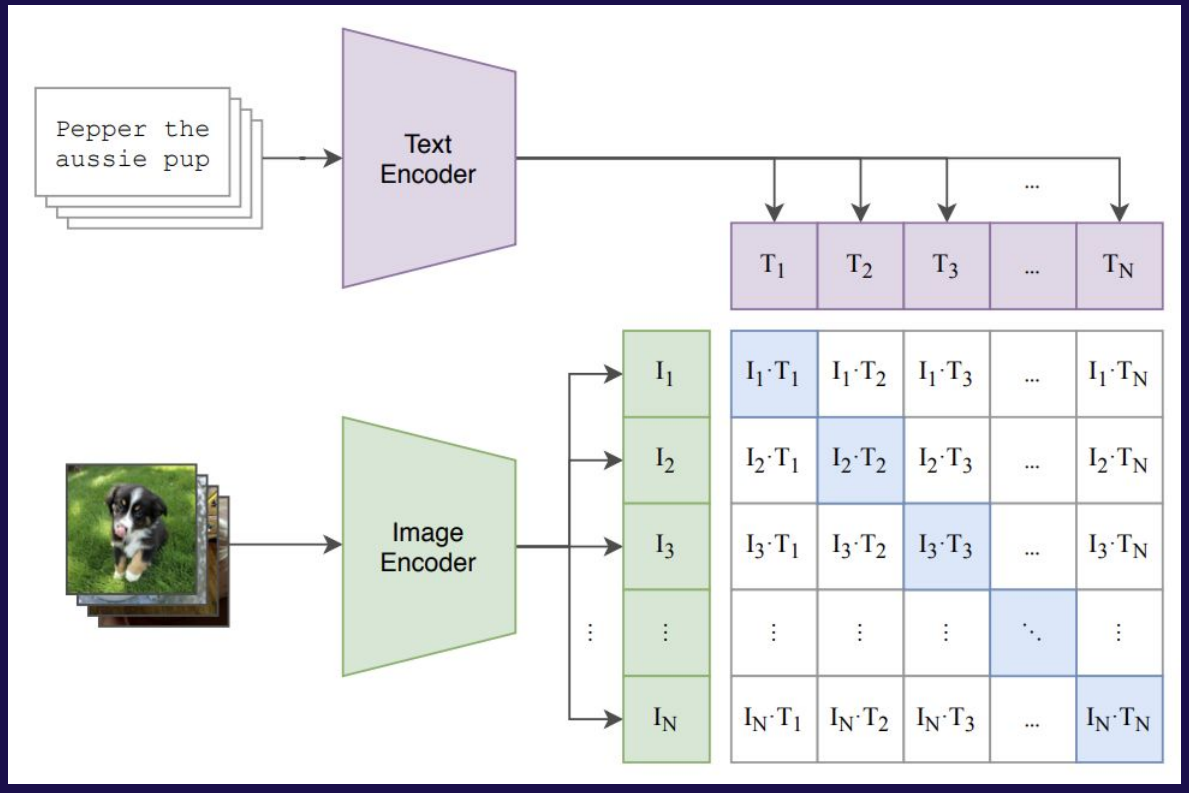
# CLIP: Contrastive Language-Image Pre-training



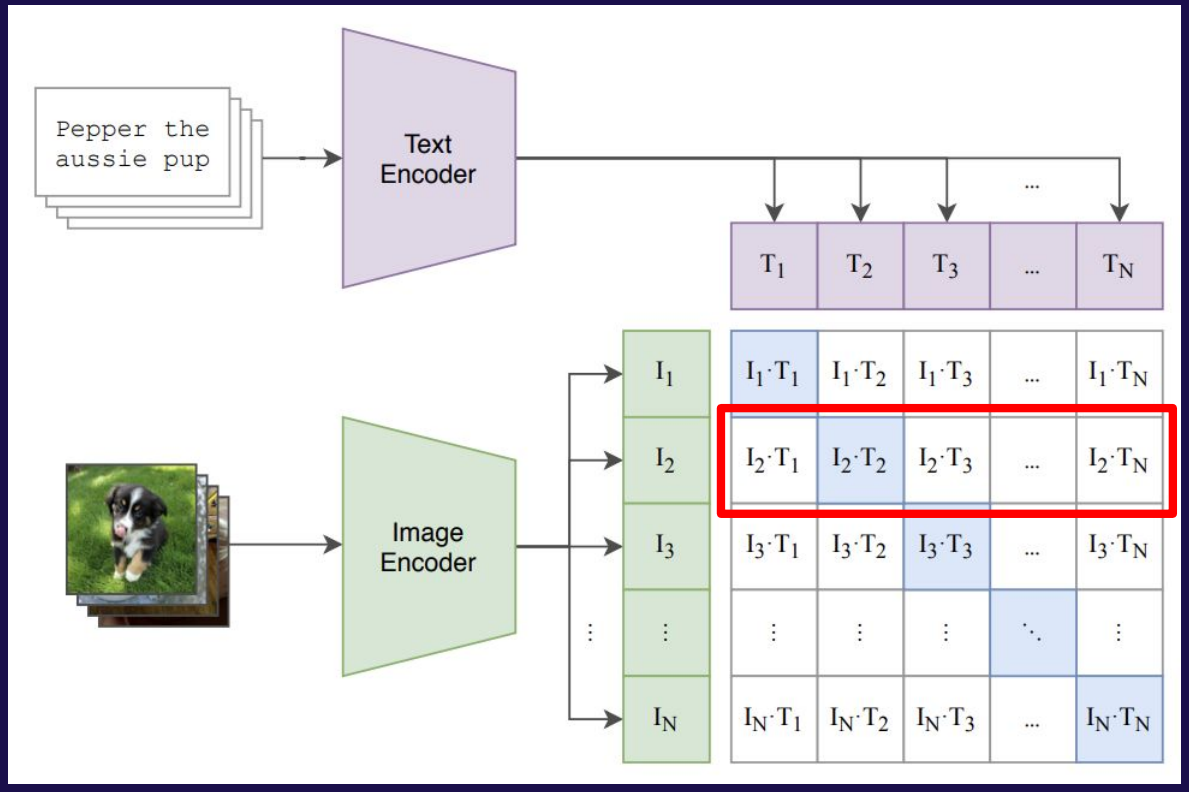
# CLIP: Contrastive Language-Image Pre-training



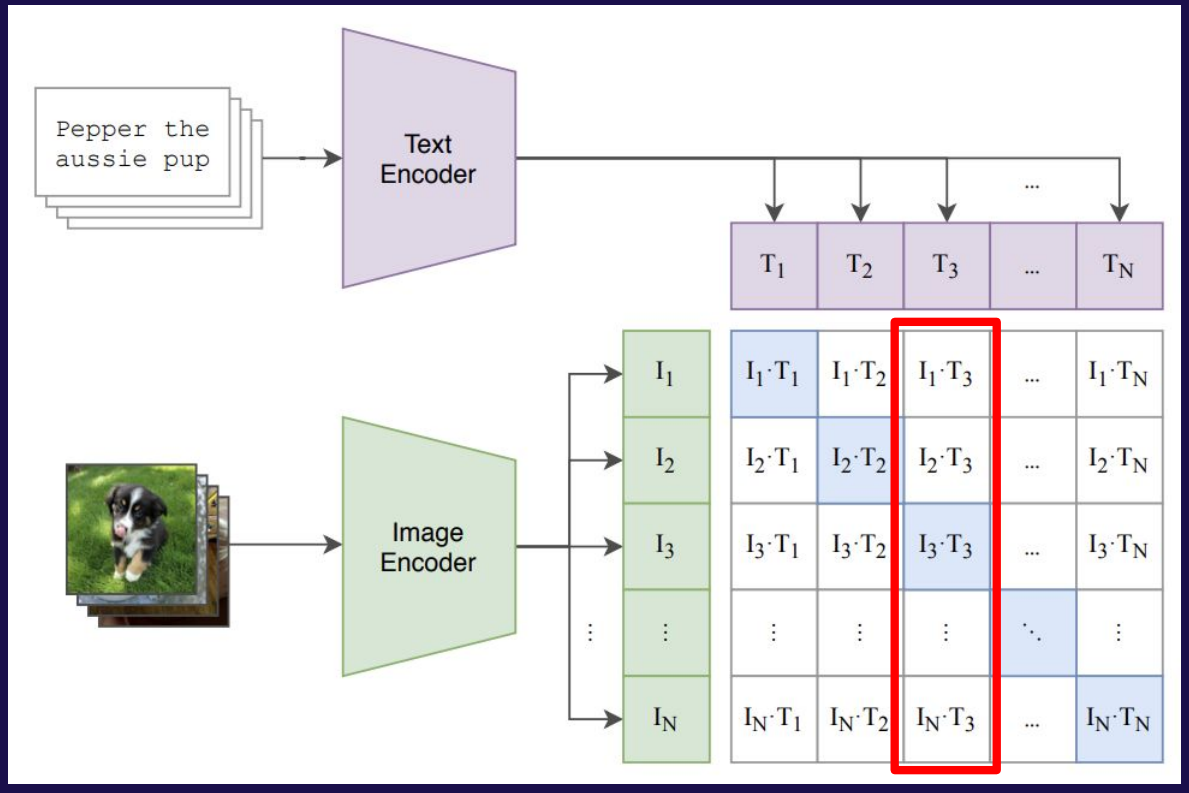
# CLIP: Contrastive Language-Image Pre-training



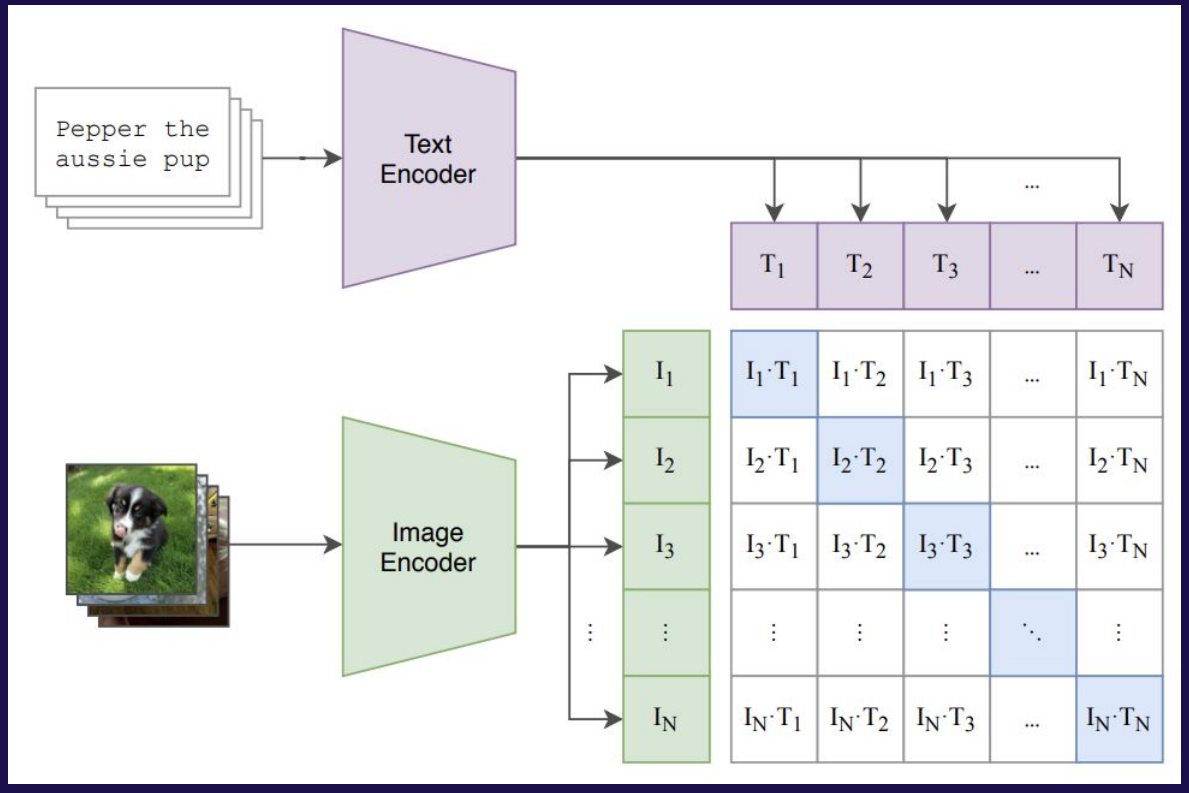
# CLIP: Contrastive Language-Image Pre-training



# CLIP: Contrastive Language-Image Pre-training



# CLIP: Contrastive Language-Image Pre-training

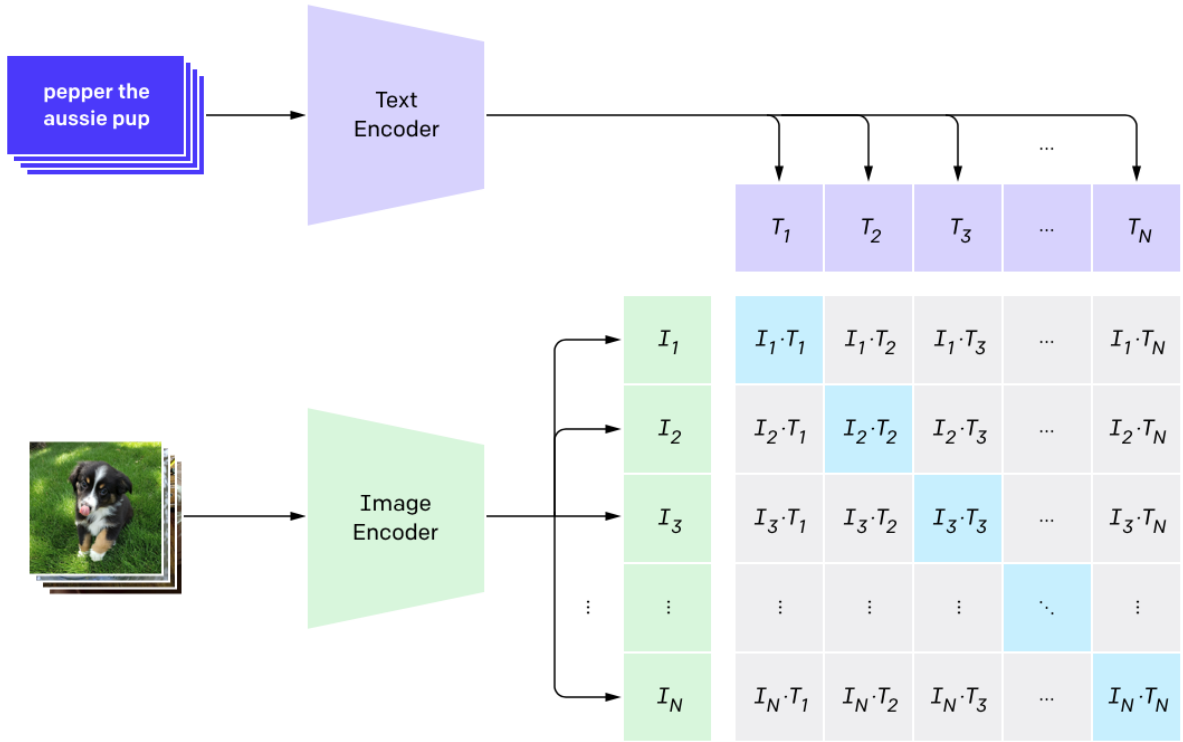


# Pretrained Visual(-Linguistic) Models

## CLIP

### CLIP: Contrastive Pretraining

[Radford et al., 2021a, Radford et al., 2021b]



# Pretrained Visual(-Linguistic) Models

## CLIP

### CLIP: Contrastive Loss [Radford et al., 2021a, Radford et al., 2021b]

- Predict which of given $N \times N$ possible (image, text) pairings occurred in data
- Compute pairwise cosine similarity of each T+V pairing
- Loss: Symmetric cross entropy over similarity scores

$$\text{pw\_sim} = \text{cosine}_{sim}(I_{emb}, T_{emb}) * exp(\tau)$$
$$\text{loss} = (loss_i + loss_t)/2, \text{ where}$$
$$\text{loss}_x = \text{cross-entropy-loss}(\text{pw\_sim}, T)$$

$I_{emb}$ normalised image embeddings

$T_{emb}$ normalised text embeddings

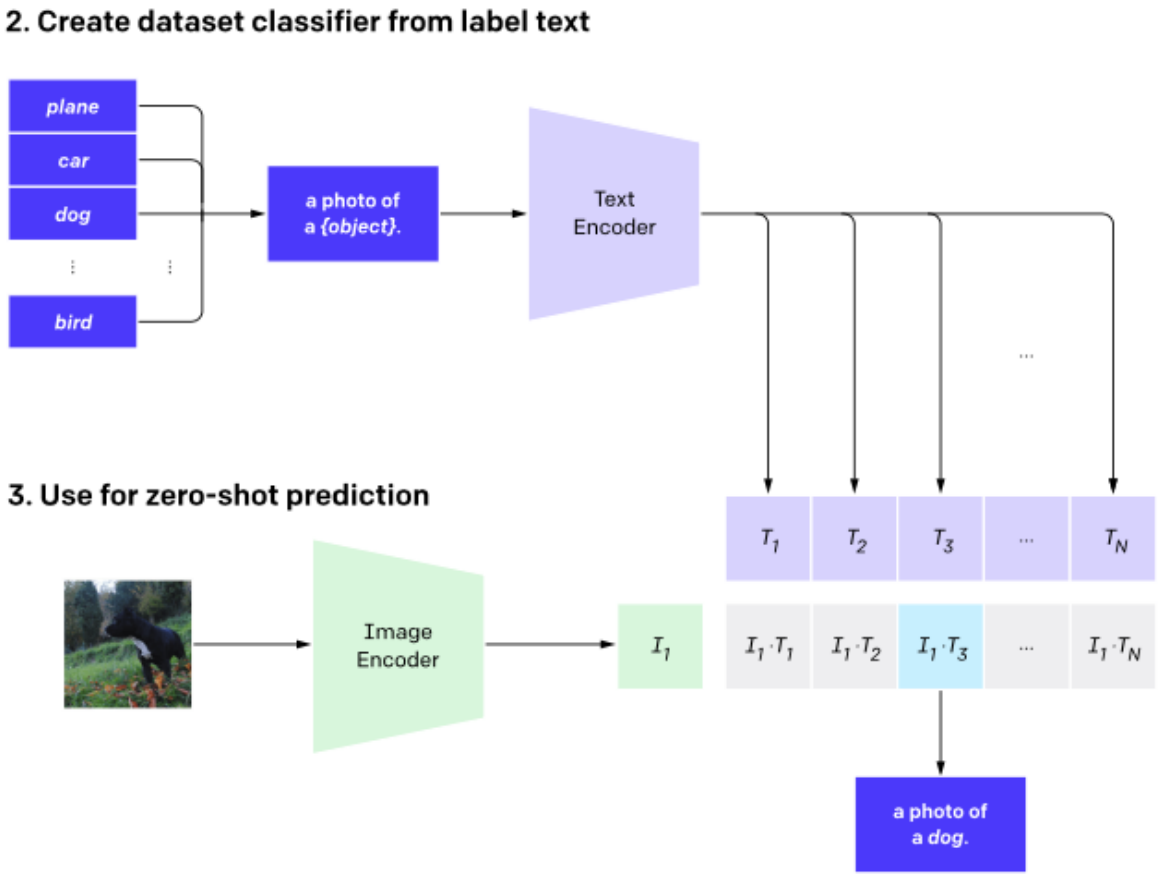
$\tau$ learned temperature parameter

# Pretrained Visual(-Linguistic) Models

## CLIP

### CLIP: Zero-shot Prediction

[Radford et al., 2021a, Radford et al., 2021b]

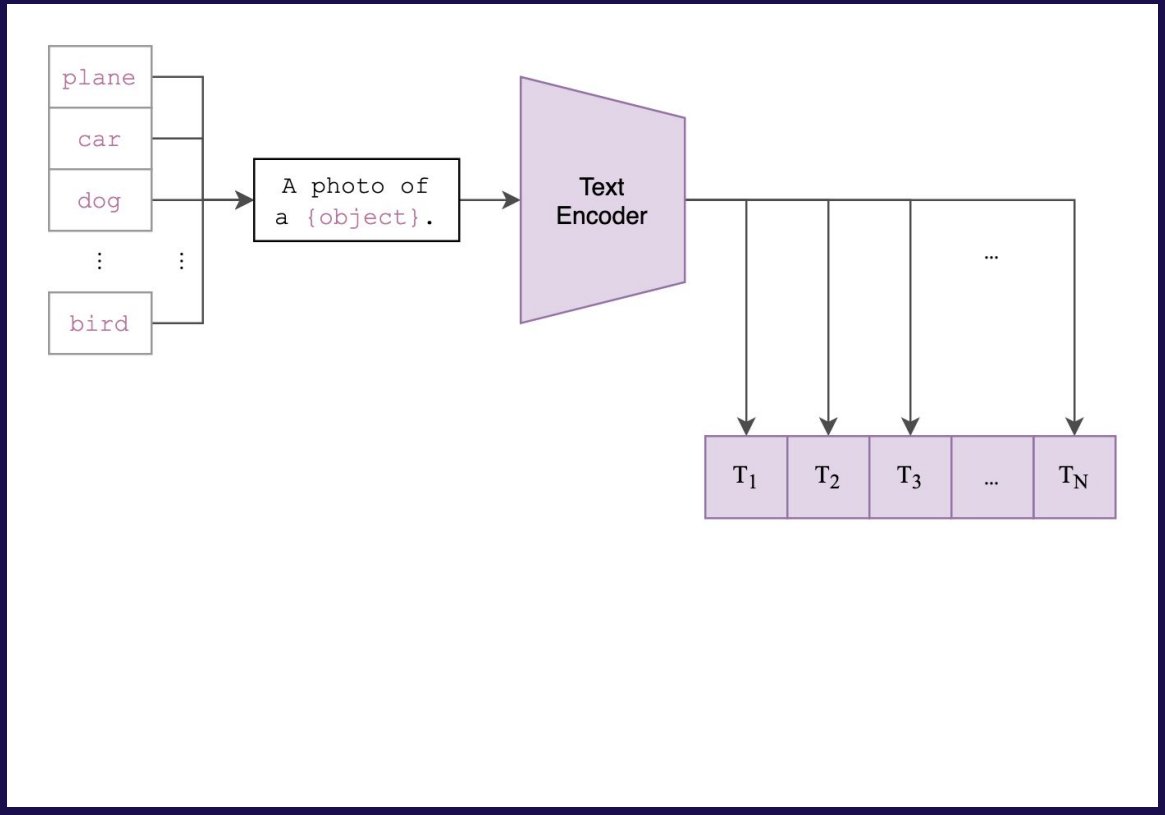


# CLIP: Contrastive Language-Image Pre-Training

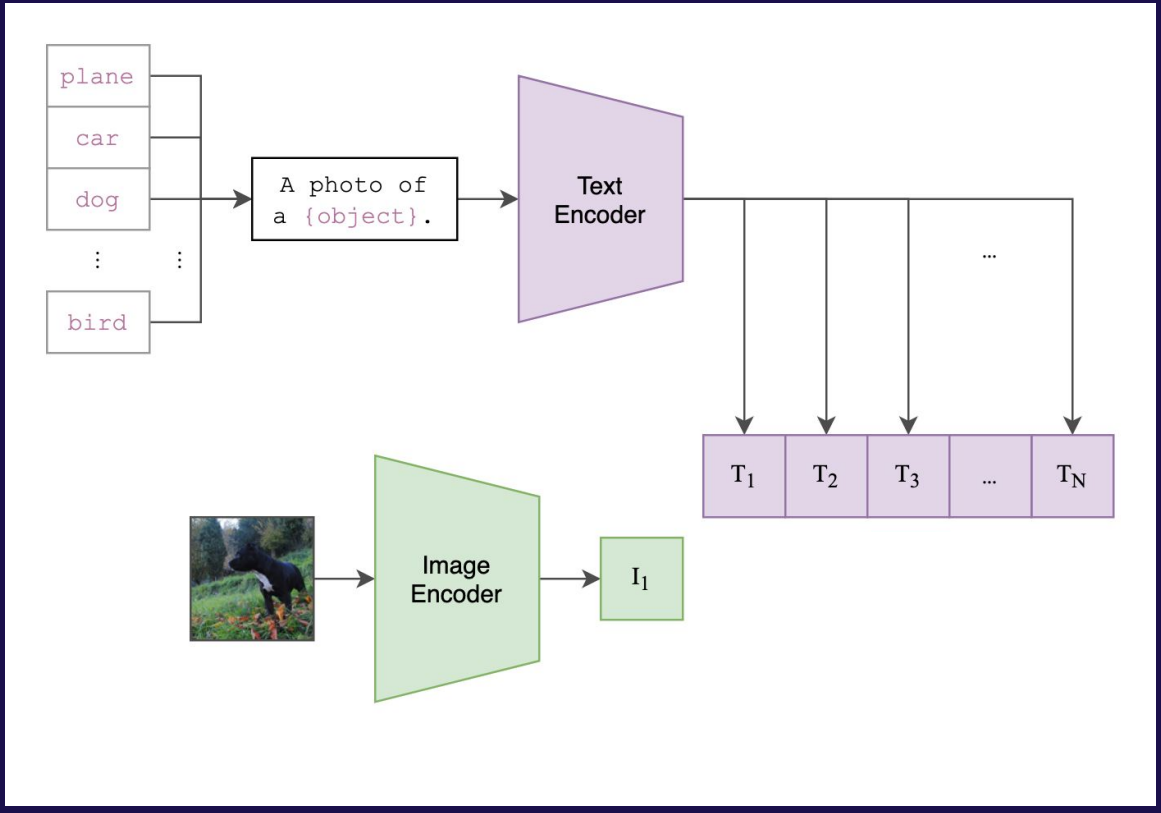
## Effectiveness of CLIP

[Radford et al., 2021a]

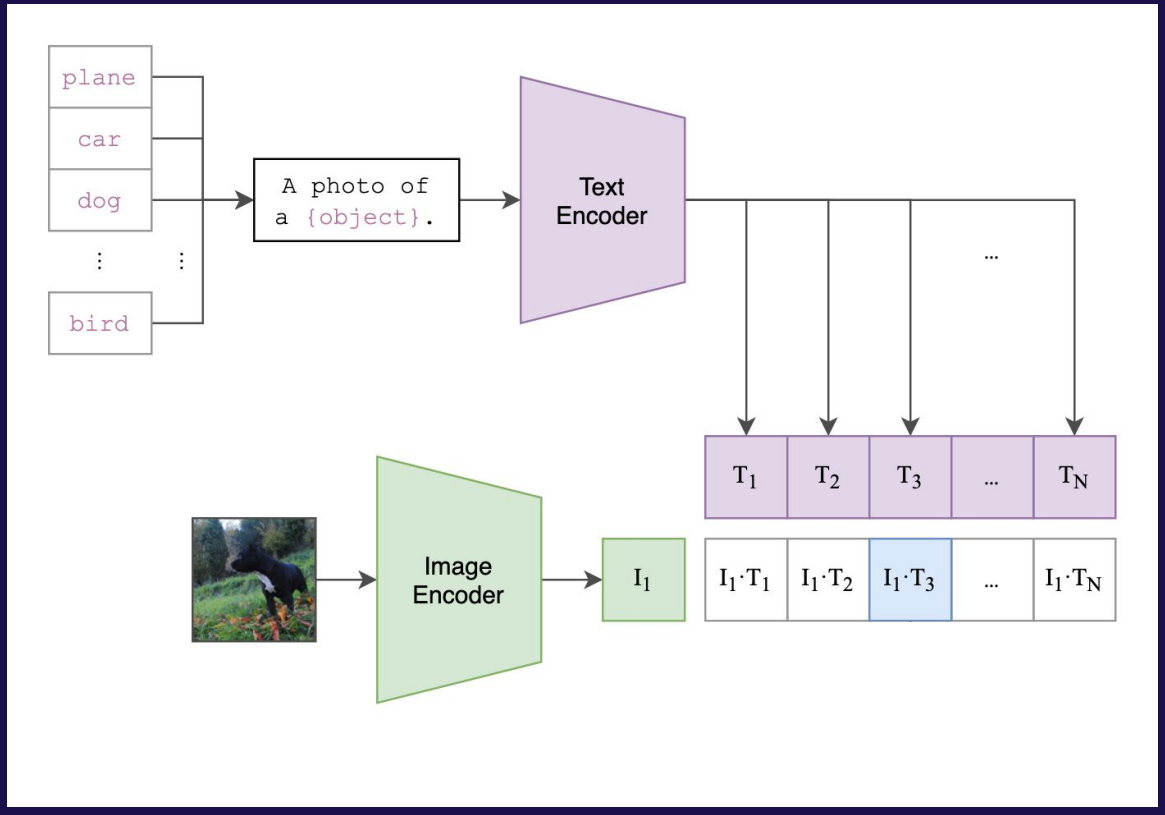
# Zero-shot image classification



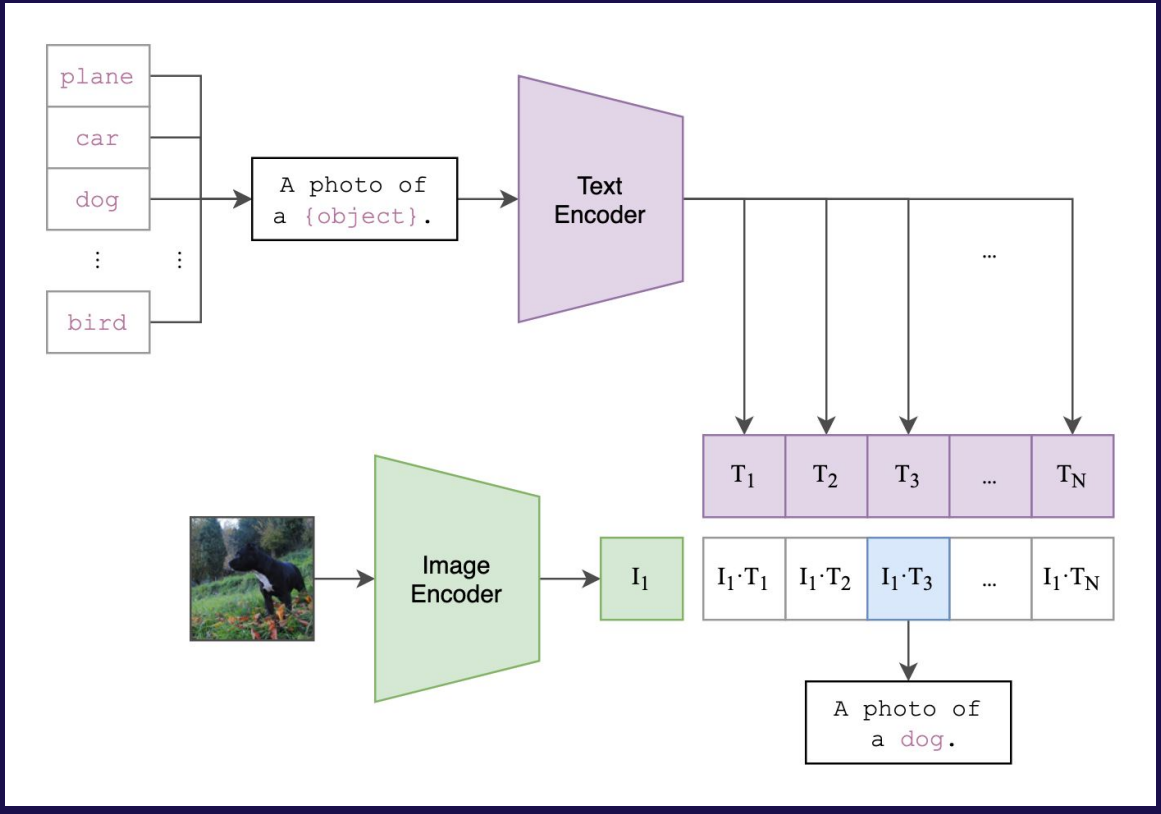
# Zero-shot image classification



# Zero-shot image classification



# Zero-shot image classification

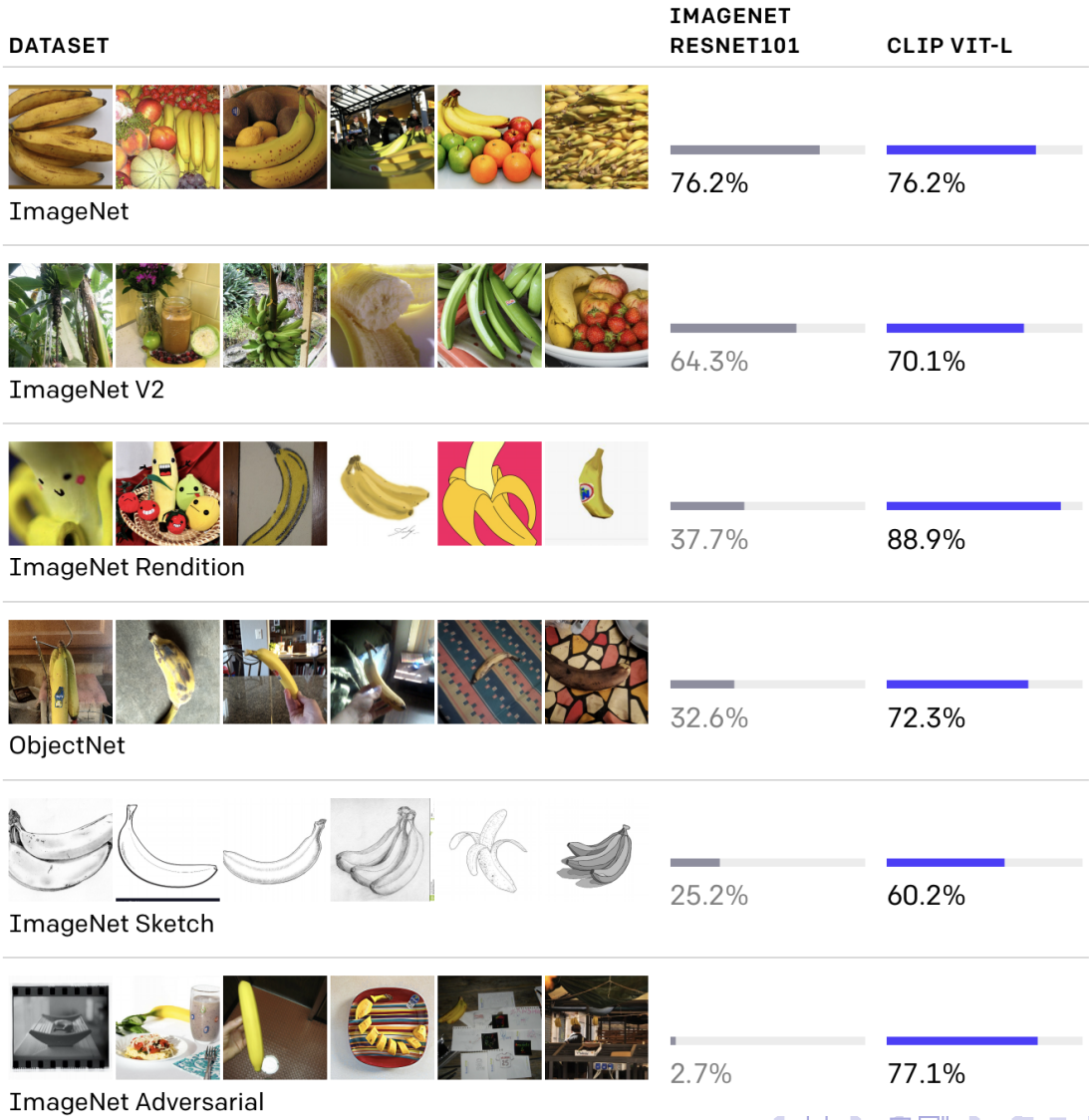


# CLIP : Contrastive Language-Image Pre-Training

## Effectiveness of CLIP

[Radford et al., 2021a]

- Zero-shot CLIP is more robust



# CLIP : Contrastive Language-Image Pre-Training

## Effectiveness of CLIP [Radford et al., 2021a]

- Zero-shot CLIP is more robust
- But requires *textual prompts*, due to training data of *sequences* of words (descriptions): → "A photo of a {label}"
- Benefits from prompt-engineering
  ensembling over multiple zero-shot classifiers, based on different context prompts,
  → e.g., "A photo of a big label", "A photo of a small label"

# Multimodal (VL) Representations

## Early models

Visual representations
+
Textual representations

**Fusion techniques:**

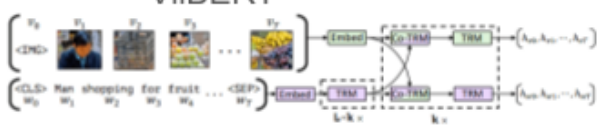
word2vec
autoencoders
CCA
concatenation
...

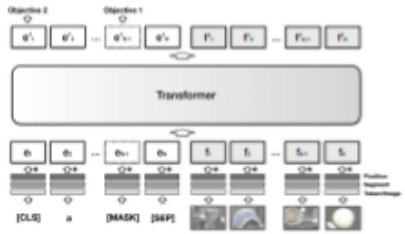
## Transformers

### BERT-like models

Visual encoder
+
Textual encoder

ViLBERT






VisualBERT

### Contrastive learning models

Visual encoder
+
Textual encoder

CLIP

### Instruction-tuned and few-shot learning models

Flamingo
BLIP-2
BLIPInstruct
LLaVA

# Vision–Language Models

[Ghosh et al., 2024]

- **Vision-Language Understanding (dual encoder)**
  **CLIP** (images), VideoCLIP (videos)
- **Text Generation with Multimodal Input (Alignment)**
  GPT-4V, **GIT, ViLT**, **LLaVa-1.5**, Flamingo, PALM-E, BLIP-2, **InstructBLIP** (images)
  LLaMa-VID, Video-LLaMa (videos)
- **Multimodal Input-Output**
  Gemini

# Vision–Language Models

[Ghosh et al., 2024]

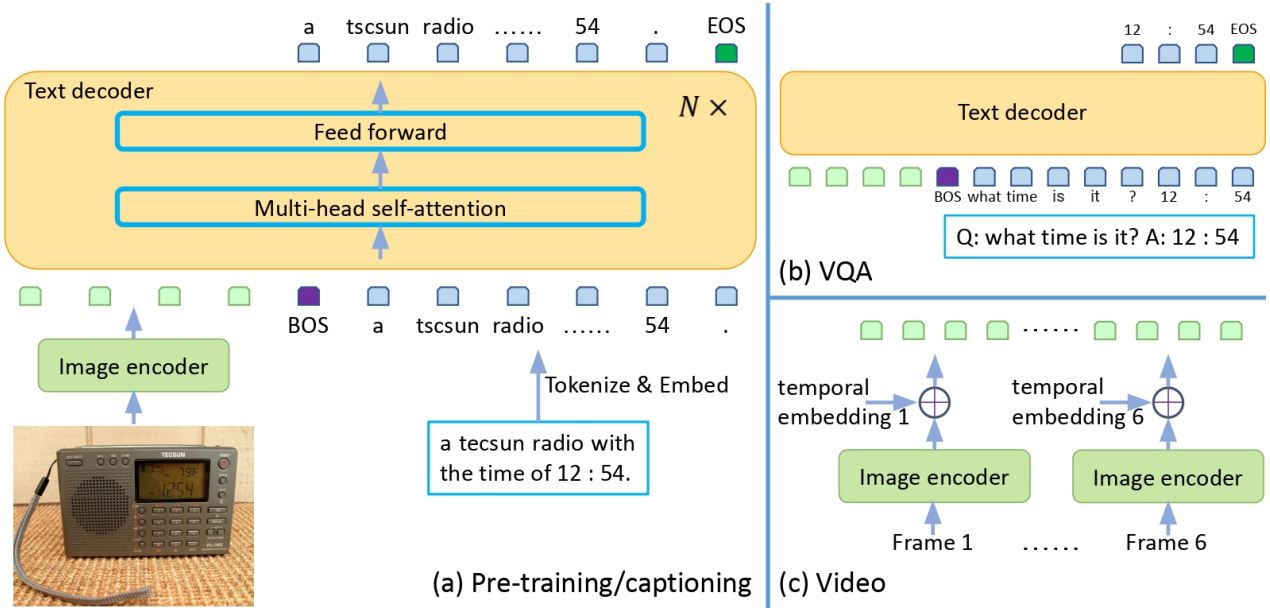
## Text Generation with Multimodal Input (Alignment)

- **GIT**, **ViLT**, **LLaVa-1.5**, **Flamingo**, **InstructBLIP**
- GPT-4V, PALM-E, BLIP-2,
- LLaMa-VID, Video-LLaMa (videos)

# Vision-Language Models: GIT

## GIT (GenerativeImage2Text)

- Transformer **decoder**, conditioned on both CLIP image patch tokens and text tokens
- Goal: predict the next text tokens, given the image tokens and the previous text tokens

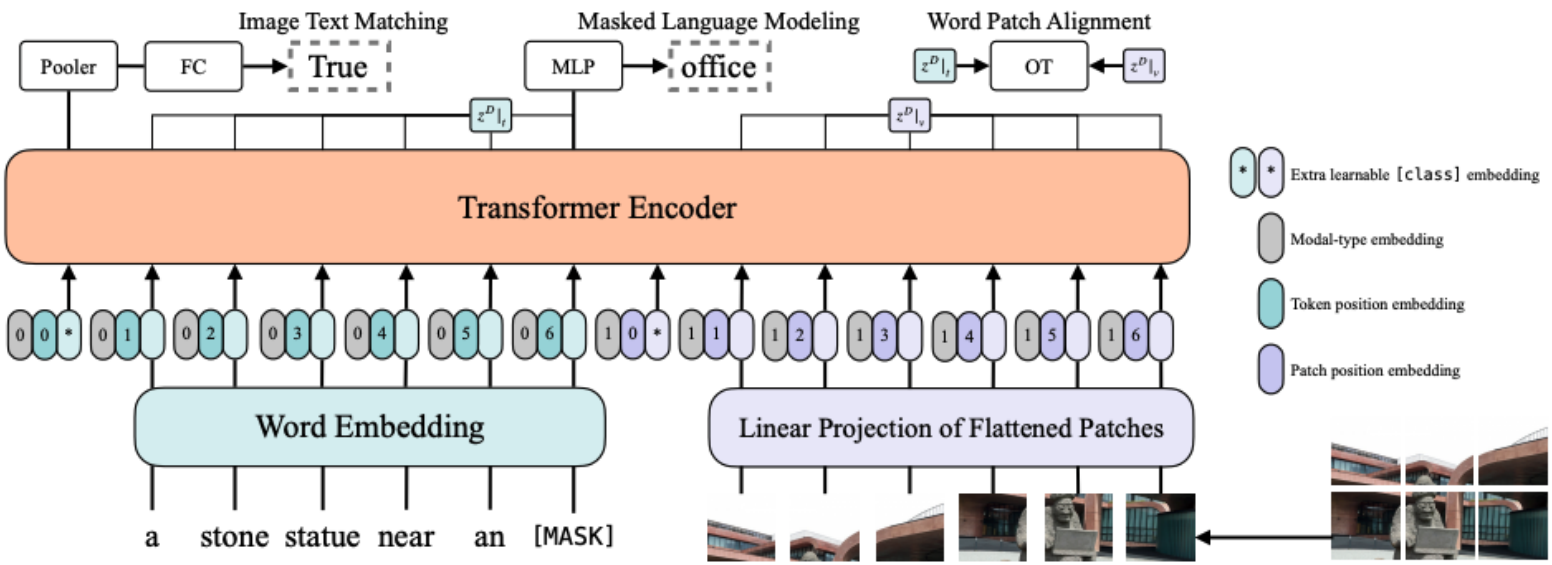


# Vision-Language Models: ViLT

[Kim et al., 2021]

## Vision-and-Language Transformer (ViLT)

- Minimal VL transformer **encoder**
- Visual inputs are processed like text (see ViT): images $\rightarrow$ flattened pixel patches $\rightarrow$ mapped to vectors + position embedding
- Pre-trained on image text matching, masked language modeling, word patch alignment.



# Vision–Language Models: Interim

## Instruction-tuned Large Language Models (LLMs)

- Idea: Learn to perform a new task given a short instruction
- Use language as an interface to an LLM
- Guide the LLM towards solving a task of interest
- Found to improve zero- and few-shot generalisation abilities of LLMs

# Vision–Language Models

[Alayrac et al., 2022b, Awadalla et al., 2023]

## In-context Few-shot Learning Models

- VL models trained with a contrastive objective (e.g., CLIP) allow for zero-shot adaptation to novel tasks without fine-tuning
- But only limited use cases (simple similarity score between image and text)
- No language generation ability

# Vision–Language Models

[Alayrac et al., 2022b, Awadalla et al., 2023]

## In-context Few-shot Learning Models: Flamingo

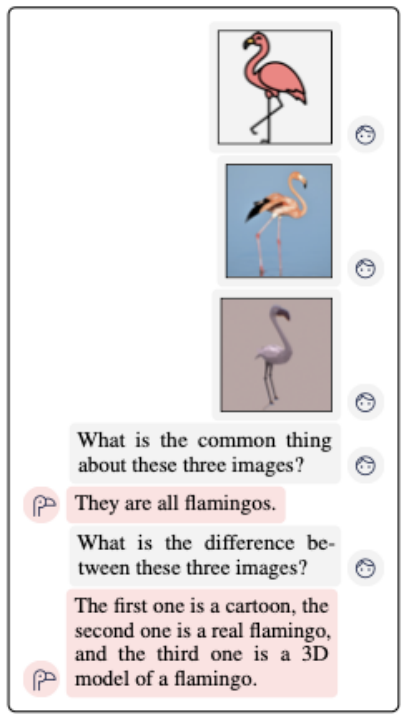
- Idea: Prompt model with a few input-output examples such that it rapidly adapts to a range of VL tasks
- Visually-conditioned autoregressive text generation model
- Combines vision model and LLM, both pre-trained and frozen
- Trained on multimodal web data

# Vision–Language Models

[Alayrac et al., 2022b, Awadalla et al., 2023]

## In-context Few-shot Learning Models: Flamingo

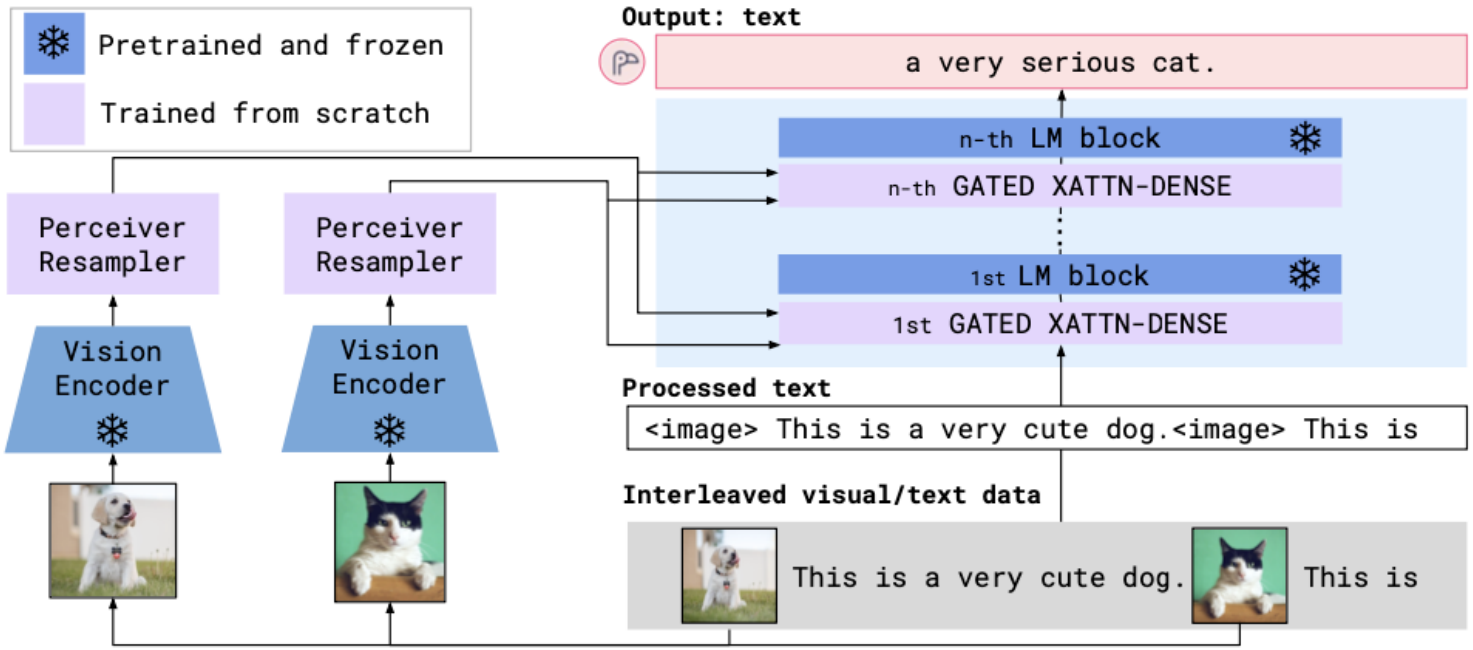
- Idea: Prompt model with a few input-output examples such that it rapidly adapts to a range of VL tasks
- Visually-conditioned autoregressive text generation model
- Combines vision model and LLM, both pre-trained and frozen
- Trained on multimodal web data



# Vision–Language Models

[Alayrac et al., 2022b, Awadalla et al., 2023]

## In-context Few-shot Learning Models: Flamingo




$$p(y|x) = \sum_{l=1}^{L} p(y_l|y_{<l}, x_{\leq l})$$

$y_l$: $l$-th language token of input; $y_{<l}$: set of preceding tokens

$x_{\leq l}$: set of images/videos preceding $y_l$

# Vision–Language Models

[Alayrac et al., 2022b, Awadalla et al., 2023]

## Few-shot Learning Models: Flamingo

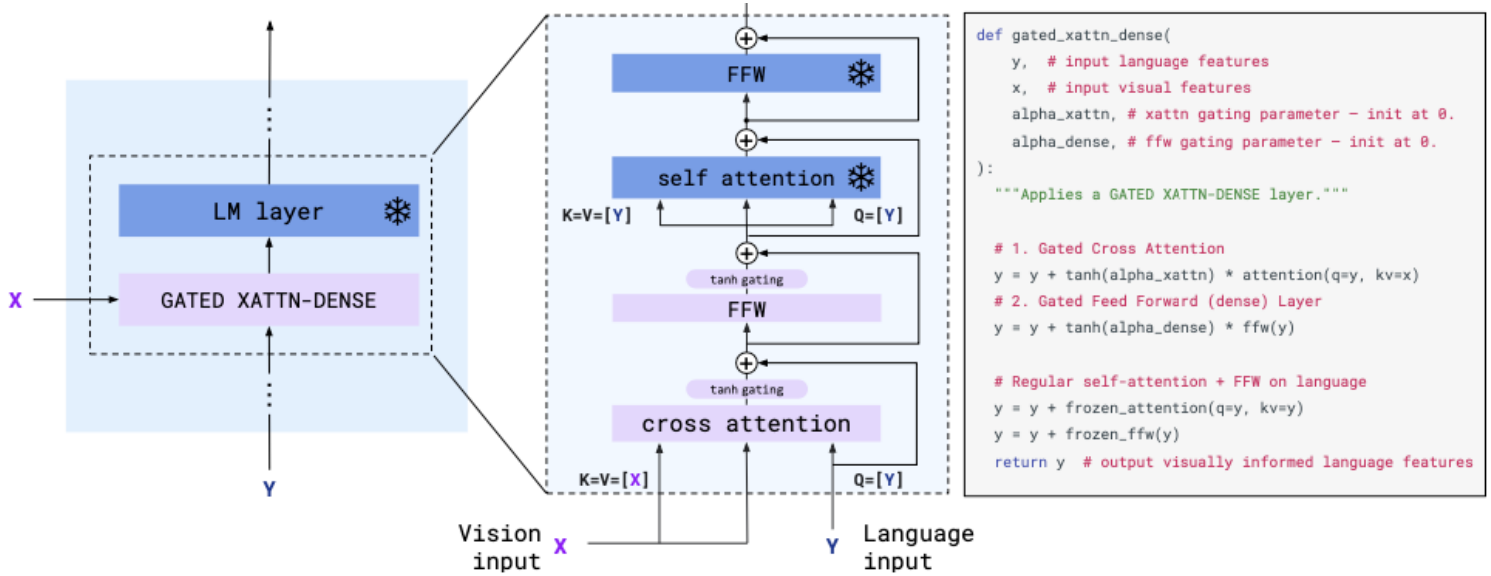



Figure 4: **GATED XATTN-DENSE layers.** To condition the LM on visual inputs, we insert new cross-attention layers between existing pretrained and frozen LM layers. The keys and values in these layers are obtained from the vision features while the queries are derived from the language inputs.

How are modalities combined?

- Pretrained LM blocks are frozen
- *Gated cross-attention dense* blocks are inserted between these layers and trained from scratch

# Vision–Language Models: LLaVA-1.5/LLaVA-NeXT

[Liu et al., 2023b, Liu et al., 2023a, Liu et al., 2024]

## Instruction-tuning for Vision–Language Models

- Goal: Create a general-purpose instruction-following visual agent

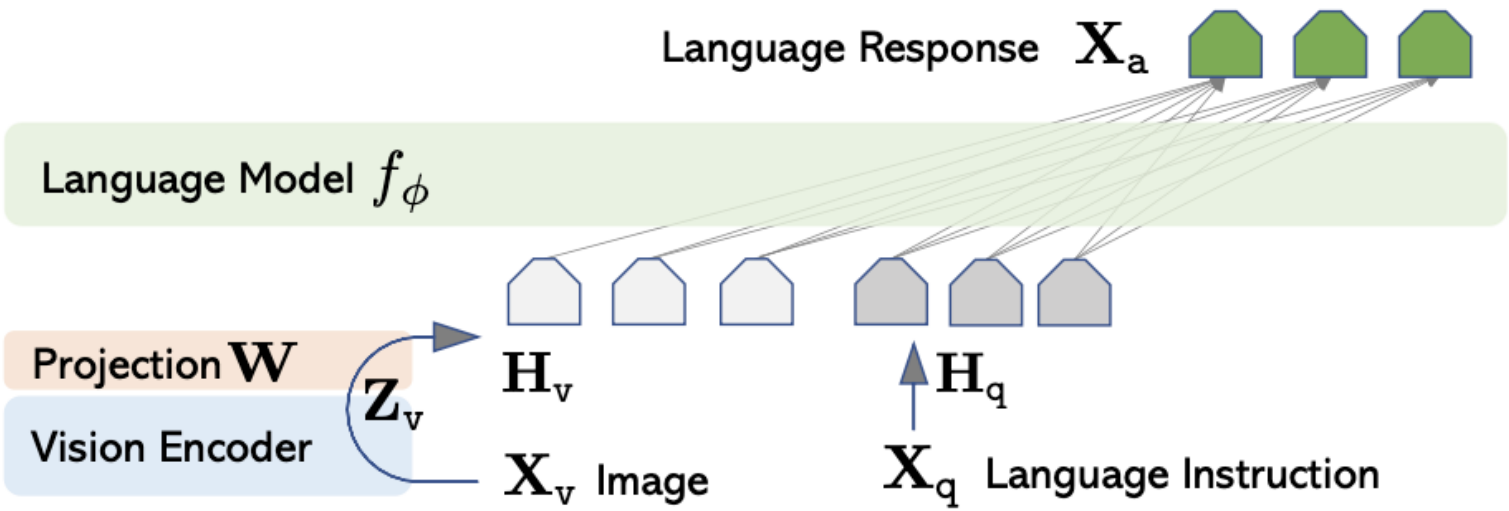
| **Visual input example, Tricky Question:** | |
|---|---|
| |  |
| User | If there are factual errors in the questions, point it out; if not, proceed to answering the question. What's happening in the desert? |
| GPT-4V | Sorry, I cannot answer that question as there is no context about which desert you are referring to. |
| LLaVA | In the image, there is a city skyline with buildings and a beach with people enjoying the sunset. The scene is set in a desert, which is unusual for a beach setting. |
| LLaVA-1.5 | There are no deserts in the image. The image features a beach with palm trees, a city skyline, and a large body of water. |

# Vision–Language Models: LLaVA-1.5/LLaVA-NeXT

[Liu et al., 2023b, Liu et al., 2023a, Liu et al., 2024]

## Instruction-tuning for Vision–Language Models

- Goal: Create a general-purpose instruction-following visual agent
- Idea: Connect a visual encoder with a language model (decoder/encoder), train it on instructional vision–language data



# Vision-Language Models: LLaVA-1.5/LLaVA-NeXT

[Liu et al., 2023b, Liu et al., 2023a, Liu et al., 2024]

## How to obtain instructional vision–language data?

- Use GPT-4/ChatGPT to **create instruction-following data** with **visual content** from an image–caption dataset ([CC3M])
- Encode the image in order to prompt text-only GPT: captions and bounding boxes (object class + location)

# Vision-Language Models: LLaVA-1.5/LLaVA-NeXT

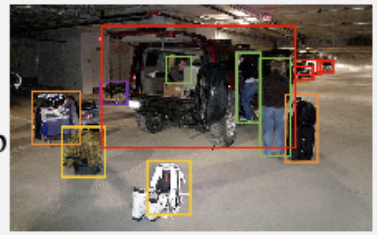
[Liu et al., 2023b, Liu et al., 2023a, Liu et al., 2024]

## How to obtain instructional vision–language data?

- Use GPT-4/ChatGPT to **create instruction-following data** with **visual content** from an image–caption dataset ([CC3M])
- Encode the image in order to prompt text-only GPT: captions and bounding boxes (object class + location)

**Context type 1: Captions**

A group of people standing outside of a black vehicle with various luggage.
Luggage surrounds a vehicle in an underground parking area
People try to fit all of their luggage in an SUV.
The sport utility vehicle is parked in the public garage, being packed for a trip
Some people with luggage near a van that is transporting it.



**Context type 2: Boxes**

person: [0.681, 0.242, 0.774, 0.694], backpack: [0.384, 0.696, 0.485, 0.914], suitcase: ...<omitted>

---

**Response type 1: conversation**

Question: What type of vehicle is featured in the image?
Answer: The image features a black sport utility vehicle (SUV) ...<omitted>

**Response type 2: detailed description**

The image is an underground parking area with a black sport utility vehicle (SUV) parked. There are three people in the scene, with one person standing closer to the left side of the vehicle, another person in the middle, and the third person on the right side. They are all working together to pack their luggage into the SUV for a trip. ...<omitted>

**Response type 3: complex reasoning**

Question: What challenges do these people face?

# Vision-Language Models: LLaVA-1.5/LLaVA-NeXT

[Liu et al., 2023b, Liu et al., 2023a, Liu et al., 2024]

## Prompt to GPT-4 to create instruction-following data

```
messages = [ {"role":"system", "content": f"""You are an AI visual assistant, and you are
seeing a single image. What you see are provided with five sentences, describing the same image you
are looking at. Answer all questions as you are seeing the image.

Design a conversation between you and a person asking about this photo. The answers should be in a
tone that a visual AI assistant is seeing the image and answering the question. Ask diverse questions
and give corresponding answers.

Include questions asking about the visual content of the image, including the object types, counting
the objects, object actions, object locations, relative positions between objects, etc. Only include
questions that have definite answers:
(1) one can see the content in the image that the question asks about and can answer confidently;
(2) one can determine confidently from the image that it is not in the image. Do not ask any question
that cannot be answered confidently.

Also include complex questions that are relevant to the content in the image, for example, asking
about background knowledge of the objects in the image, asking to discuss about events happening in
the image, etc. Again, do not ask about uncertain details. Provide detailed answers when answering
complex questions. For example, give detailed examples or reasoning steps to make the content more
convincing and well-organized. You can include multiple paragraphs if necessary."""}
]
for sample in fewshot_samples:
    messages.append({"role":"user", "content":sample['context']})
    messages.append({"role":"assistant", "content":sample['response']} )
messages.append({"role":"user", "content":'\n'.join(query)})
```

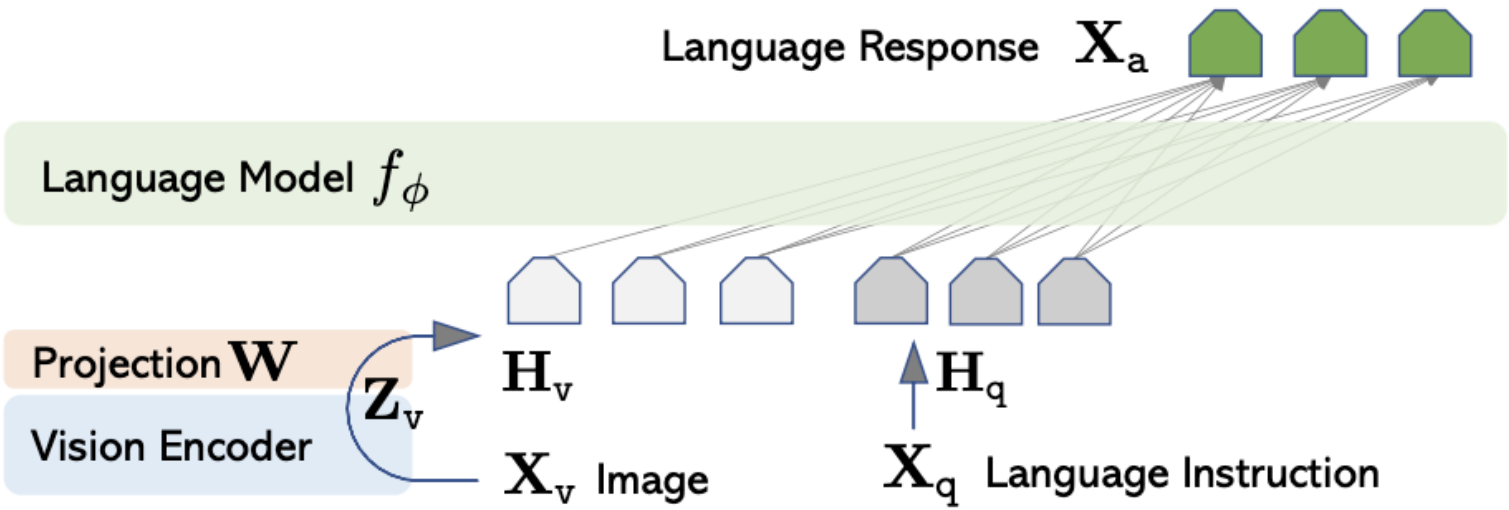
Table 13: For each query, we illustrate the prompt construction process for ChatGPT/GPT-4 to collect `query['response']` from `query['context']`, using few-shot in-context-learning, where examples are from `fewshot_samples`, each example including input `sample['context']` and output `sample['response']`. Note that `messages` is the final prompt. In this example, we provide

# Vision–Language Models: LLaVA-1.5/LLaVA-NeXT

[Liu et al., 2023b, Liu et al., 2023a, Liu et al., 2024]

## Training LLaVA: Two stage process

1. Pre-train for feature alignment: Model is instructed to generate a description for the image (target: original caption) $\Rightarrow$ train visual tokeniser
2. Fine-tune projection layer *and LLM* towards instruction-following agent abilities



# Vision–Language Models

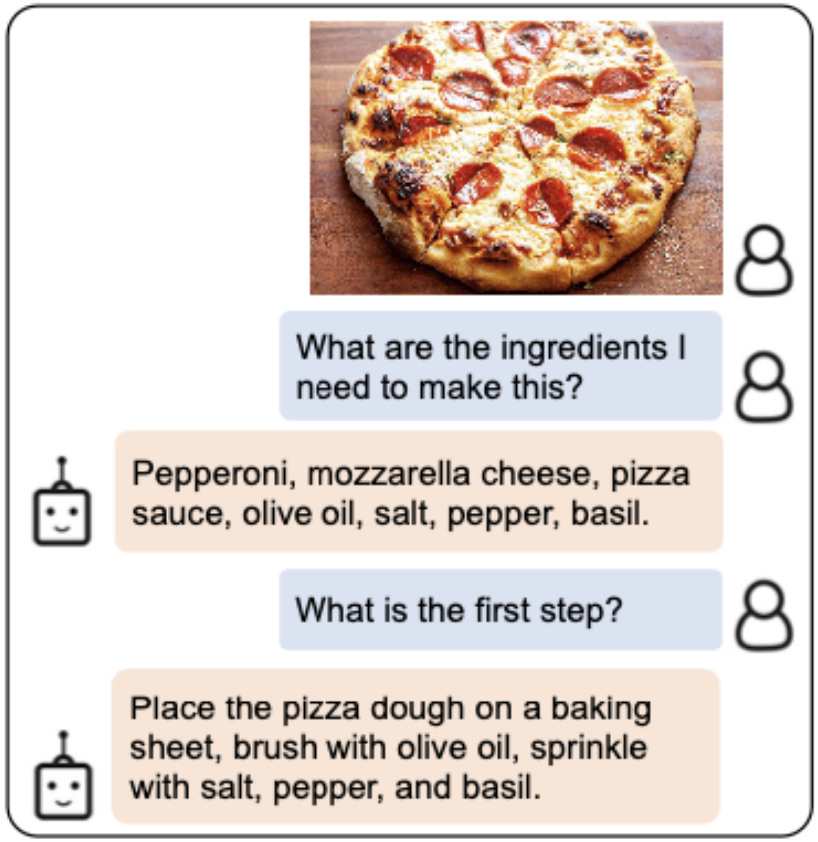
## Instruction-tuned Models

- LLaVA: First to propose instruction-tuning for vision–language models
- Flamingo and BLIP-2: More sophisticated schemes to obtain vision–language representations, not instruction-tuned
- BLIP-2/InstructBLIP: Both visual encoder and language model are kept frozen

# Vision–Language Models: BLIP-2, InstructBLIP

[Li et al., 2023, Dai et al., 2023]

- BLIP-2: a zero-shot image-to-text generation method from which static (task-agnostic) visual features can be extracted
- Examples:



# Vision–Language Models: BLIP-2, InstructBLIP

[Li et al., 2023, Dai et al., 2023]

- BLIP-2: a zero-shot image-to-text generation method from which static (task-agnostic) visual features can be extracted
- Examples:

# Vision–Language Models: BLIP-2, InstructBLIP

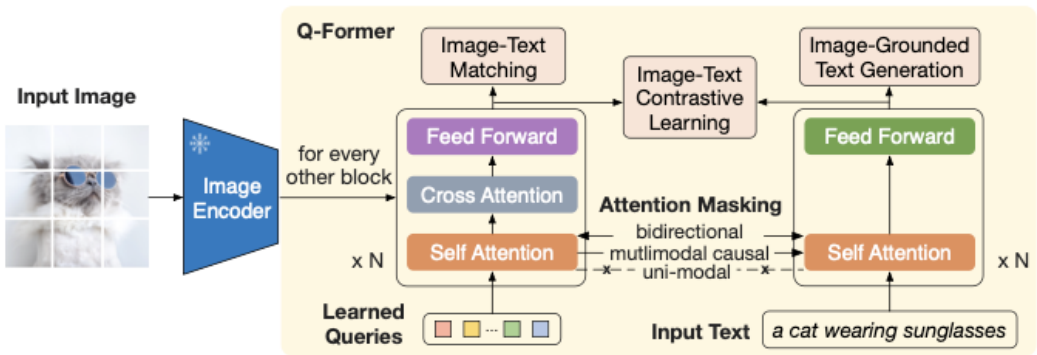
[Li et al., 2023, Dai et al., 2023]

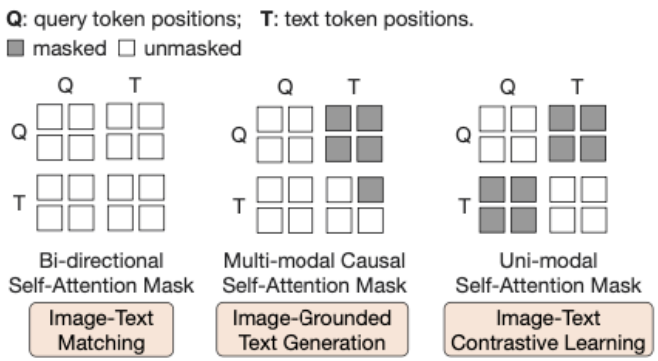
- BLIP-2: a zero-shot image-to-text generation method from which static (task-agnostic) visual features can be extracted
- InstructBLIP: allows instruction-aware visual feature extraction, i.e., features are conducive to the task at hand

# Vision–Language Models: BLIP-2, InstructBLIP

[Li et al., 2023, Dai et al., 2023]

- BLIP-2: a zero-shot image-to-text generation method from which static (task-agnostic) visual features can be extracted
- InstructBLIP: allows instruction-aware visual feature extraction, i.e., features are conducive to the task at hand
- How? Q-Former (Query Transformer) is used to extract visual features from a frozen image encoder

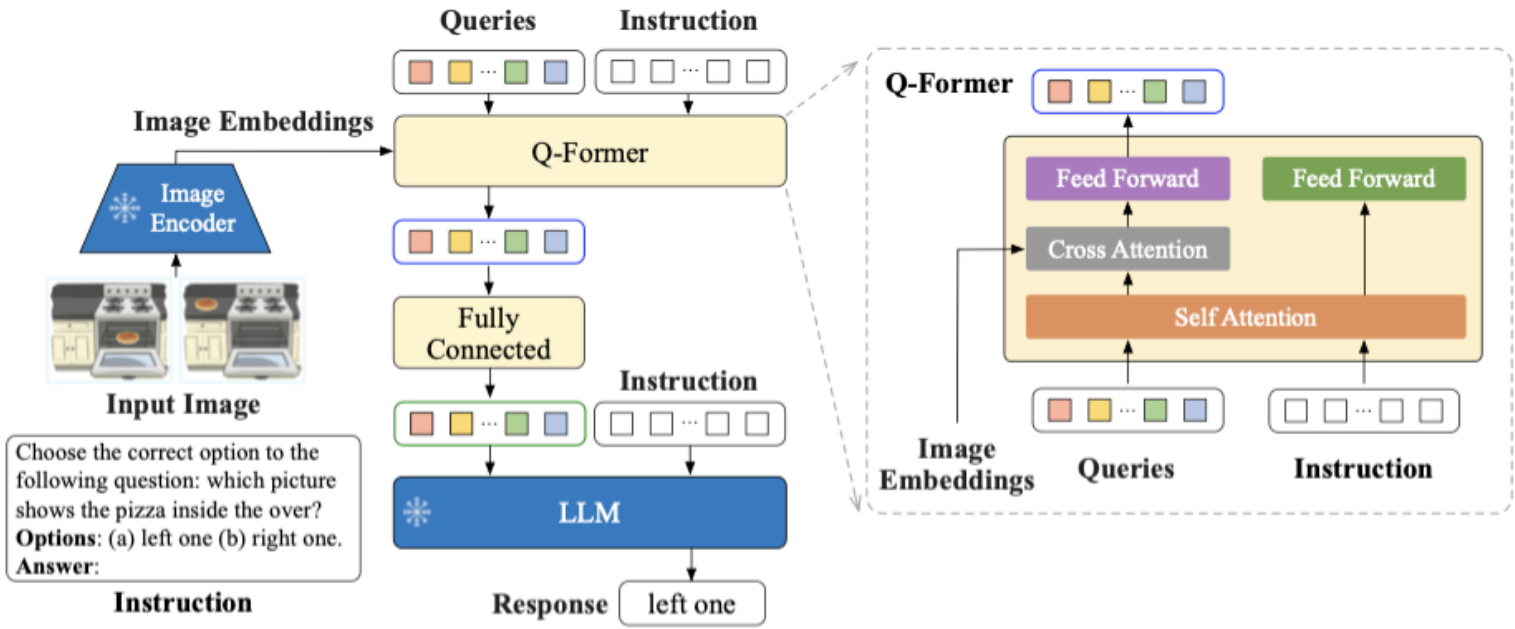


# Vision–Language Models: BLIP-2, InstructBLIP

[Li et al., 2023, Dai et al., 2023]

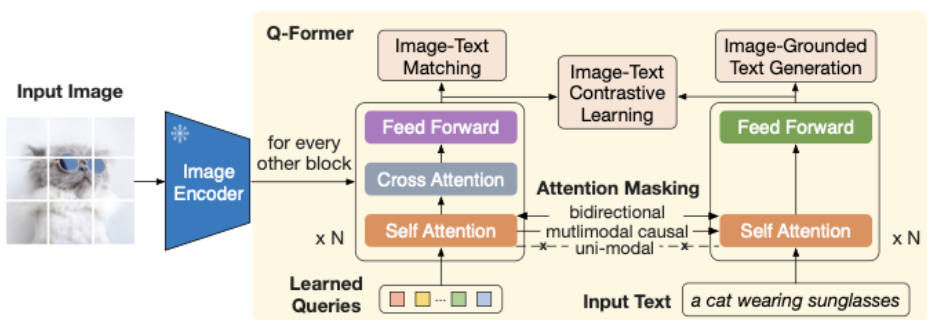
- BLIP-2: a zero-shot image-to-text generation method from which static (task-agnostic) visual features can be extracted
- InstructBLIP: allows instruction-aware visual feature extraction, i.e., features are conducive to the task at hand
- How? Q-Former (Query Transformer) is used to extract visual features from a frozen image encoder
- BLIP-2's Q-Former receives the queries as input, while InstructBLIP's Q-Former receives also the instruction as input

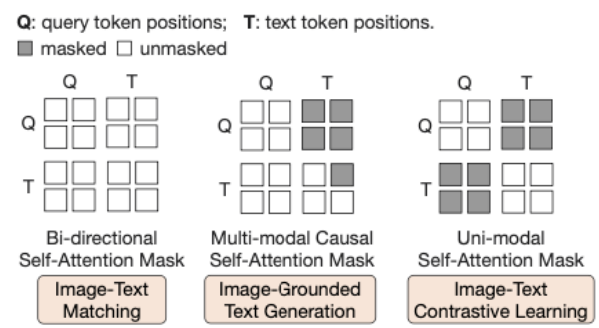


# Vision–Language Models: BLIP-2, InstructBLIP

[Li et al., 2023, Dai et al., 2023]

- BLIP-2: a zero-shot image-to-text generation method from which static (task-agnostic) visual features can be extracted
- Idea: Use Q-Former as a bridge between *frozen LLM and frozen visual encoder*
- Q-Former (Query Transformer) is pre-trained in two stages:

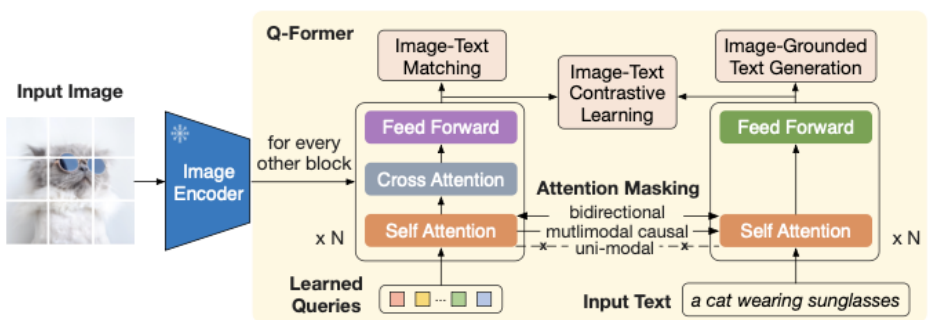


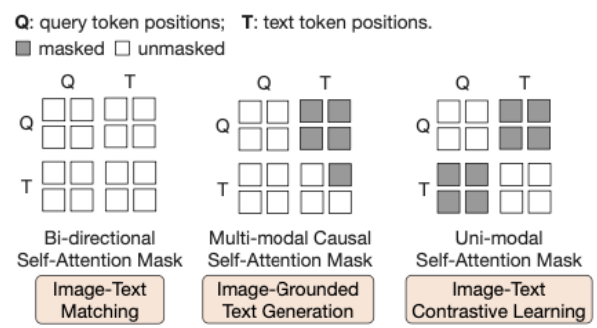



# Vision–Language Models: BLIP-2, InstructBLIP

[Li et al., 2023, Dai et al., 2023]

- BLIP-2: a zero-shot image-to-text generation method from which static (task-agnostic) visual features can be extracted
- Idea: Use Q-Former as a bridge between *frozen LLM and frozen visual encoder*
- Q-Former (Query Transformer) is pre-trained in two stages:
  1. perform vision-language representation learning (visual features shall be most relevant to the text)
     3 objectives: image–text contrastive learning, image-grounded text generation, image–text matching

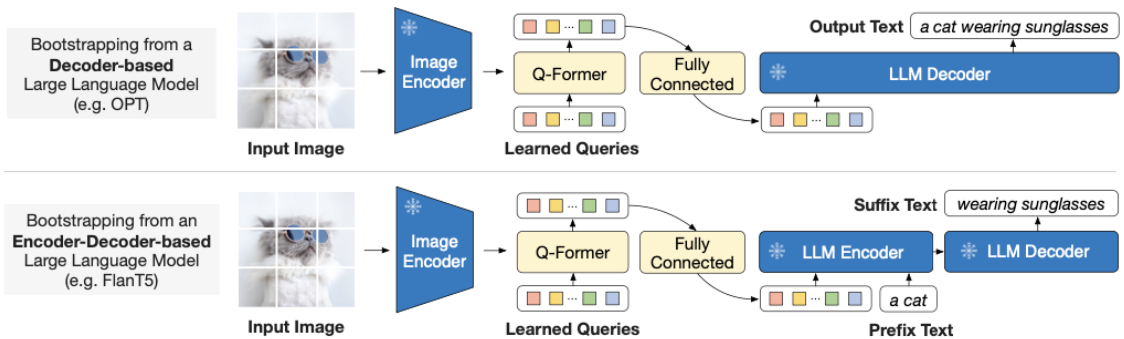


# Vision–Language Models: BLIP-2, InstructBLIP

[Li et al., 2023, Dai et al., 2023]

- BLIP-2: a zero-shot image-to-text generation method from which static (task-agnostic) visual features can be extracted
- Idea: Use Q-Former as a bridge between *frozen LLM and frozen visual encoder*
- Q-Former (Query Transformer) is pre-trained in two stages:
  1. perform vision-language representation learning (visual features shall be most relevant to the text)
     3 objectives: image–text contrastive learning, image-grounded text generation, image–text matching
  2. perform vision-to-language generation learning (the visual output of the Q-Former shall be interpretable by the frozen LLM to generate textual output)



# Further References: Visually Grounded Representations

## Surveys

- [Xu et al., 2023] Multimodal Learning with Transformers: A Survey
- [Chen et al., 2022] VLP: A Survey on Vision-Language Pre-training
- [Agrawal et al., 2022] Vision-Language Pretraining: Current Trends and the Future
- [Du et al., 2022] A Survey of Vision-Language Pre-Trained Models
- [Uppal et al., 2022] Multimodal Research in Vision and Language: A Review of Current and Emerging Trends

## Videos and Language

- Models: VideoBERT, CBT, HERO, and many more
- Survey on Videos+Language: [Ruan and Jin, 2022]

# References I


Agrawal, A., Teney, D., and Nematzadeh, A. (2022).

Vision-language pretraining: Current trends and the future.

In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics: Tutorial Abstracts*, pages 38–43, Dublin, Ireland. Association for Computational Linguistics.

Alayrac, J.-B., Donahue, J., Luc, P., Miech, A., Barr, I., Hasson, Y., Lenc, K., Mensch, A., Millican, K., Reynolds, M., et al. (2022a).

Flamingo: a visual language model for few-shot learning.

*Advances in Neural Information Processing Systems*, 35:23716–23736.

Alayrac, J.-B., Donahue, J., Luc, P., Miech, A., Barr, I., Hasson, Y., Lenc, K., Mensch, A., Millican, K., Reynolds, M., Ring, R., Rutherford, E., Cabi, S., Han, T., Gong, Z., Samangooei, S., Monteiro, M., Menick, J., Borgeaud, S., Brock, A., Nematzadeh, A., Sharifzadeh, S., Binkowski, M., Barreira, R., Vinyals, O., Zisserman, A., and Simonyan, K. (2022b).

Flamingo: a visual language model for few-shot learning.

*ArXiv*, abs/2204.14198.

Awadalla, A., Gao, I., Gardner, J., Hessel, J., Hanafy, Y., Zhu, W., Marathe, K., Bitton, Y., Gadre, S., Sagawa, S., Jitsev, J., Kornblith, S., Koh, P. W., Ilharco, G., Wortsman, M., and Schmidt, L. (2023).

Openflamingo: An open-source framework for training large autoregressive vision-language models.

*arXiv preprint arXiv:2308.01390*.

Beinborn, L., Botschen, T., and Gurevych, I. (2018).

Multimodal grounding for language processing.

In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 2325–2339. Association for Computational Linguistics.

# References II


Chen, F., Zhang, D., Han, M., Chen, X., Shi, J., Xu, S., and Xu, B. (2022).
Vlp: A survey on vision-language pre-training.
*arXiv preprint arXiv:2202.09061*.

Dai, W., Li, J., Li, D., Tiong, A. M. H., Zhao, J., Wang, W., Li, B., Fung, P., and Hoi, S. (2023).
InstructBLIP: Towards general-purpose vision-language models with instruction tuning.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., and Houlsby, N. (2021).
An image is worth 16x16 words: Transformers for image recognition at scale.
In *International Conference on Learning Representations*.

Du, Y., Liu, Z., Li, J., and Zhao, W. X. (2022).
A Survey of Vision-Language Pre-Trained Models.

Ghosh, A., Acharya, A., Saha, S., Jain, V., and Chadha, A. (2024).
Exploring the frontier of vision-language models: A survey of current methodologies and future directions.
*arXiv preprint arXiv:2404.07214*.

Jia, C., Yang, Y., Xia, Y., Chen, Y.-T., Parekh, Z., Pham, H., Le, Q., Sung, Y.-H., Li, Z., and Duerig, T. (2021).
Scaling up visual and vision-language representation learning with noisy text supervision.
In *International Conference on Machine Learning*, pages 4904–4916. PMLR.

Kim, W., Son, B., and Kim, I. (2021).
Vilt: Vision-and-language transformer without convolution or region supervision.
In Meila, M. and Zhang, T., editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 5583–5594. PMLR.

# References III



Li, J., Li, D., Savarese, S., and Hoi, S. (2023).
BLIP-2: bootstrapping language-image pre-training with frozen image encoders and large language models.
In *ICML*.


Liu, H., Li, C., Li, Y., and Lee, Y. J. (2023a).
Improved baselines with visual instruction tuning.


Liu, H., Li, C., Li, Y., Li, B., Zhang, Y., Shen, S., and Lee, Y. J. (2024).
Llava-next: Improved reasoning, ocr, and world knowledge.


Liu, H., Li, C., Wu, Q., and Lee, Y. J. (2023b).
Visual instruction tuning.


Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al. (2021a).
Learning transferable visual models from natural language supervision.
In *International conference on machine learning*, pages 8748–8763. PMLR.


Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., Krueger, G., and Sutskever, I. (2021b).
Learning Transferable Visual Models From Natural Language Supervision.
*CoRR*, abs/2103.00020.


Ruan, L. and Jin, Q. (2022).
Survey: Transformer based video-language pre-training.
*AI Open*, 3:1–13.

# References IV


Singh, A., Hu, R., Goswami, V., Couairon, G., Galuba, W., Rohrbach, M., and Kiela, D. (2022).
Flava: A foundational language and vision alignment model.
In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15638–15650.

Uppal, S., Bhagat, S., Hazarika, D., Majumder, N., Poria, S., Zimmermann, R., and Zadeh, A. (2022).
Multimodal research in vision and language: A review of current and emerging trends.
*Inf. Fusion*, 77(C):149–171.

Xu, P., Zhu, X., and Clifton, D. A. (2023).
Multimodal learning with transformers: A survey.
*IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(10):12113–12132.