CS-461

Foundation Models and Generative AI

Multimodality in Foundation Models

Charlotte Bunne, Fall Semester 2025/26

Announcements

- Next week we have a guest lecture!

Adaptation, Fine-Tuning, and Test-Time Training

Andreas Krause
ETHZ

Jonas Hübotter
ETHZ

- Lecture as usual **in PO 01 on Tuesday, 1 pm**

- Next week, **Assignment 2** on test-time learning will be online!

  Deadline: Wednesday, December 3 at 23:59.

2 Weeks!

Feedback from Evaluation

Lecture

"Amazing course in terms of lectures and guest lectures… The slides are great."

Exercises

"The exercises and labs are really hard, I was expecting more assistants guiding us."

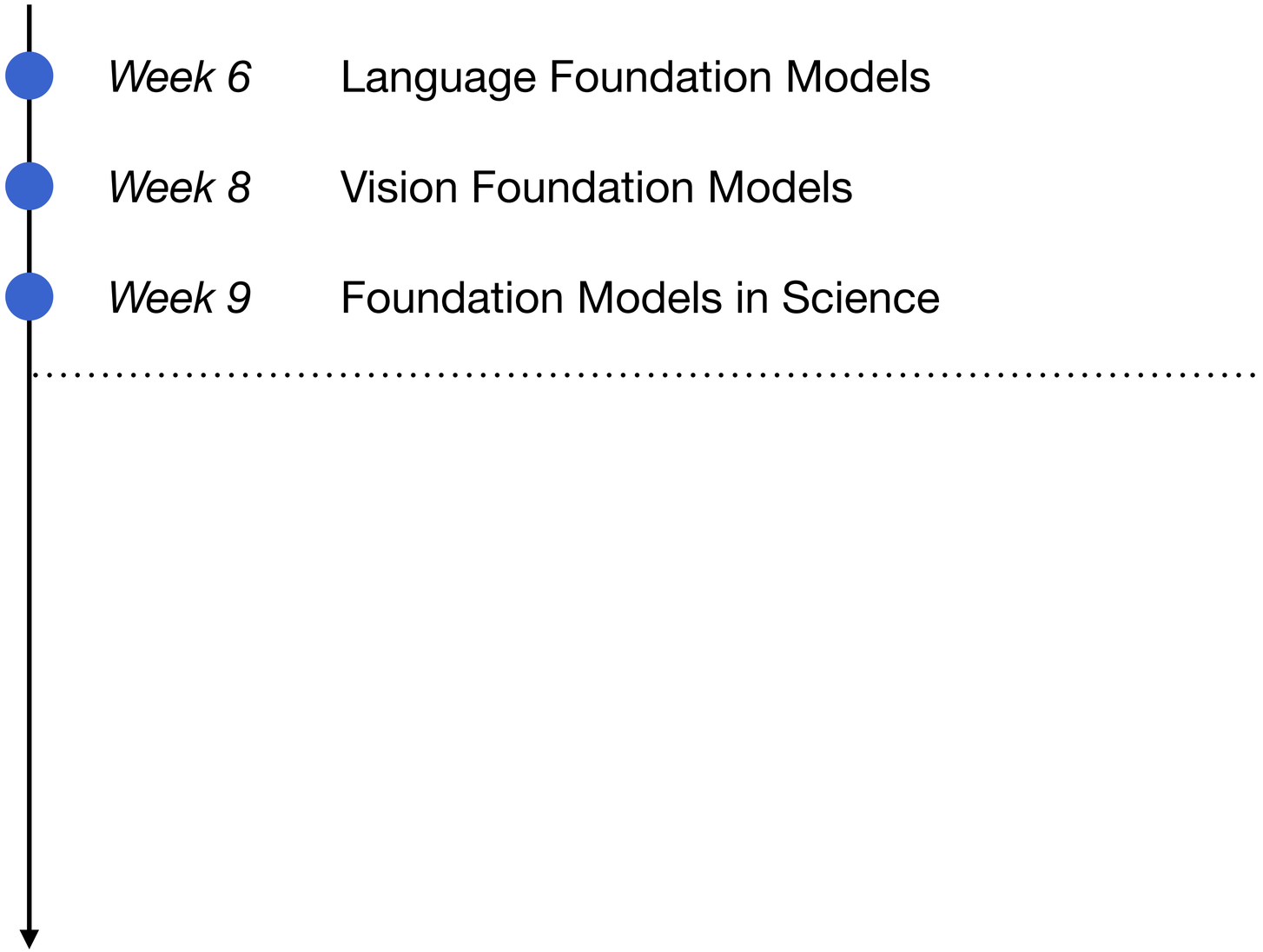
"The exercises, although some of them extremely difficult and long, help provide a very deep understand."

"There is no TA session where you can raise your hand and ask specific questions directly to a TA."

Assignments

"The graded assignment 1 was a bit frustrating because it was really open ended, and not really about us applying concepts from the course but rather trial an error things."

Where are We?


Week 6 Language Foundation Models
Week 8 Vision Foundation Models
Week 9 Foundation Models in Science

Week 8's Exercise Sheet

Exercise 6 · Task 1

Differences among BERT, T5, and GPT Models.

Compare the three Transformer families by architecture, positional encoding, and training objective. Explain how encoder-only, encoder-decoder, and decoder-only models differ in attention flow and masking; outline each model's positional encoding and its trade-offs; and summarize their objectives, i.e., BERT's masked LM, T5's span corruption with sentinels, and GPT's autoregressive prediction.

Week 8's Code Demonstration

Code Notebook 6 · Task 1

Implementation of BERT, T5, and GPT

This exericise walks through the architecture and training pipelines for BERT, T5 and GPT models. Concretely, it covers encoder, decoder, encoder-decoder architecture implementations, their different training objectives with inputs and targets construction.

→ Jupyter notebook exercise

Week 9’s Exercise Sheet

Exercise 7 · Task 1 & 2

MAE: Concept and Optimal Masking Ratios

Understand how masked pretraining adapts from language to vision, why MAE prevailed against alternative pretext tasks, and how different masking strategies affect the learned representations. Analyse through information theory why BERT-style models and MAE have different optimal masking ratios.

Exercise 7 · Task 3 & 4

iBOT: Concept and Mode Collapse Prevention

Compare iBOT's latent space versus MAE's pixel-level reconstruction, explain how iBOT was adapted from BERT for images, and analyse how iBOT's student-teacher framework differs from BYOL. Understand the mode collaps and how it can be prevented through teacher representation centering.

Week 9's Code Demonstration

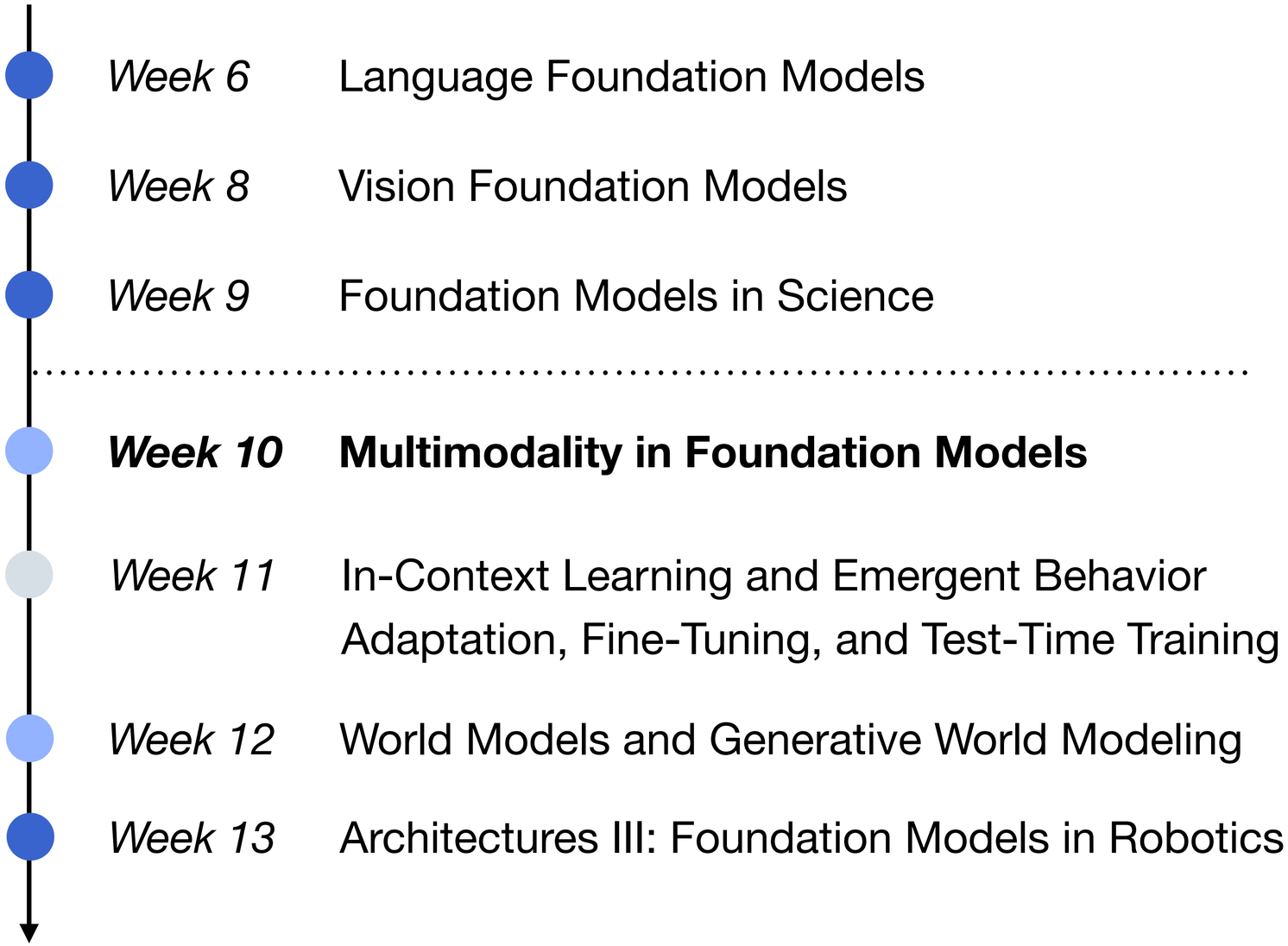
Code Notebook 7 · Task 5

Implementation, Training and PCA evaluation of DINOv2

This exercise walks through the architecture and training pipeline of DINOv2, analyses the convergence behaviour of the different loss parts and evaluates the captured semantics in the patch tokens through PCA.

→ Jupyter notebook exercise

Where are We?


Week 6 Language Foundation Models
Week 8 Vision Foundation Models
Week 9 Foundation Models in Science
Week 10 **Multimodality in Foundation Models**
Week 11 In-Context Learning and Emergent Behavior
Adaptation, Fine-Tuning, and Test-Time Training
Week 12 World Models and Generative World Modeling
Week 13 Architectures III: Foundation Models in Robotics

What is Multimodality?

Modality

A distinct type or format of data that represents information through a specific sensory channel or structured representation.

Goal of Multimodal Learning:

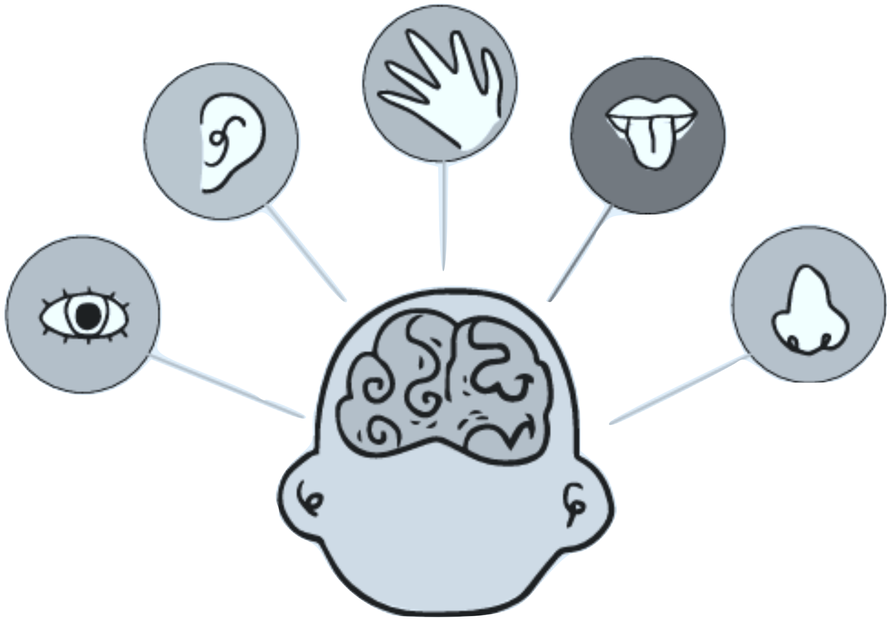
Learn a model that allows understanding, reasoning, or generation across modalities.

Multimodal systems integrate multiple modalities to leverage complementary information from different data types.

Unimodal Learning

Multimodal Learning

Why Combine Modalities?



Complementarity:
Each modality provides a unique view of the world.

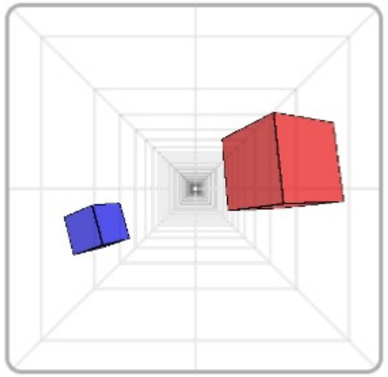
Disambiguation:
Combining modalities reduces uncertainty.

Generalization:
Joint learning across modalities improves transfer to new domains and tasks.

Human analogy:
Humans integrate sight, sound, and language seamlessly; multimodal AI aims to do the same.

Examples of Multimodal Models: Vision-Language Models

Visual Prompt



 # Abstract Question

Which object is located on the left side, the **blue** cube or the **red** cube?

: The **blue** cube is located on the **left** side of the **red** cube

 # Object Abstraction

We provide a color-object map that maps each colored box to an object:

 # Color-Object Map

- **blue** cube → **snowman**
- **red** cube → **horse**

Given the **color-object map** above, change all the **colored boxes** in their respective **objects**.

[Original Response]: The **blue** cube is located on the **left** side of th

: The **snowman** is located on the **left** side of the **horse**.

Based on the previous response, answer the question: **{ Question }**

Allocentric Question

Question:
Given the image, consider the real-world 3D locations and orientations of the objects. If you stand at the **man's** position facing where he's facing, Is the dog on the **left** or **right** of the man?

Answer:

✔ Human: "on the **right**"

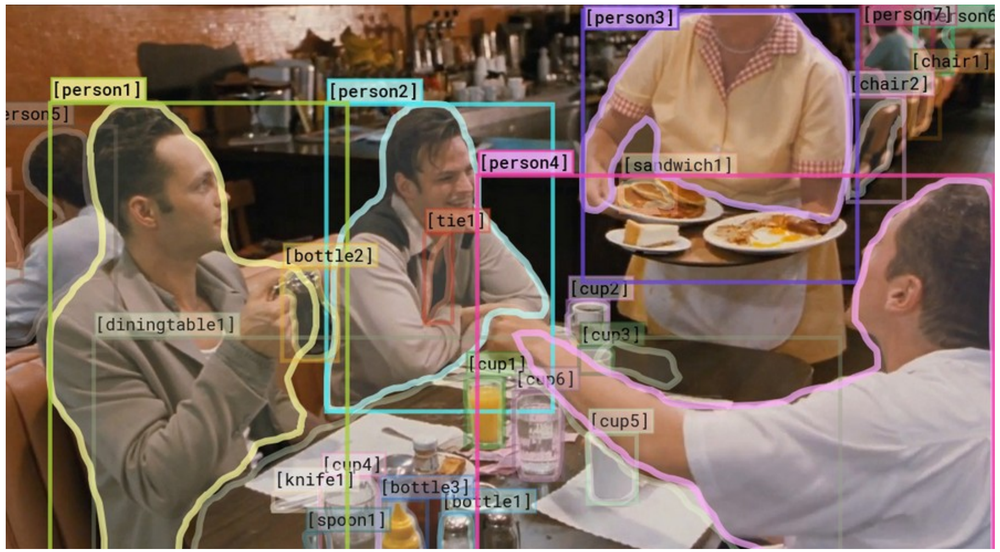
✘ VLM: "on the **left**"

Examples of Multimodal Models: Vision-Language Models

"An astronaut riding a horse in a photorealistic style."

Disclaimer!

We will use vision-language models as running example of today's lecture but principles translate to other modalities and domains.



Why is [person4] pointing at [person1]?

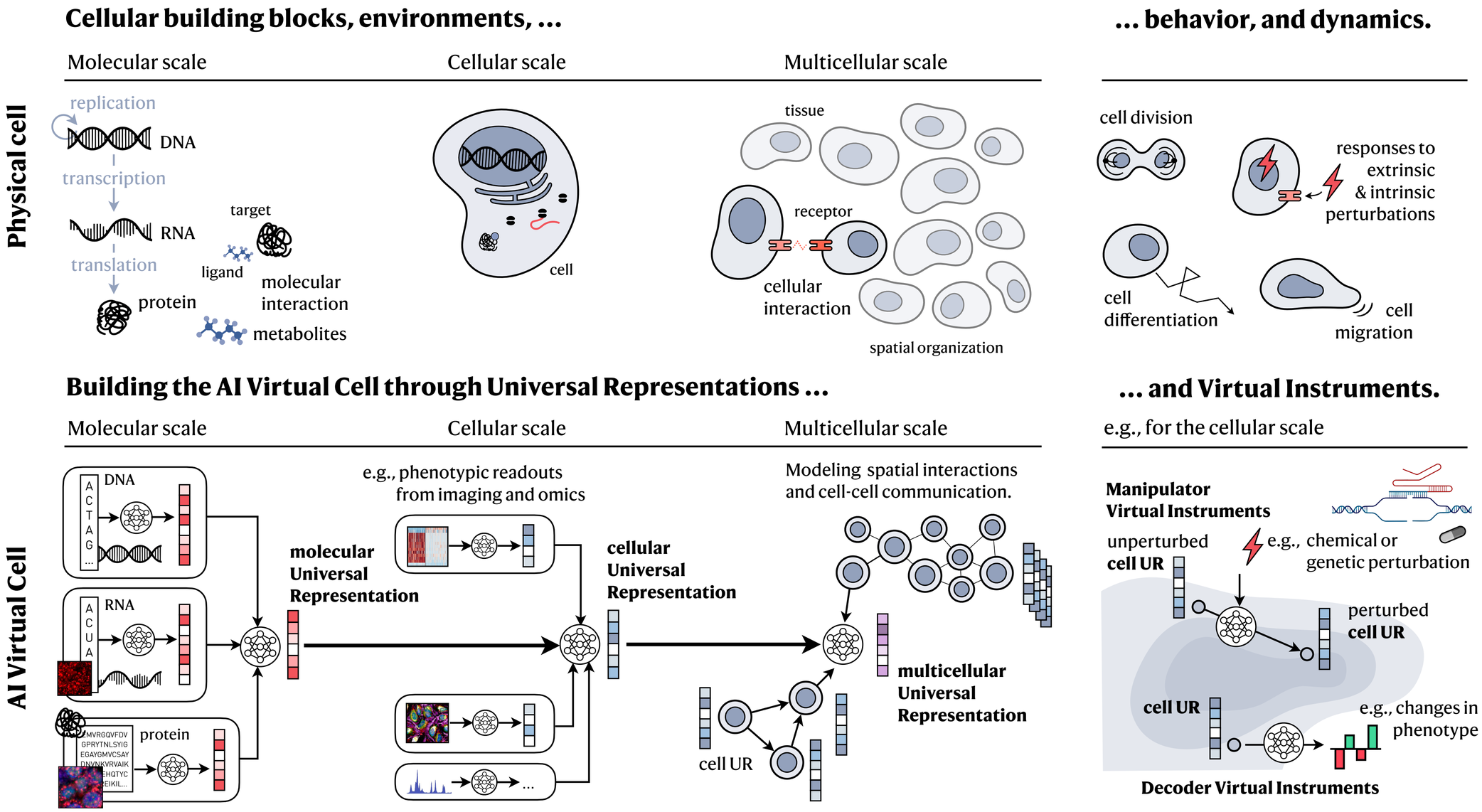
a) He is telling [person3] that [person1] ordered the pancakes.
b) He just told a joke.
c) He is feeling accusatory towards [person1].
d) He is giving [person1] directions.

*I chose **a)** because…*

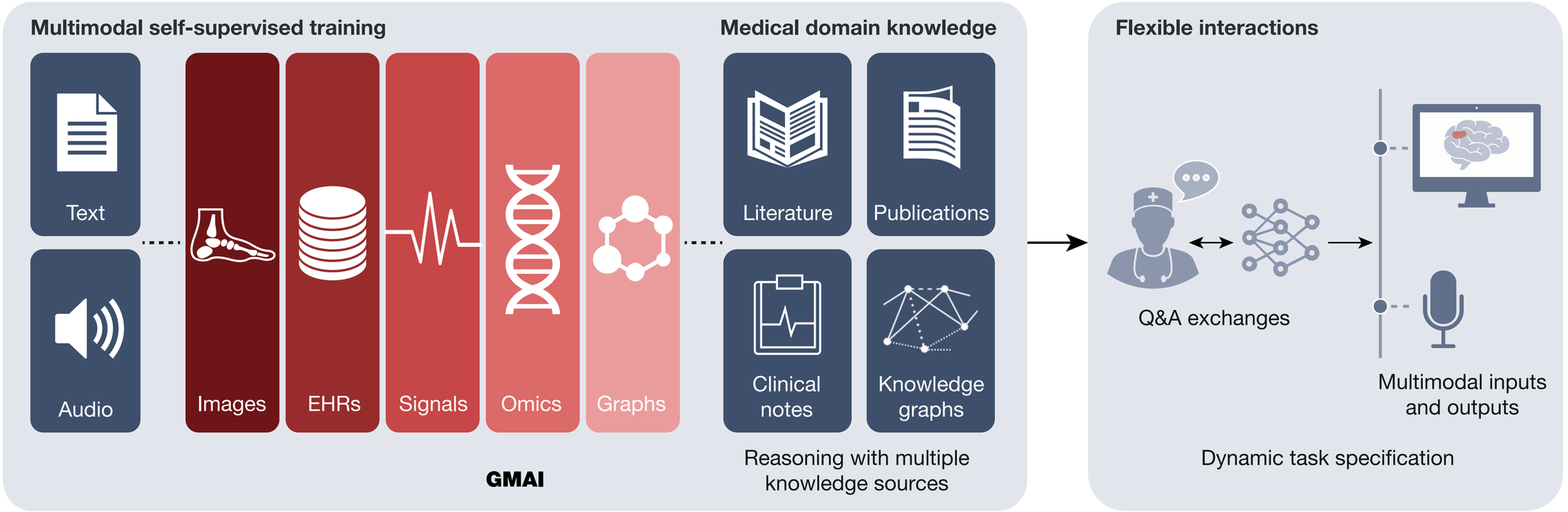
a) [person1] has the pancakes in front of him.
b) [person4] is taking everyone's order and asked for clarification.
c) [person3] is looking at the pancakes and both she and [person2] are smiling slightly.
d) [person3] is delivering food to the table, and she might not know whose order is whose.

Examples of Multimodal Models: Audio-Vision Language Models

Examples of Multimodal Models in Biology (Bunne et al., 2024)



Examples of Multimodal Models in Medicine

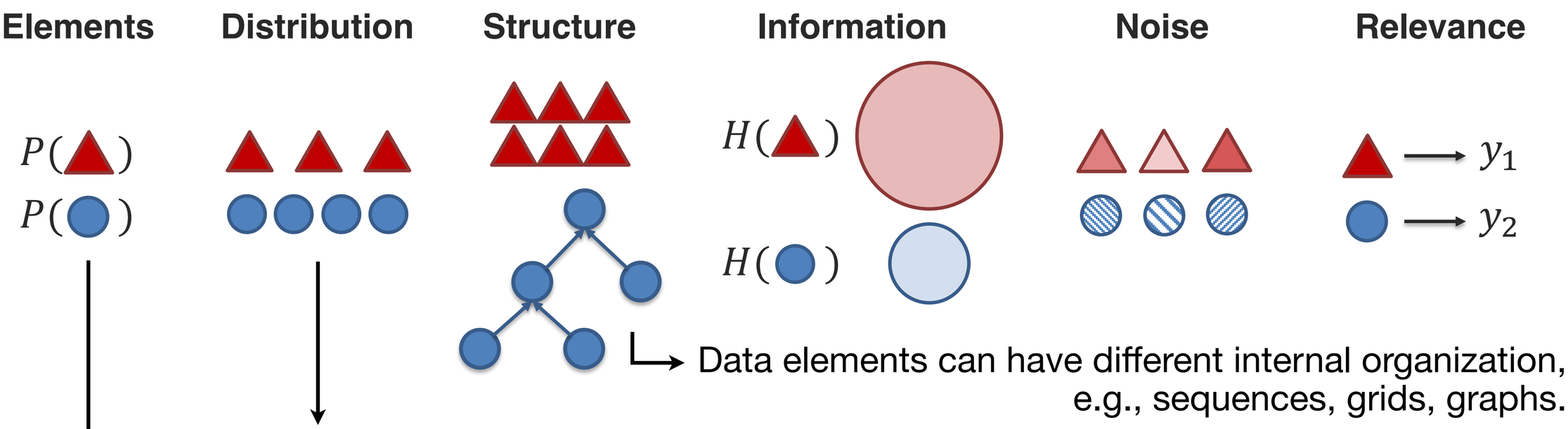


(Moor et al., 2023)

Multimodal Machine Learning: Dimensions of Heterogeneity

Multimodal learning must handle heterogeneity:

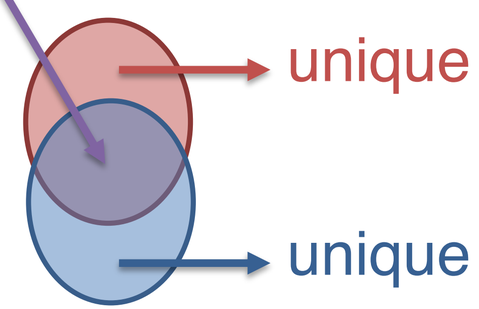
Differences across modalities in *distribution*, *structure*, and *informativeness*, etc.

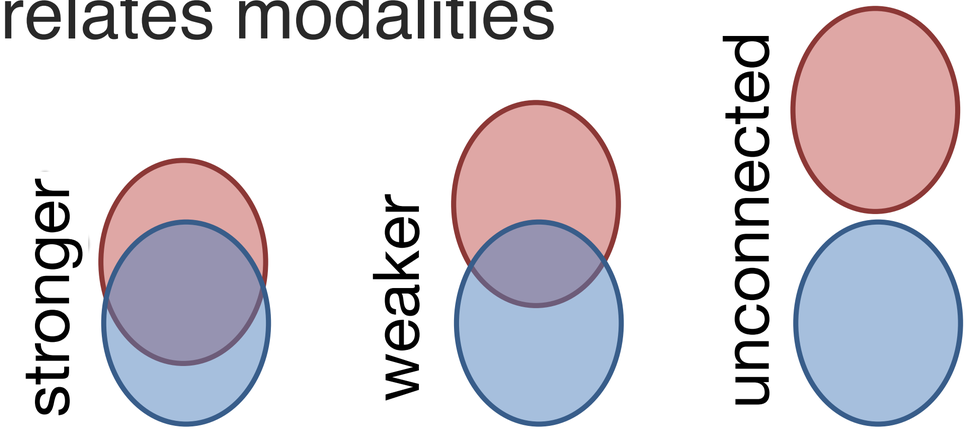


(Liang et al., 2024)

Multimodal Machine Learning: Modality Connections

Connections: Shared information that relates modalities





Statistical **Semantic**

Association Dependency Correspondence Relationship

e.g., correlation, co-occurrence

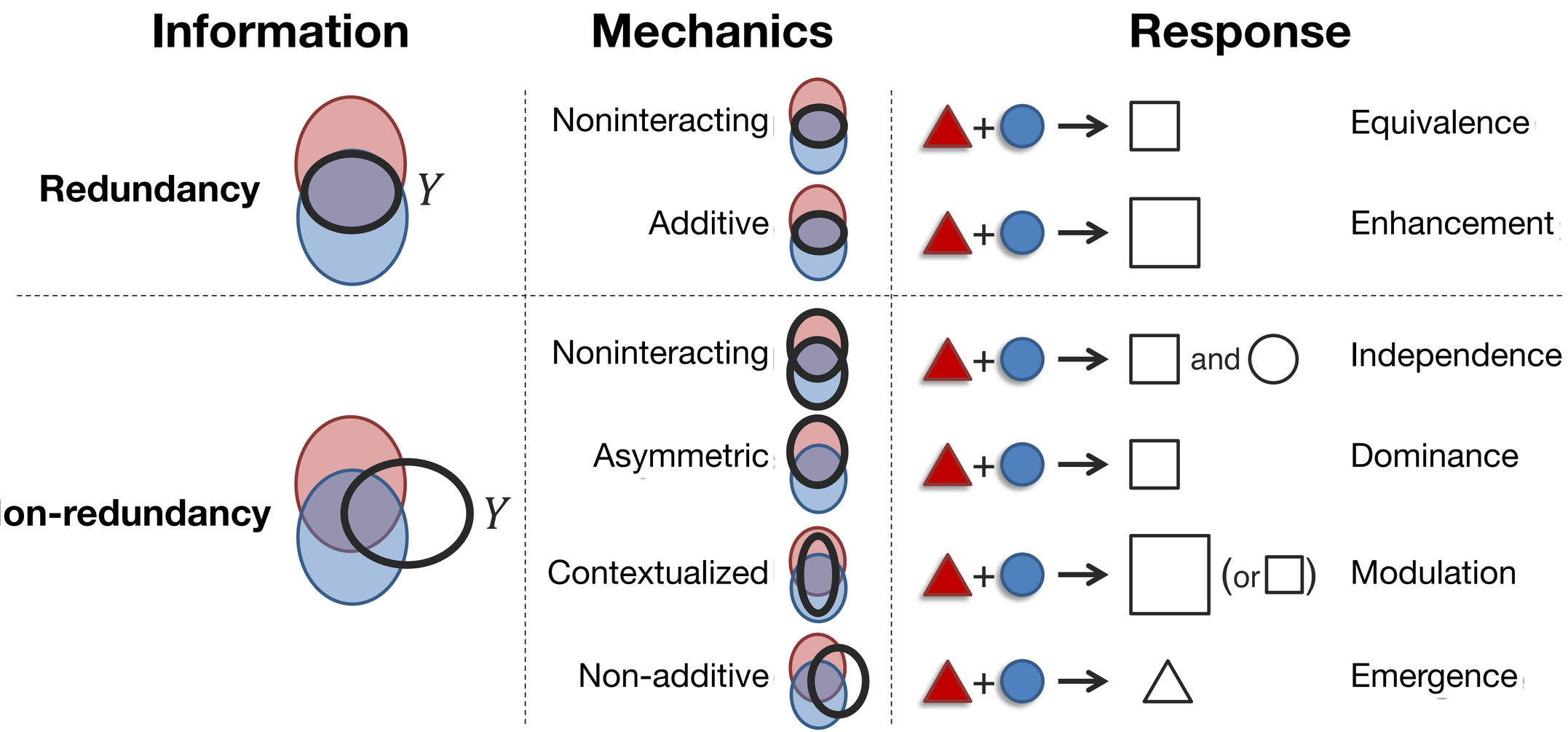
e.g., causal, temporal

e.g., grounding

e.g., function

(Liang et al., 2024)

Multimodal Machine Learning: Modality Interactions



(Liang et al., 2024)

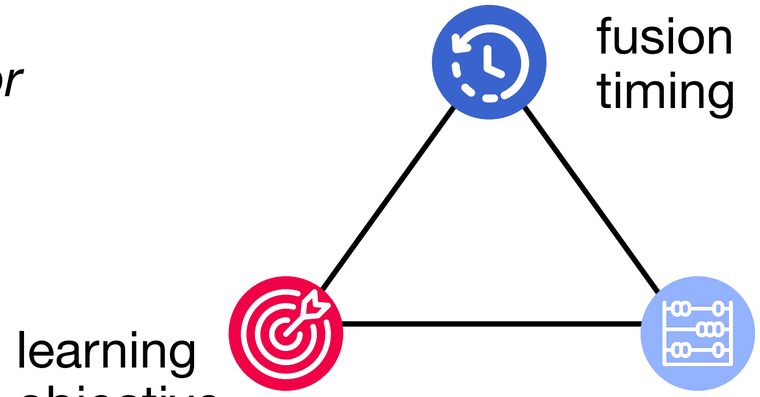
How to Build Multimodal Models: The Design Triangle

Multimodal model design

= choosing *when*, *how*, and *what for* modalities interact.

Each corner of the triangle constrains the others:

The choice of objective influences feasible fusion operations, and both determine viable training schemes.



early and late fusion training schemes

architectures, building blocks, etc.

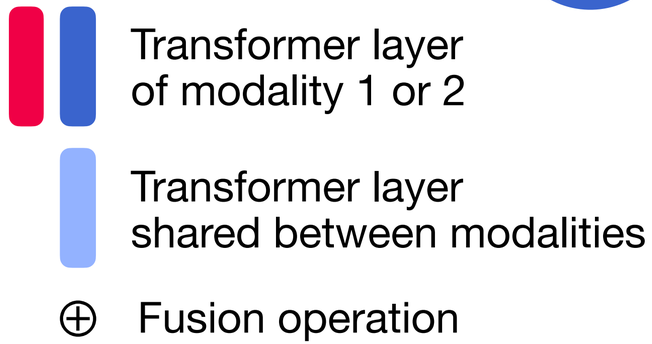
contrastive, masking, autoregressive and *objectives emerge from cross-modal structure*

Early and Late Fusion Schemes

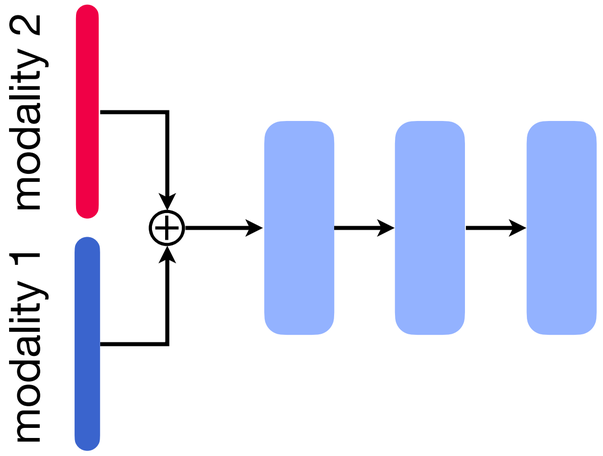
Early: Single model processes all modalities together from the start.

Late: Separate models for each modality, combined at output or loss.

Middle: At intermediate feature layers (via fusion or adaptors).

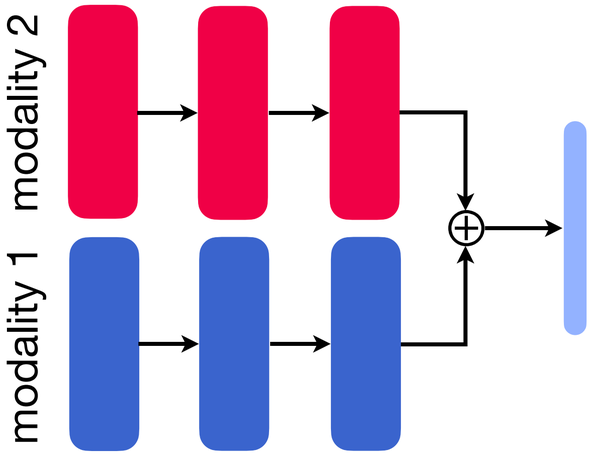


Early Fusion



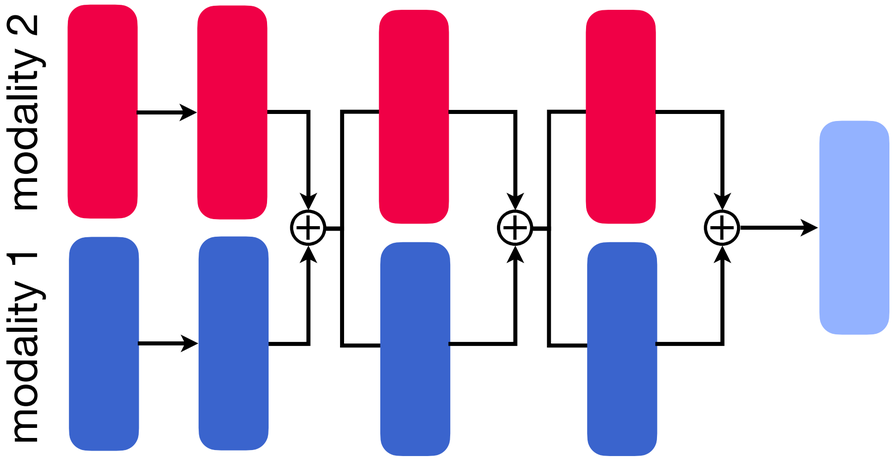
e.g., ViLT, etc.

Late Fusion



e.g., CLIP, etc.

Middle Fusion

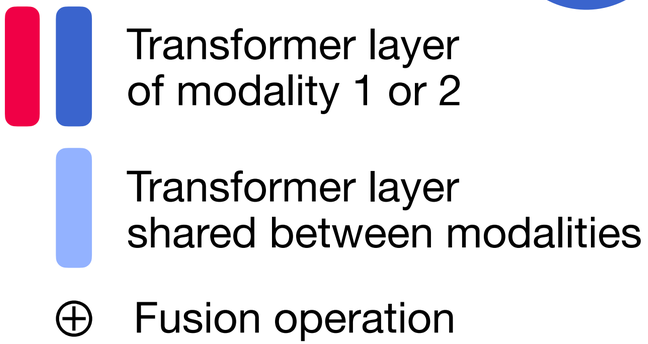


e.g., Flamingo, LLaVA, etc.

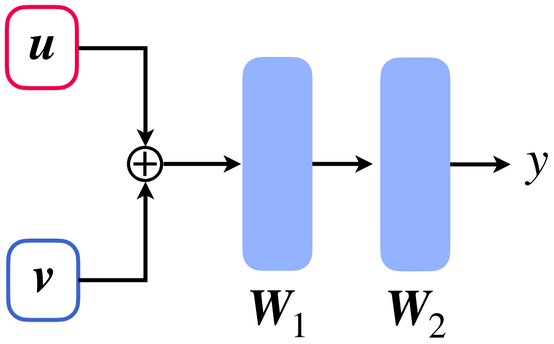
Early and Late Fusion Schemes

Every multimodal model makes a fundamental choice about when modalities interact.

Let's formalize this with a binary classifier taking inputs $\boldsymbol{u}$ and $\boldsymbol{v}$:



Early Fusion



Modalities mix at the input level

$$y = \sigma\left(\boldsymbol{W}_2\sigma\left(\boldsymbol{W}_1[\boldsymbol{u};\boldsymbol{v}] + \boldsymbol{b}_1\right) + \boldsymbol{b}_2\right)$$

where $[\boldsymbol{u};\boldsymbol{v}]$ denotes the selected fusion operator.

Early and Late Fusion Schemes

Every multimodal model makes a fundamental choice about when modalities interact.

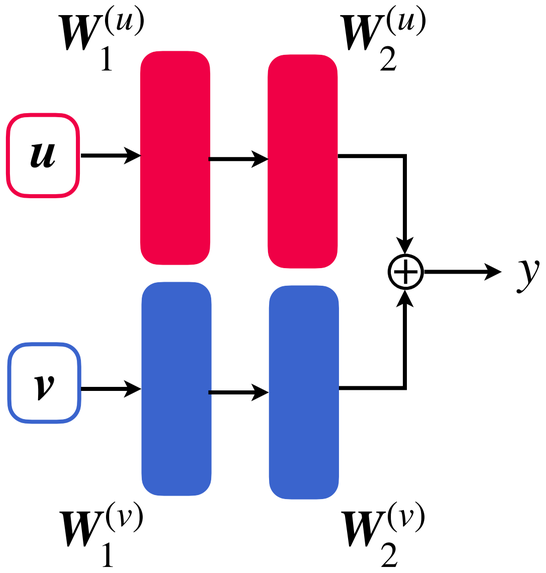
Let's formalize this with a binary classifier taking inputs $\boldsymbol{u}$ and $\boldsymbol{v}$:

Transformer layer of modality 1 or 2

Transformer layer shared between modalities

$\oplus$ Fusion operation

Late Fusion



Combine final decisions:

$$y = \frac{1}{2}\left(\sigma\left(\boldsymbol{W}_2^{(u)}\sigma\left(\boldsymbol{W}_1^{(u)}\boldsymbol{u} + \boldsymbol{b}_1^{(u)}\right) + \boldsymbol{b}_2^{(u)}\right)\right.$$
$$\left. + \sigma\left(\boldsymbol{W}_2^{(v)}\sigma\left(\boldsymbol{W}_1^{(v)}\boldsymbol{v} + \boldsymbol{b}_1^{(v)}\right) + \boldsymbol{b}_2^{(v)}\right)\right)$$

Early and Late Fusion Schemes

Every multimodal model makes a fundamental choice about when modalities interact.

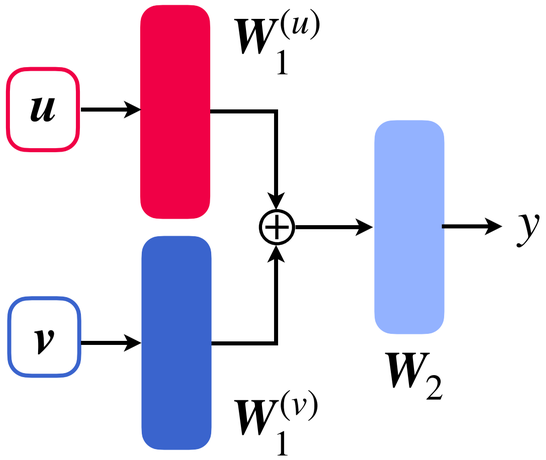
Let's formalize this with a binary classifier taking inputs $\boldsymbol{u}$ and $\boldsymbol{v}$:

Transformer layer of modality 1 or 2

Transformer layer shared between modalities

$\oplus$ Fusion operation

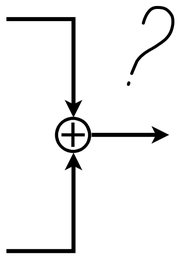
Middle Fusion



Process separately then combine features:

$$y = \sigma\left(\boldsymbol{W}_2\left[\sigma\left(\boldsymbol{W}_1^{(u)}\boldsymbol{u} + \boldsymbol{b}_1^{(u)}\right);\ \sigma\left(\boldsymbol{W}_1^{(v)}\boldsymbol{v} + \boldsymbol{b}_1^{(v)}\right)\right] + \boldsymbol{b}_2\right)$$

Building Blocks: **Fusion Operations**



Attention-based

$$\begin{aligned}
\text{Weighted:} \quad & \boldsymbol{z} = \alpha \boldsymbol{W}_u \boldsymbol{u} + \beta \boldsymbol{V}_v \boldsymbol{v} \\
\text{where:} \quad & [\alpha, \beta] = \text{softmax}([\boldsymbol{u}^\top \boldsymbol{w}_\alpha, \boldsymbol{v}^\top \boldsymbol{w}_\beta]) \\
\text{Modulation:} \quad & \boldsymbol{z} = [\alpha \boldsymbol{u}, (1-\alpha)\boldsymbol{v}]
\end{aligned}$$

Linear

$$\begin{aligned}
\text{Concatenation:} \quad & \boldsymbol{z} = [\boldsymbol{u}; \boldsymbol{v}] \in \mathbb{R}^{d_u + d_v} \\
\text{Summation:} \quad & \boldsymbol{z} = \boldsymbol{W}_u \boldsymbol{u} + \boldsymbol{V}_v \boldsymbol{v} \\
\text{Maximum:} \quad & \boldsymbol{z} = \max(\boldsymbol{W}_u \boldsymbol{u}, \boldsymbol{V}_v \boldsymbol{v})
\end{aligned}$$

Multiplicative

$$\begin{aligned}
\text{Element-wise:} \quad & \boldsymbol{z} = \boldsymbol{W}_u \boldsymbol{u} \odot \boldsymbol{V}_v \boldsymbol{v} \\
\text{Gating:} \quad & \boldsymbol{z} = \sigma(\boldsymbol{W}_u \boldsymbol{u}) \odot \boldsymbol{V}_v \boldsymbol{v}
\end{aligned}$$

Bilinear

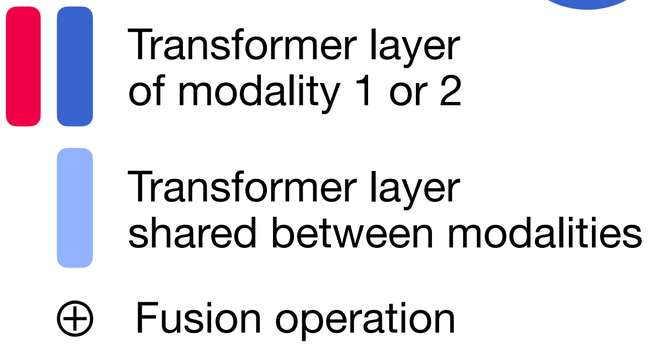
$$\begin{aligned}
\text{Full Bilinear:} \quad & z = \boldsymbol{u}^\top \boldsymbol{W} \boldsymbol{v} \in \mathbb{R} \\
\text{Bilinear Gated:} \quad & \boldsymbol{z} = (\boldsymbol{u}^\top \boldsymbol{W} \boldsymbol{v}) \sigma(\boldsymbol{v})
\end{aligned}$$

Early and Late Fusion Schemes

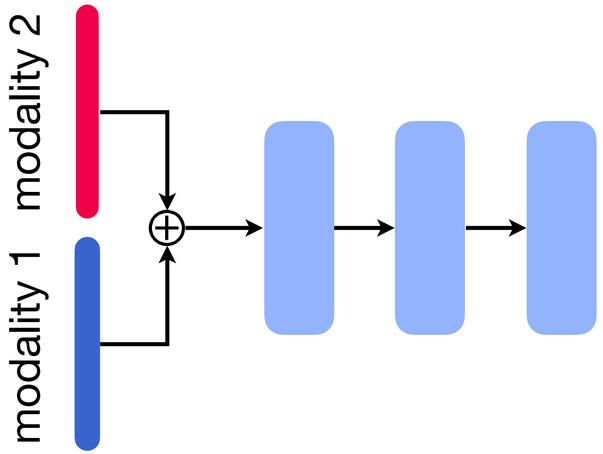
Early: Single model processes all modalities together from the start.

Late: Separate models for each modality, combined at output or loss.

Middle: At intermediate feature layers (via fusion or adaptors).

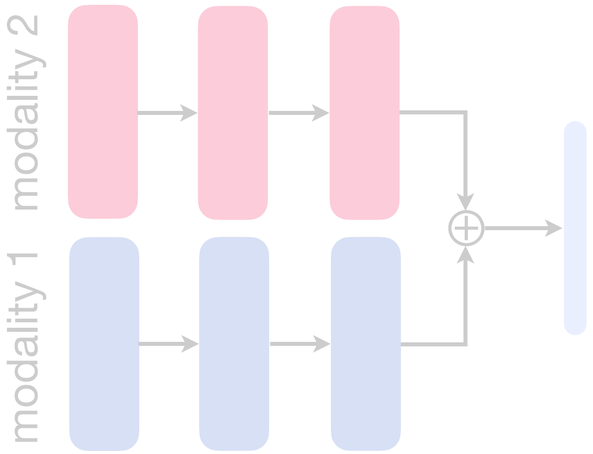


Early Fusion



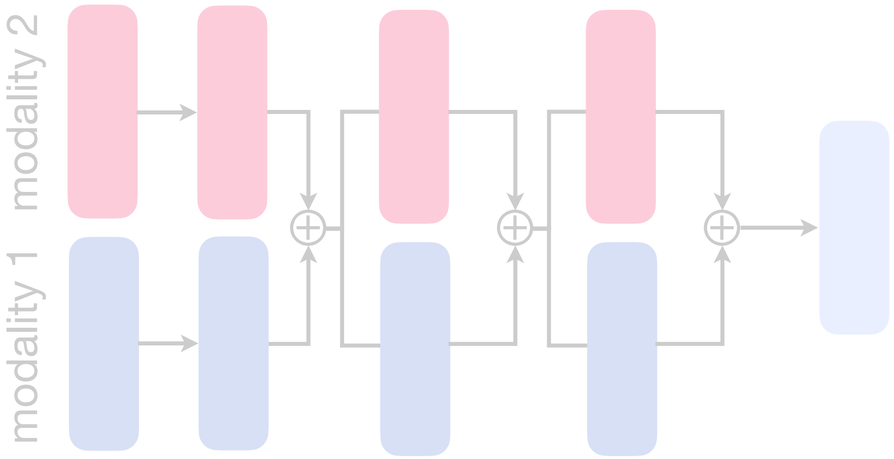
e.g., ViLT, etc.

Late Fusion



e.g., CLIP, etc.

Middle Fusion



e.g., Flamingo, LLaVA, etc.

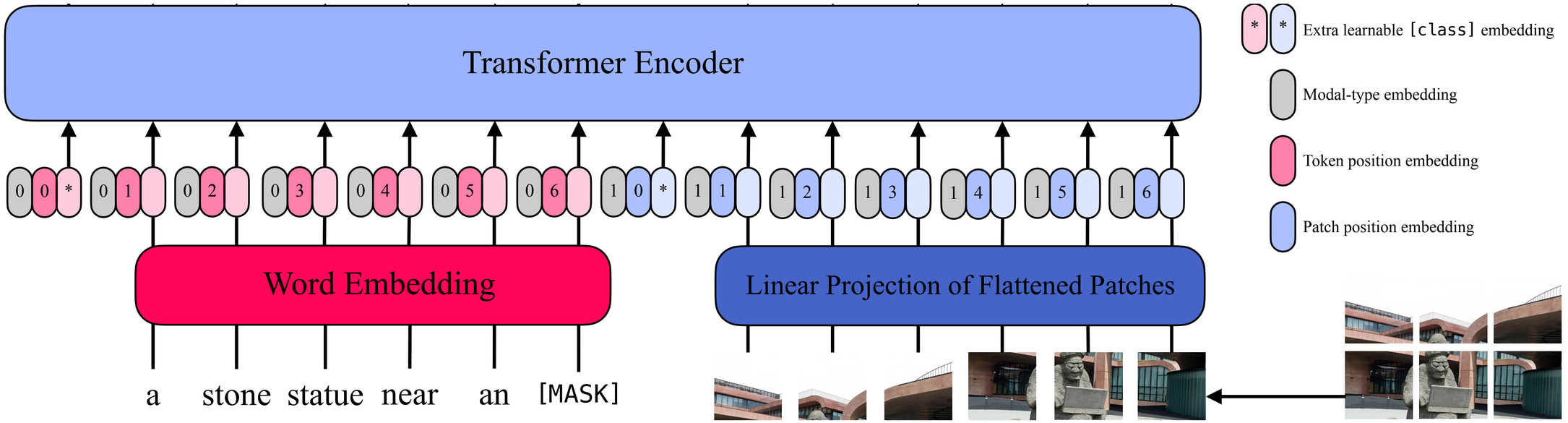
Examples of Multimodal Models: ViLT (Kim et al., 2021)

Example of **Early Fusion** Model

Vision and Language Transformer (ViLT)

Key Idea:

Single-stream early fusion: tokenize text and image patches, add modality and position embeddings, then process the interleaved tokens with one transformer encoder.



Examples of Multimodal Models: ViLT (Kim et al., 2021)

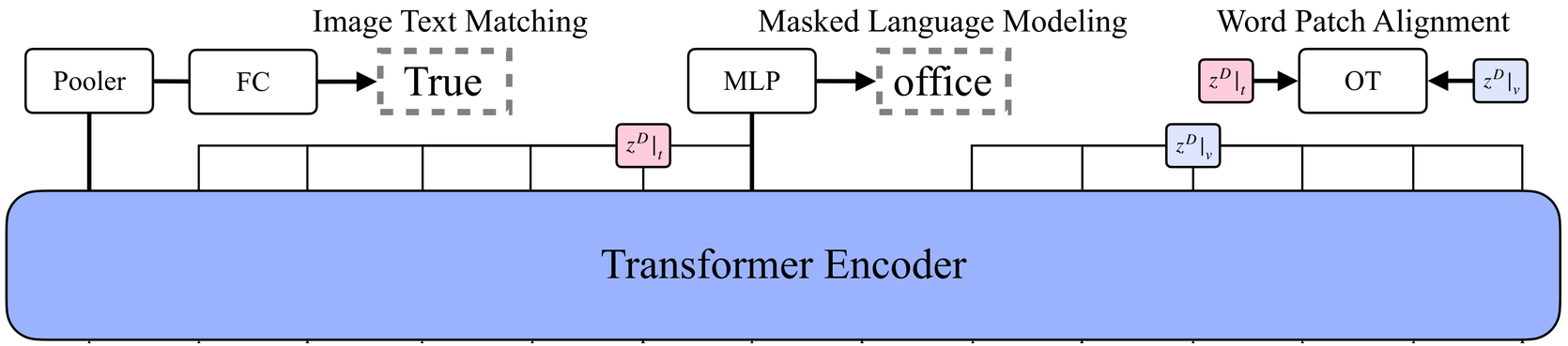
Example of **Early Fusion** Model

Vision and Language Transformer (ViLT)

Self-supervised or weakly supervised signals that emerge from the multimodal pairing itself!

ViLT Losses

- Image-Text Matching:
  Binary classification on [CLS] to decide if the image-sentence pair is **matched or mismatched**.
- Masked Language Modeling:
  Mask some text tokens and predict them conditioned on the image and remaining text.
- **Word-Patch Alignment via Optimal Transport:**
  Encourage **fine-grained correspondences** between word embeddings and image patch embeddings by minimizing an **OT distance** between the two token sets.



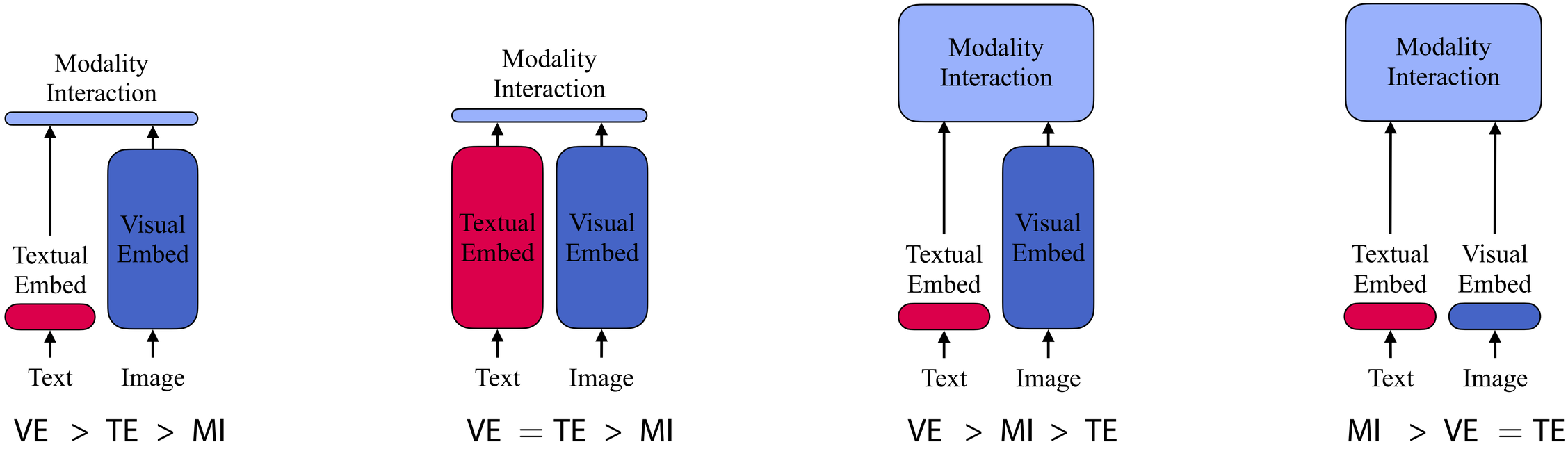
Examples of Multimodal Models: ViLT (Kim et al., 2021)

Example of **Early Fusion** Model

Vision and Language Transformer (ViLT)

VE: Vision Encoder
TE: Text Encoder
MI: Modality Interaction

Taxonomy and **design choices** of vision-language models:



e.g., CLIP
as example of **late fusion**

e.g., ViLT
as example of **early fusion**

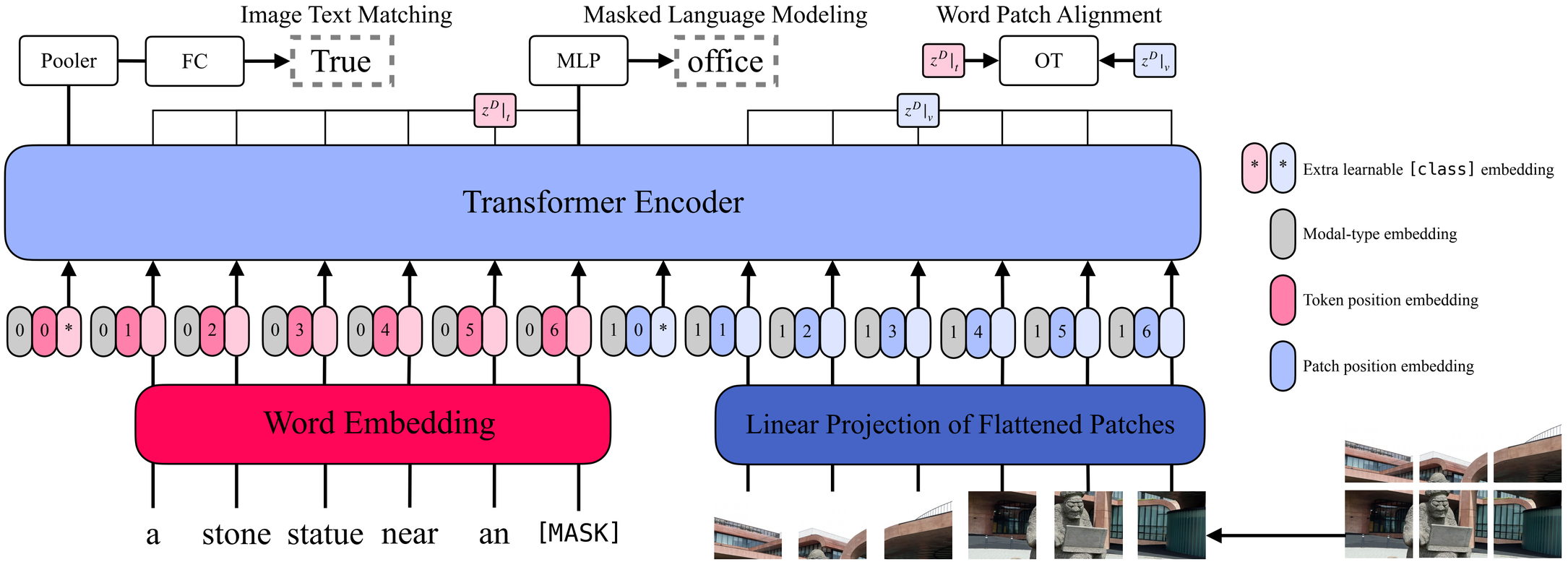
Examples of Multimodal Models: ViLT (Kim et al., 2021)

Example of **Early Fusion** Model

Vision and Language Transformer (ViLT)

Summary

ViLT shows early fusion: text tokens and image patches enter one shared transformer, which learns cross-modal interactions via self-attention and multimodal self-supervised losses.

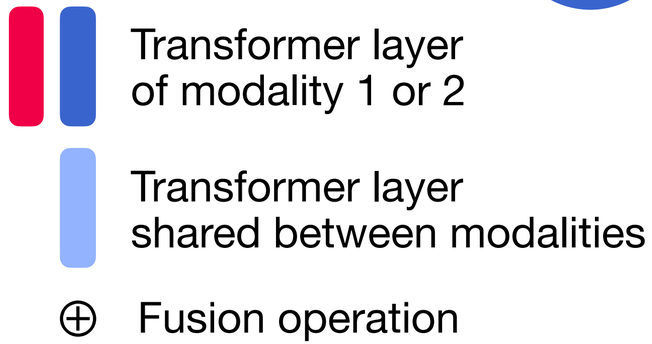


Early and Late Fusion Schemes

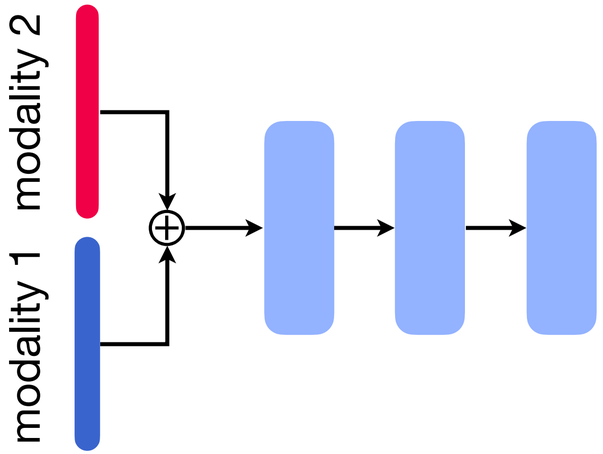
Early: Single model processes all modalities together from the start.

Late: Separate models for each modality, combined at output or loss.

Middle: At intermediate feature layers (via fusion or adaptors).

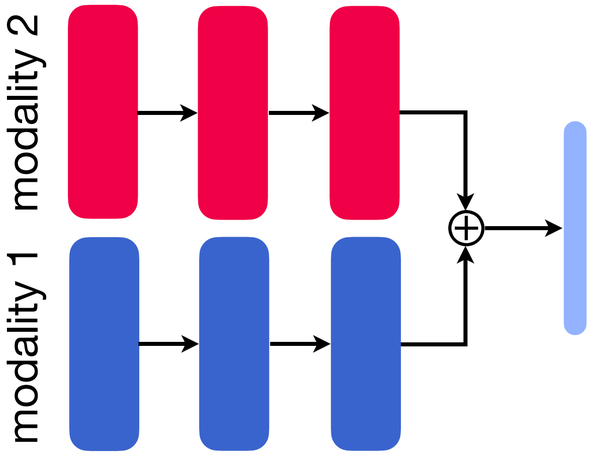


Early Fusion



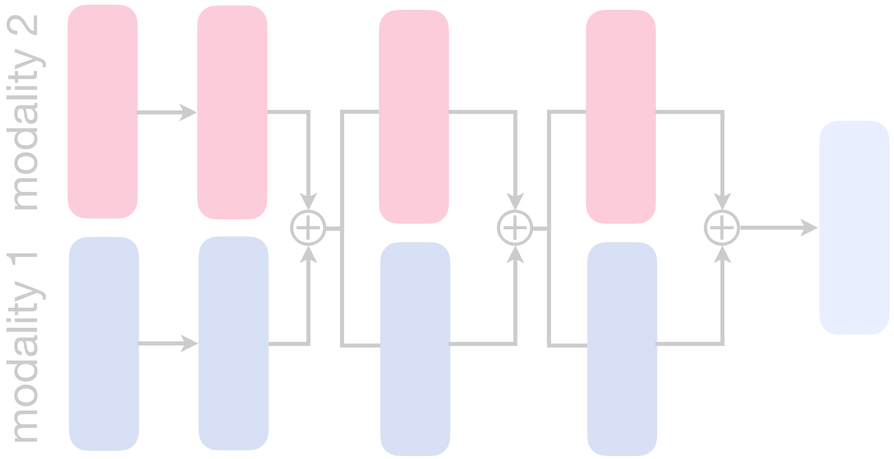
e.g., ViLT, etc.

Late Fusion



e.g., CLIP, etc.

Middle Fusion



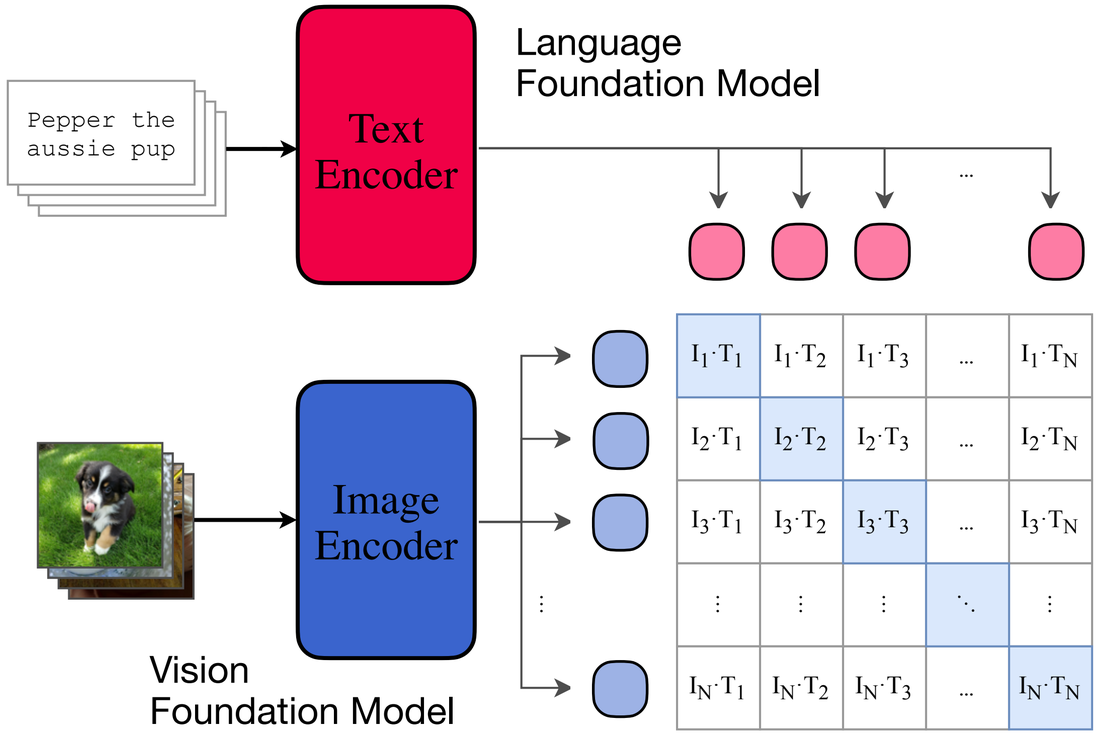
e.g., Flamingo, LLaVA, etc.

Examples of Multimodal Models: CLIP

(Radford et al., 2021)

Example of **Late Fusion** Model

Contrastive Language–Image Pretraining (CLIP) aligns image and text embeddings by training two separate encoders, a vision and a language model, on large-scale image–caption pairs.



… trained via **contrastive objective**:

model learns by maximizing the similarity of matching pairs and minimizing it for non-matching pairs.

across all pairs in a batch:

InfoNCE Loss

$$\mathcal{L}_{\text{InfoNCE}} = -\log \frac{\exp\left(\text{sim}\left(I_i, T_i\right)/\tau\right)}{\sum_j \exp\left(\text{sim}\left(I_i, T_j\right)/\tau\right)}$$

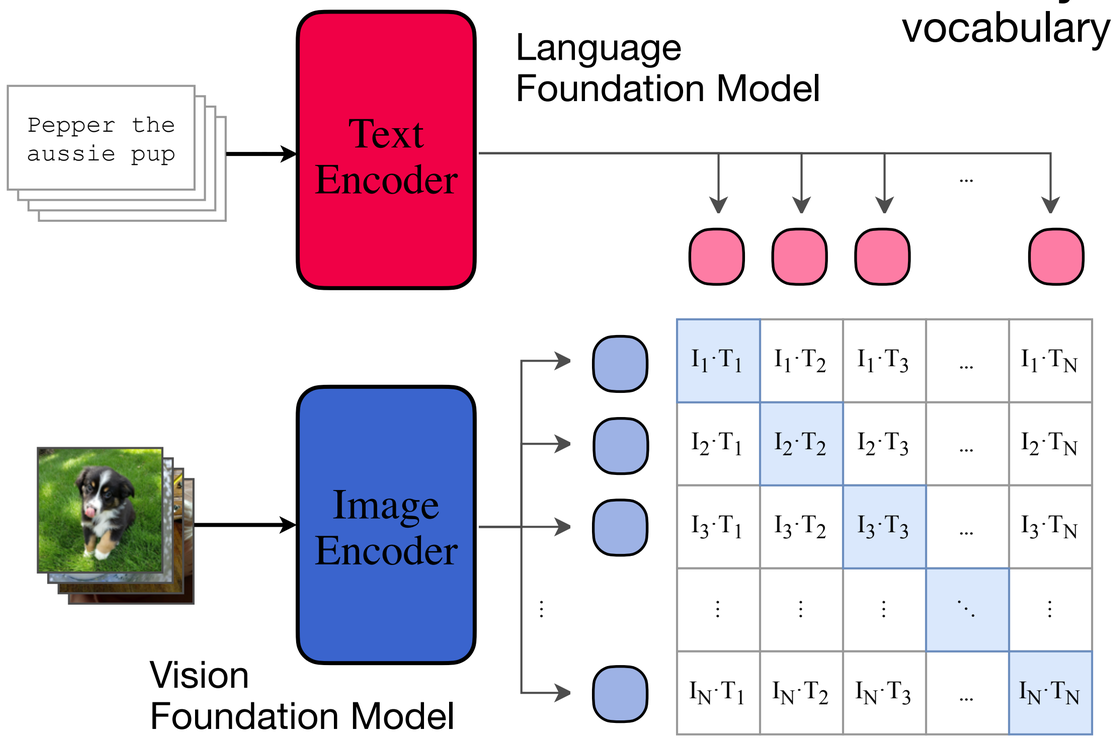
Examples of Multimodal Models: CLIP (Radford et al., 2021)

Example of **Late Fusion** Model

Why does this work?

Massive and weakly supervised: CLIP was trained on web-scale image-text pairs, capturing natural language descriptions rather than fixed labels.

Diversity is the supervision: the variety of captions provides open-vocabulary semantic grounding that fixed category datasets cannot offer.



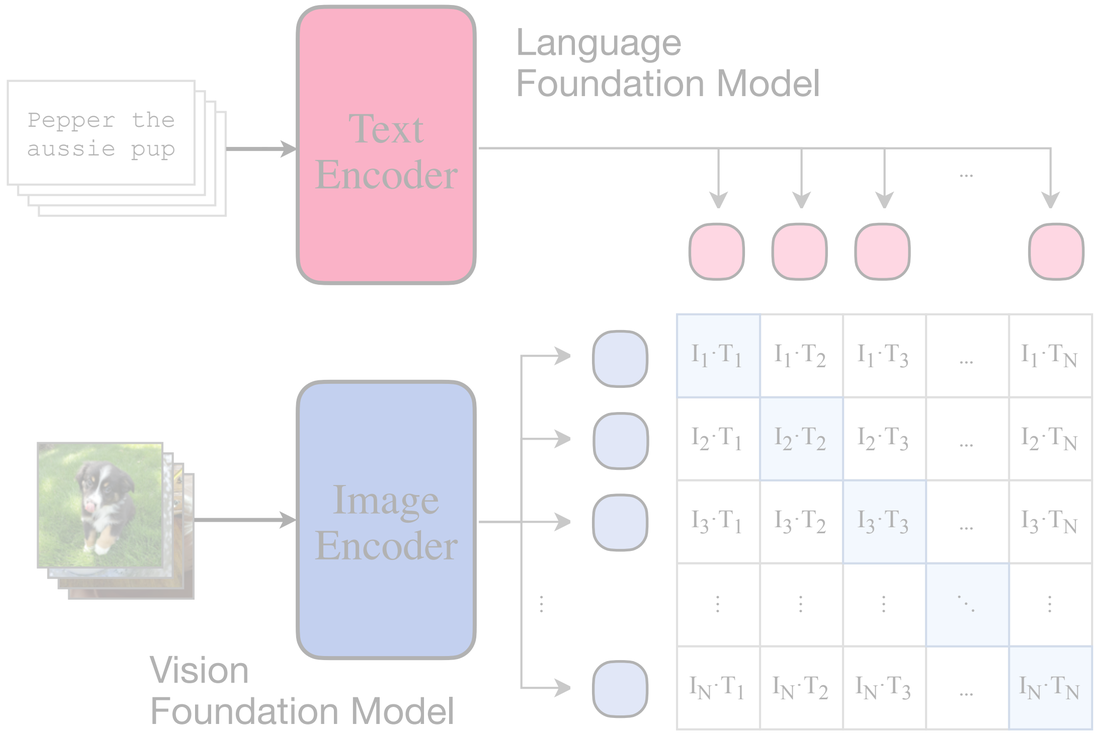
Dataset

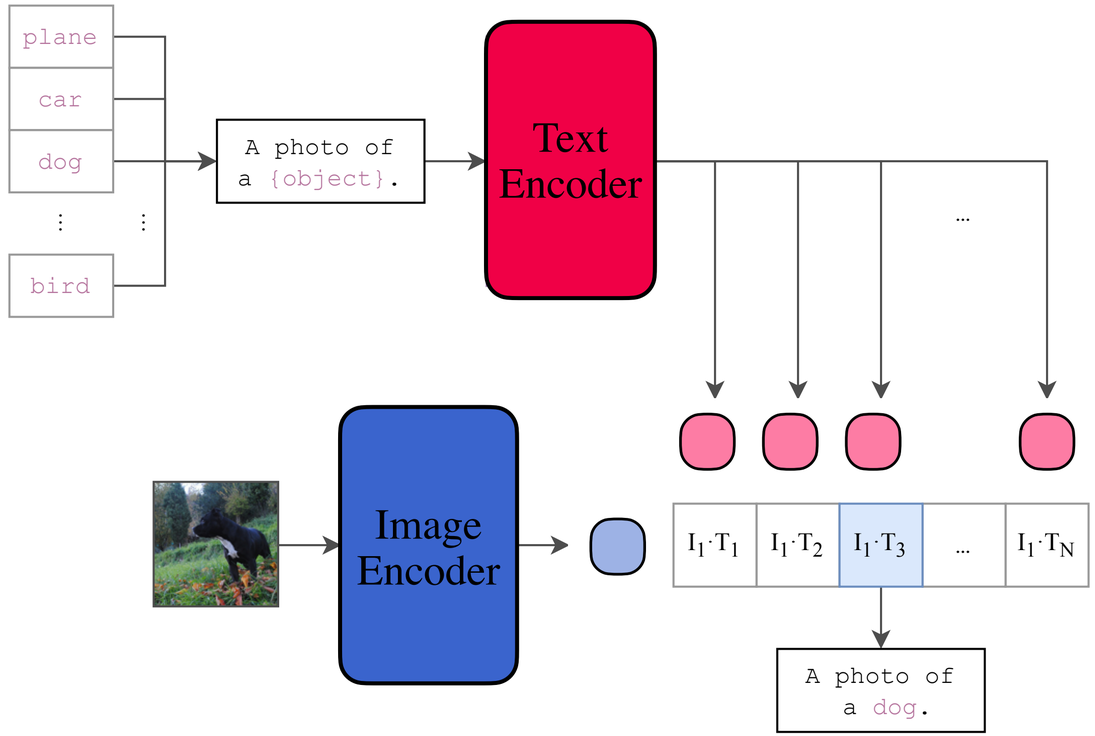
- 400M (image, text) pairs from the Internet
- 500k queries, i.e., words occuring at least 100 times in English Wikipedia
- ~20k (image, text) pairs per query

Examples of Multimodal Models: CLIP (Radford et al., 2021)

Example of **Late Fusion** Model

Zero-shot inference:
CLIP matches image embeddings to text embeddings of arbitrary class descriptions, selecting the highest similarity score to classify images into categories never seen during training.



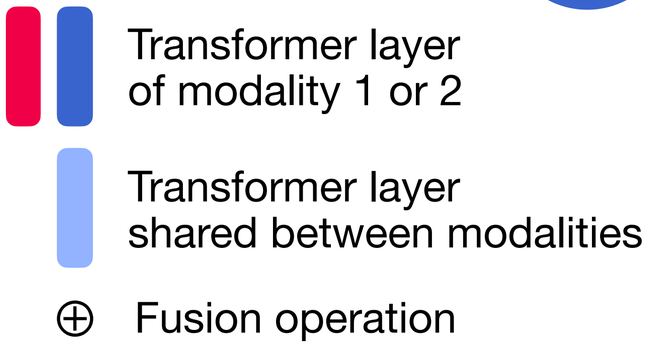


Early and Late Fusion Schemes

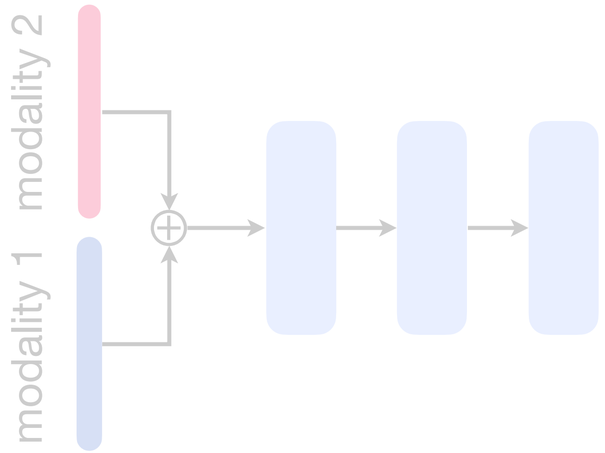
Early: Single model processes all modalities together from the start.

Late: Separate models for each modality, combined at output or loss.

Middle: At intermediate feature layers (via fusion or adaptors).

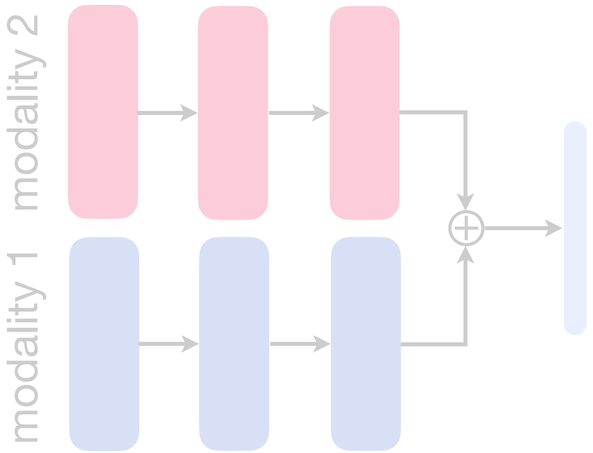


Early Fusion



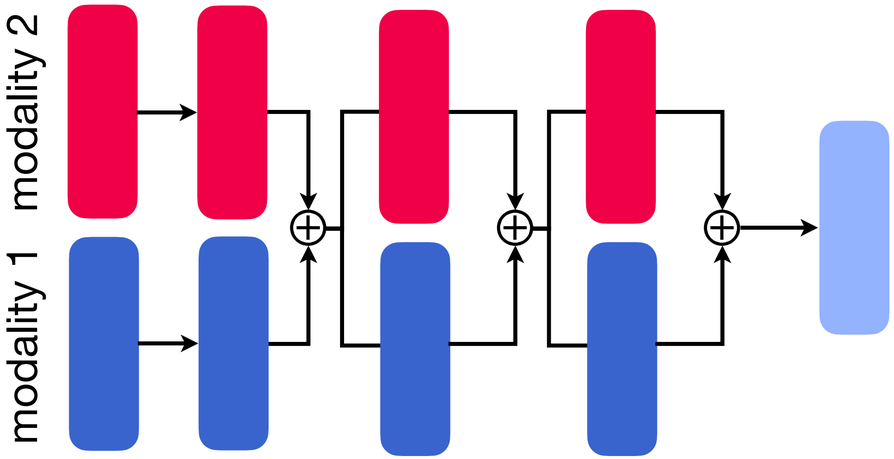
e.g., ViLT, etc.

Late Fusion



e.g., CLIP, etc.

Middle Fusion

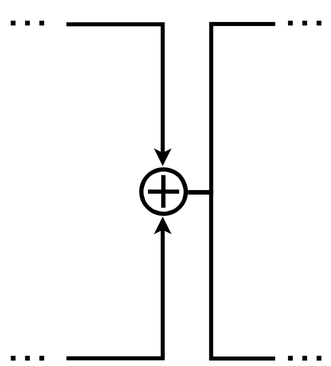


e.g., Flamingo, LLaVA, etc.

From Early and Late to Middle Fusion: **Learning Cross-Modal Interactions**

Middle fusion focuses on learning intermediate connections between modalities rather than merging only at input or output.

- Requires special building blocks to enable dynamic alignment.
- These mechanisms make it possible for one modality to contextualize or modulate another in a more flexible way than early or late fusion.



e.g. co- or cross-attention,
gating mechanisms,
or projection layers

Building Blocks: **From Attention to Cross-Attention**

Attention *Reminder!*

The general attention operation is defined as:

Attention

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

Self-Attention

Within-modality relationships where $Q = K = V$ are from the same modality.

Self-Attention

$$\text{SelfAttn}(X) = \text{softmax}\left(\frac{(XW_Q)\,(XW_K)^\top}{\sqrt{d_k}}\right)(XW_V)$$

- X : input embeddings from the same modality
- W_Q, W_K, W_V : learned projection matrices
- Each token attends to all other tokens within the same modality.

Building Blocks: **From Attention to Cross-Attention**

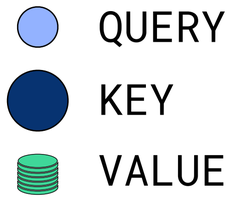
Cross-Attention

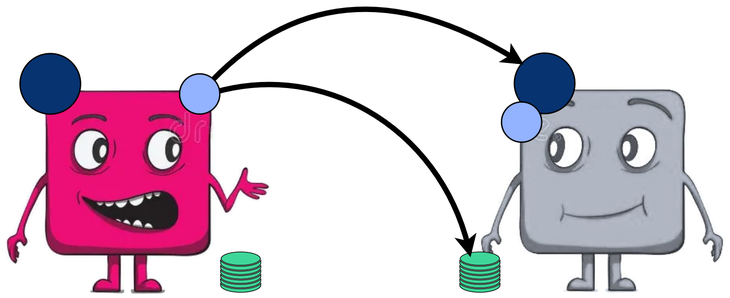
Queries come from one modality (e.g., text), while Keys and Values come from another (e.g., image features).

- X_Q : query modality (e.g., text embeddings)
- X_{KV} : key/value modality (e.g., image embeddings)
- This allows one modality to attend to and integrate information from another.

Cross-Attention

$$\text{CrossAttn}\left(X_Q, X_{KV}\right) = \text{softmax}\left(\frac{\left(X_Q W_Q\right)\left(X_{KV} W_K\right)^\top}{\sqrt{d_k}}\right)\left(X_{KV} W_V\right)$$





Building Blocks: **From Attention to Cross-Attention**

Cross-Attention

Queries come from one modality (e.g., text), while keys and values come from another (e.g., image features)

- X_Q : query modality (e.g., text embeddings)
- X_{KV} : key/value modality (e.g., image embeddings)
- This allows one modality to attend to and integrate information from another.

Cross-Attention

$$\mathrm{CrossAttn}\left(X_Q, X_{KV}\right) = \mathrm{softmax}\left(\frac{\left(X_Q W_Q\right)\left(X_{KV} W_K\right)^\top}{\sqrt{d_k}}\right)\left(X_{KV} W_V\right)$$

Co-Attention

Bidirectional attention between modalities such that modalities can attend to each other simultaneously or iteratively.

Co-Attention

$$\mathrm{CoAttn}(X, Y) = \mathrm{CrossAttn}(X, Y) + \mathrm{CrossAttn}(Y, X)$$

Building Blocks: **Gated Cross-Attention**

Key Idea: Extends cross-attention by adding a learnable gate that controls how much one modality influences the other.

The gate dynamically adjusts attention flow allowing selective fusion instead of always blending modalities equally.

- Provides **stability** when coupling large encoders, avoids overwhelming one modality with another.
- **Adaptive conditioning**: the model learns *when* another modality matters and *when* to ignore it.

Gated Cross-Attention

$$\text{GatedCrossAttn}\left(X_Q, X_{KV}\right) = g \cdot \text{CrossAttn}\left(X_Q, X_{KV}\right)$$

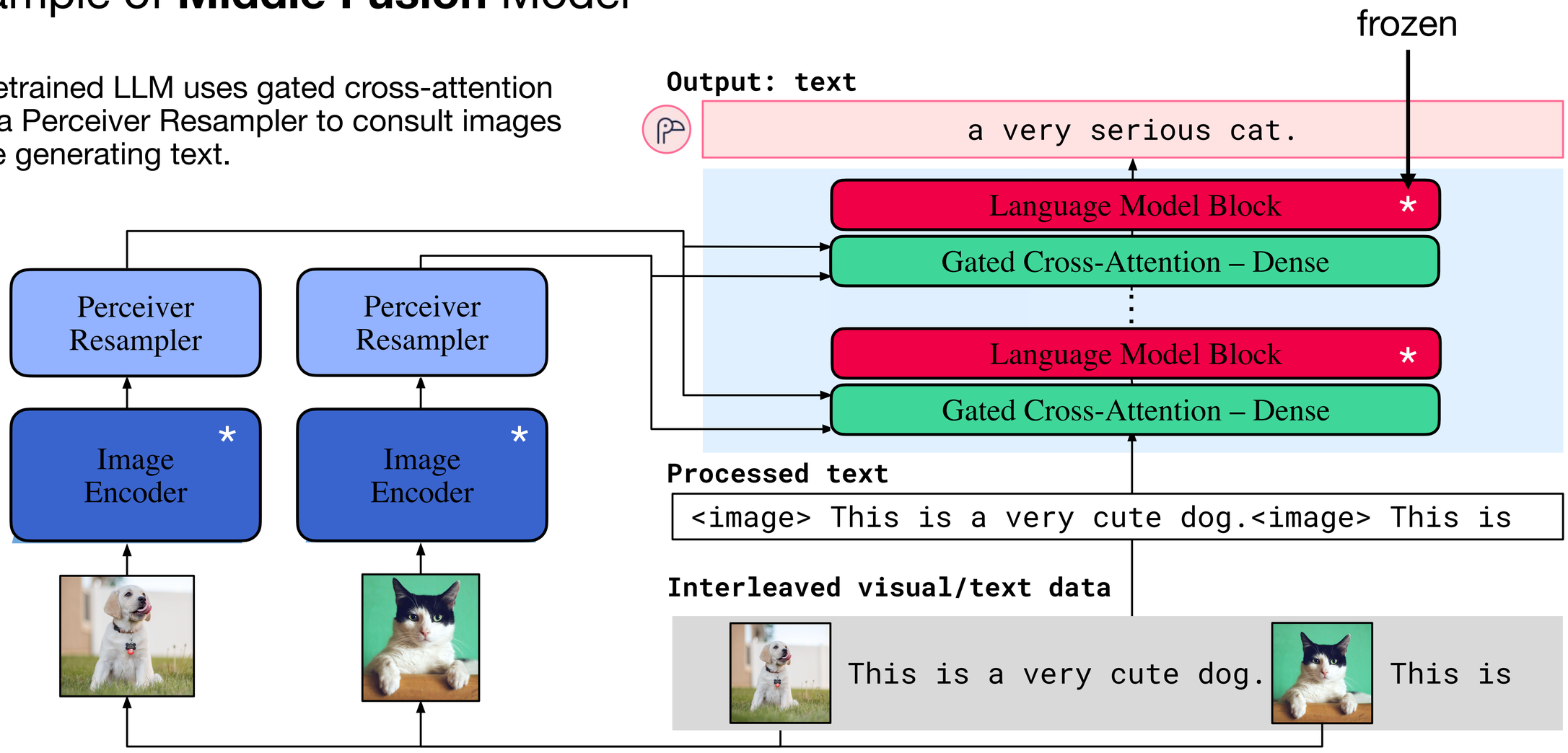
with $g \in [0,1]$: scalar or vector gate (often learned per layer or head).

Examples of Multimodal Models: Flamingo

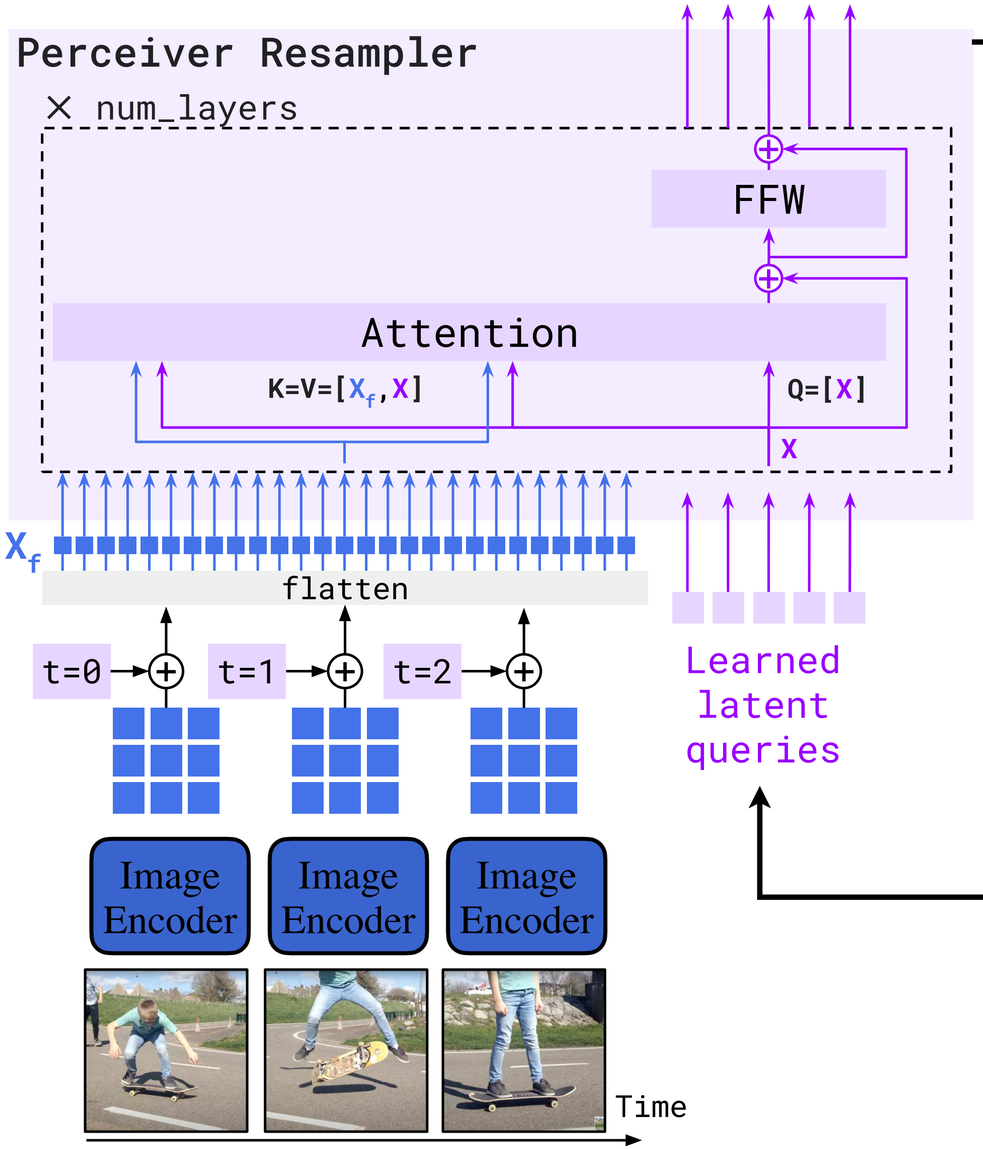
Example of **Middle Fusion** Model

Alayrac et al., (2022)

A pretrained LLM uses gated cross-attention and a Perceiver Resampler to consult images while generating text.



Building Blocks: **Perceiver Resampler**



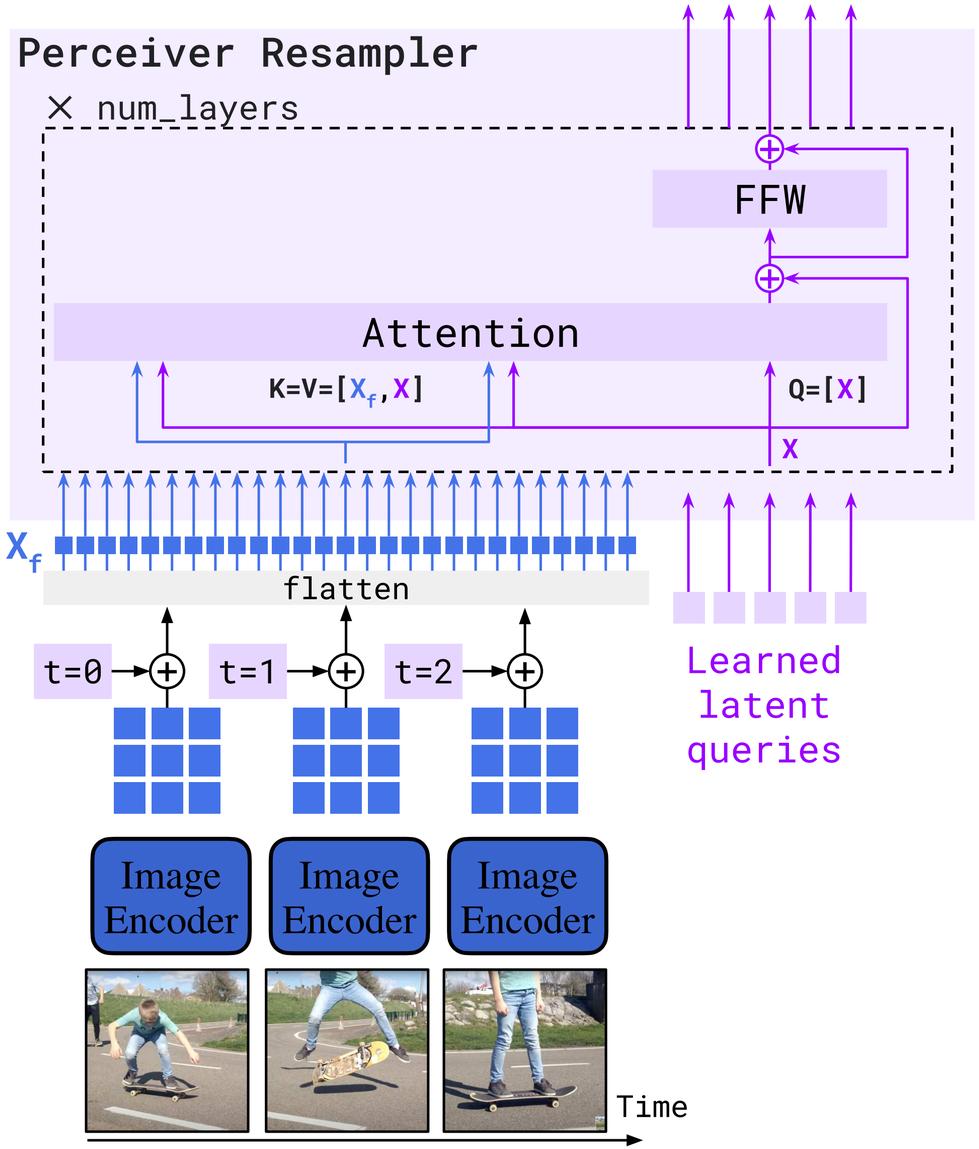
cross-attention

Purpose:
The Perceiver Resampler turns arbitrarily many visual tokens into a small, fixed set of latent tokens by letting learned queries attend to the concatenated feature grid and latents, making the interface to the LLM constant size.

Mechanism:
Uses **cross-attention** from learnable queries (latents) to all image features.

Outcome:
Produces a compact summary that preserves global context while cutting compute cost, ideal for feeding into large language models.

Building Blocks: **Perceiver Resampler**



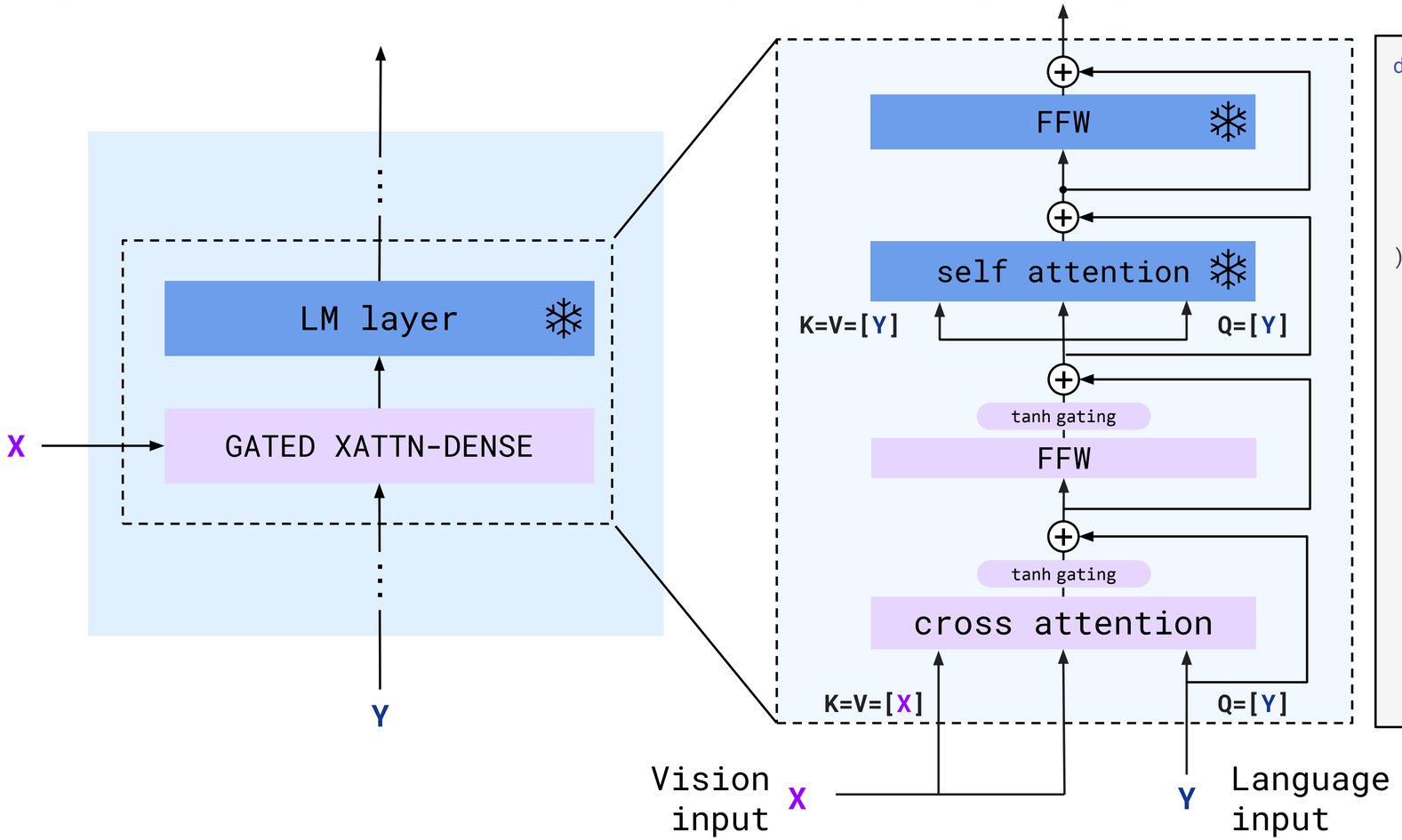

```python
def perceiver_resampler(
    x_f,  # The [T, S, d] visual features (T=time, S=space)
    time_embeddings,  # The [T, 1, d] time pos embeddings.
    x,  # R learned latents of shape [R, d]
    num_layers,  # Number of layers
):
  """The Perceiver Resampler model."""

  # Add the time position embeddings and flatten.
  x_f = x_f + time_embeddings
  x_f = flatten(x_f)  # [T, S, d] -> [T * S, d]
  # Apply the Perceiver Resampler layers.
  for i in range(num_layers):
    # Attention.
    x = x + attention_i(q=x, kv=concat([x_f, x]))
    # Feed forward.
    x = x + ffw_i(x)
  return x
```

Jaeger et al., (2021)

Building Blocks: **Gated Cross-Attention**

Gated Cross-Attention controls how much visual information flows into the language model.

Visual features x are attended to by the language stream y through a cross-attention layer, scaled by a learnable gate α that starts near zero and opens gradually during training.




```python
def gated_xattn_dense(
    y,  # input language features
    x,  # input visual features
    alpha_xattn, # xattn gating parameter – init at 0.
    alpha_dense, # ffw gating parameter – init at 0.
):
  """Applies a GATED XATTN-DENSE layer."""

  # 1. Gated Cross Attention
  y = y + tanh(alpha_xattn) * attention(q=y, kv=x)
  # 2. Gated Feed Forward (dense) Layer
  y = y + tanh(alpha_dense) * ffw(y)

  # Regular self-attention + FFW on language
  y = y + frozen_attention(q=y, kv=y)
  y = y + frozen_ffw(y)
  return y  # output visually informed language features
```

Examples of Multimodal Models: Flamingo

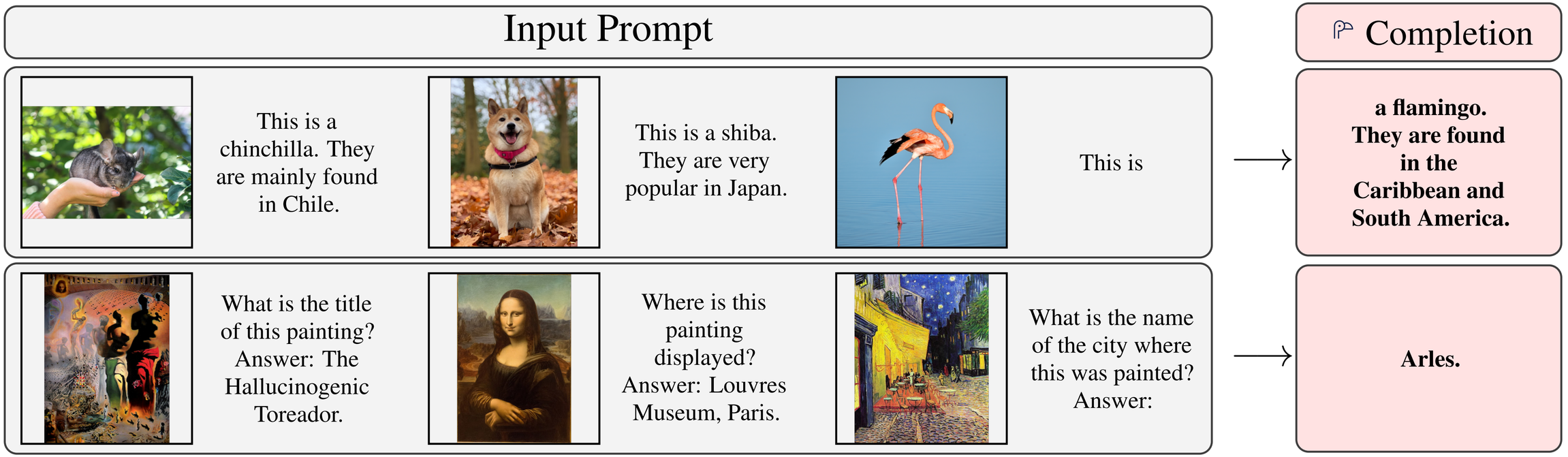
Example of **Middle Fusion** Model

Alayrac et al., (2022)

Flamingo can handle interleaved sequences of images and text as input, treating them as a single multimodal prompt.

The model can generate coherent, context-aware answers or descriptions conditioned on both visual and textual context.

Demonstrates capabilities across multimodal tasks such as captioning, visual question answering, etc,, all within one model.



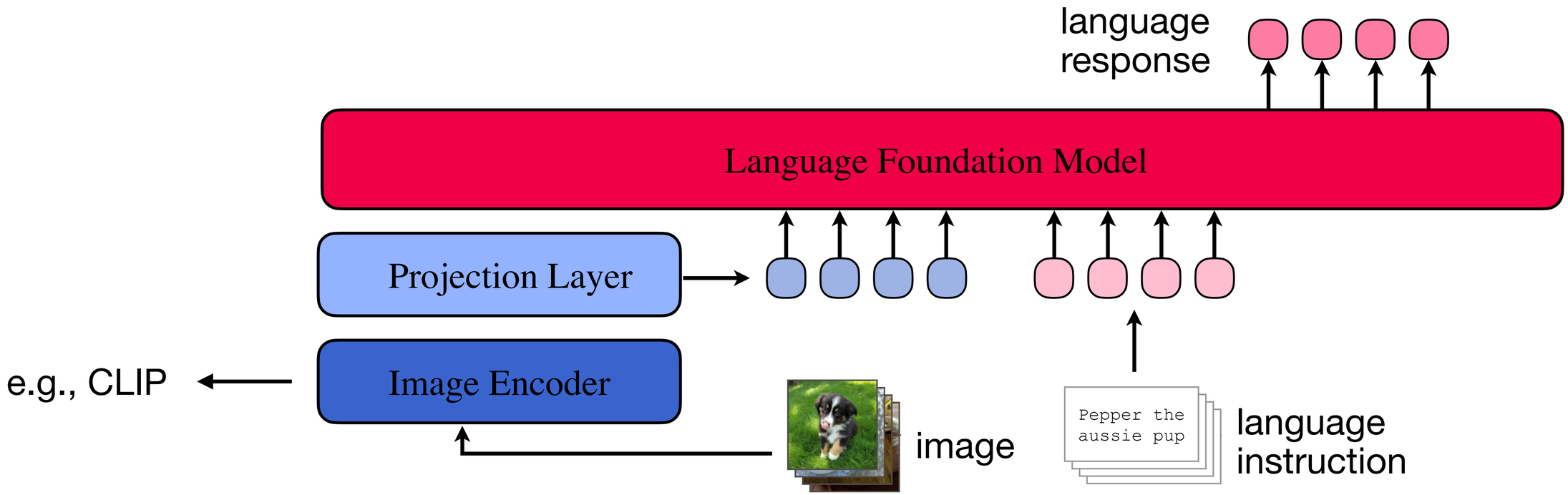
Examples of Multimodal Models: LLaVA

Liu et al., (2023)

Example of **Middle Fusion** Model

Large Language and Vision Assistant (LLaVA)

LLaVA connects a pretrained vision encoder to a language model through a **projection layer** that maps image features into the LLM's token embedding space, enabling **end-to-end multimodal instruction following and dialogue**.



Examples of Multimodal Models: LLaVA

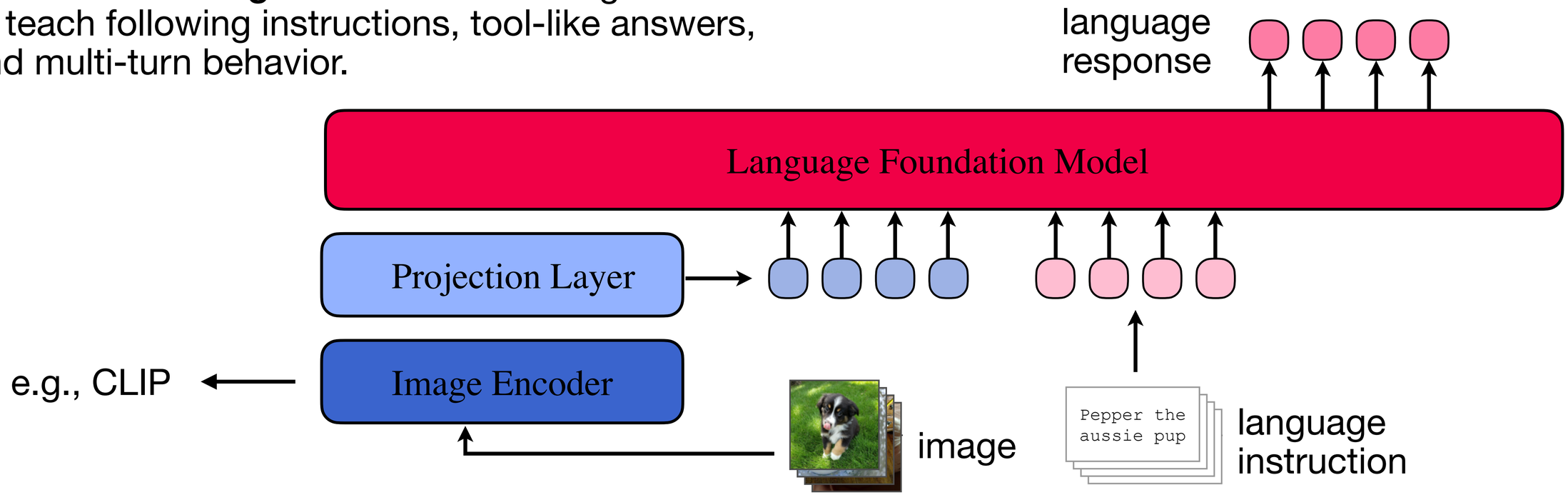
Liu et al., (2023)

Example of **Middle Fusion** Model

Large Language and Vision Assistant (LLaVA)

Training in Two Stages:

1. Feature-alignment pretraining on image-caption pairs so the projector learns the mapping.
2. **Instruction tuning** on multimodal dialogues to teach following instructions, tool-like answers, and multi-turn behavior.



Examples of Multimodal Models: LLaVA

Liu et al., (2023)

Example of **Middle Fusion** Model

Large Language and Vision Assistant (LLaVA)

Examples of capabilities:

s
What is unusual about this image?

The unusual aspect of this image is a man ironing clothes on the back of a minivan or van. This is not a typical place to perform this activity, as one would usually iron clothes in a more stationary and safe location, such as a home, using a regular ironing board. The scene depicted in the image is peculiar as it involves a makeshift ironing setup on a vehicle, which can be both unsafe and unconventional. Additionally, it is not clear how the man is able to maintain balance and stability while ironing clothes in such an unstable environment.

An open refrigerator filled with a variety of food items. In the left part of the compartment, towards the front, there is a plastic box of strawberries with a small bag of baby carrots on top. Towards the back, there is a stack of sauce containers. In the middle part of the compartment, towards the front, there is a green plastic box, and there is an unidentified plastic bag placed on it. Towards the back, there is a carton of milk. In the right part of the compartment, towards the front, there is a box of blueberries with three yogurts stacked on top. The large bottle of yogurt is Fage non-fat yogurt, and one of the smaller cups is Fage blueberry yogurt. The brand and flavor of the other smaller cup are unknown. Towards the back, there is a container with an unknown content.

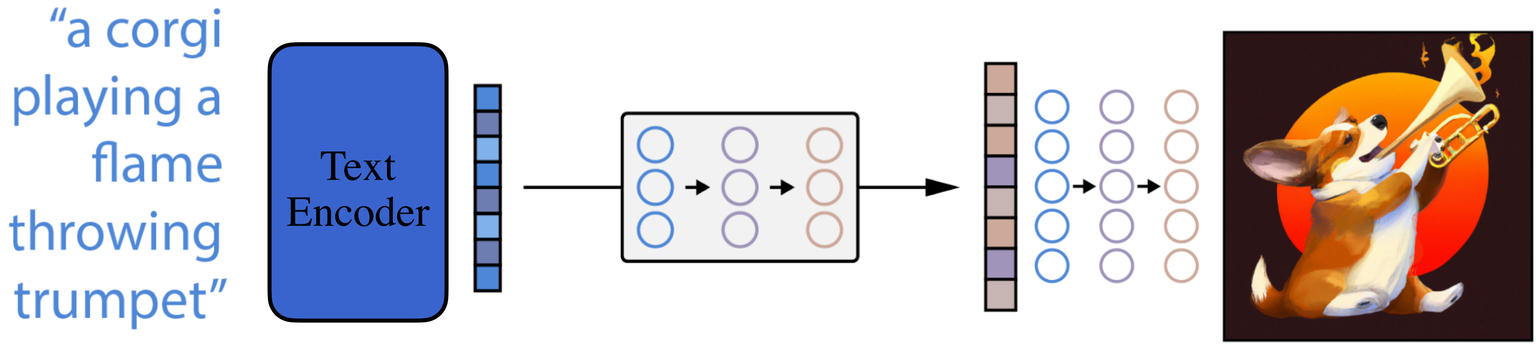
Multimodal Generative Models: Text-to-Image Generation

So far: Multimodality for image ↔ text **understanding**

What about? Multimodality for text → image **generation**

On DALL·E

> We describe a simple approach for this task based on a transformer that autoregressively models the text and image tokens as a single stream of data. With sufficient data and scale, our approach is competitive with previous domain-specific models.



DALL·E

Ramesh et al., (2021)

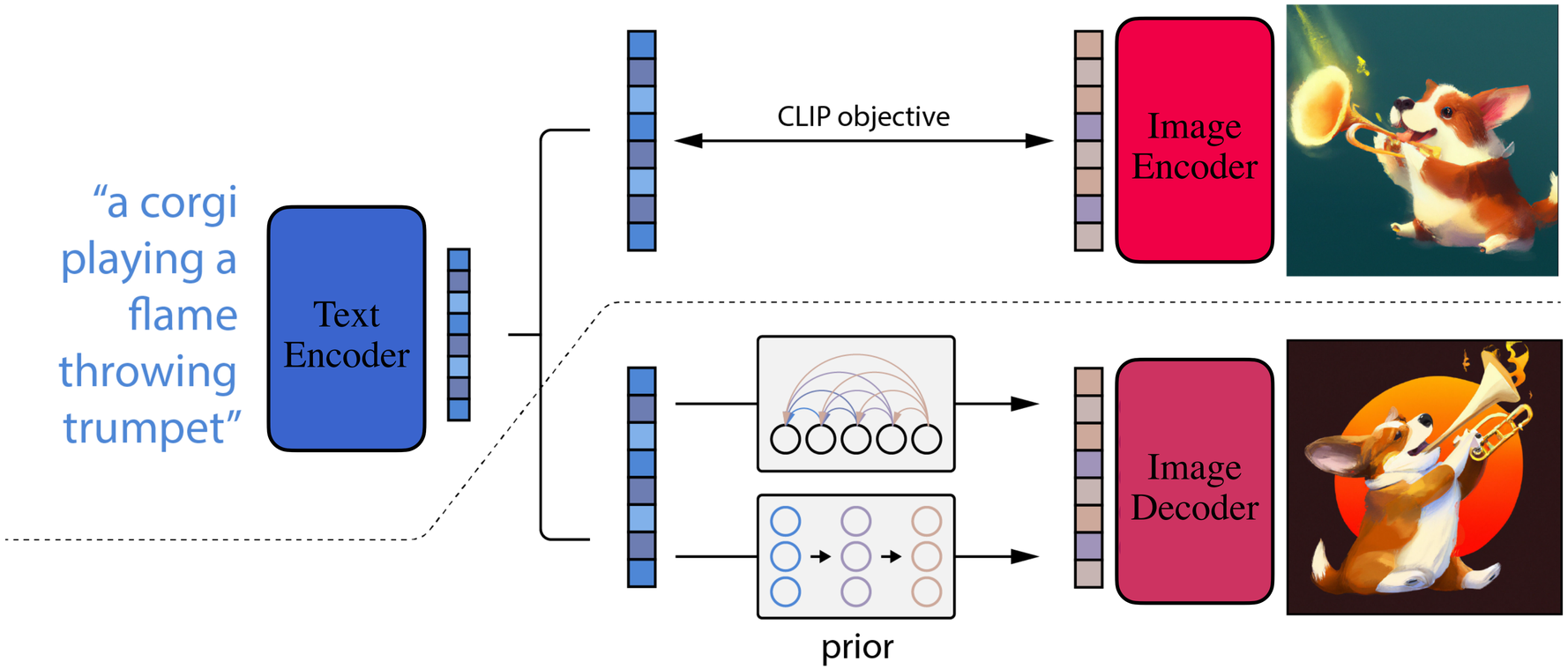
Multimodal Generative Models: Text-to-Image Generation

Two-Stage Training

1. **CLIP-like stage**: Jointly train text and image encoders with a contrastive CLIP objective to align text and image representations in a shared embedding space.
2. **Generative stage**: Train a prior and image decoder to map text embeddings to corresponding image representations and then decode them into pixels (using a VQ-VAE or *diffusion* decoder).

Intuition:

CLIP learns what should be in the image; the decoder learns how to draw it.



DALL·E

Ramesh et al., (2022)

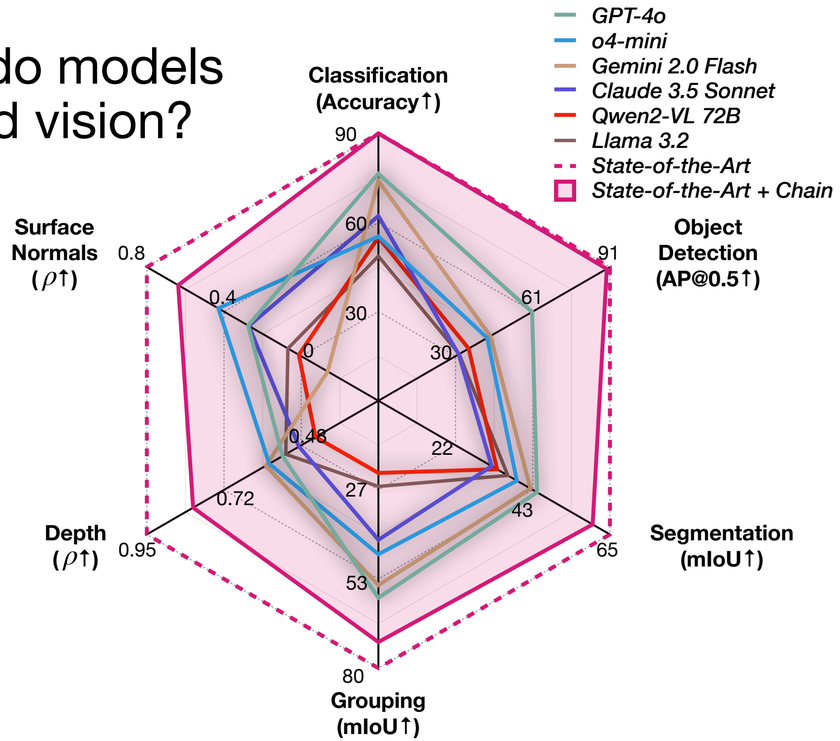
Truly *Native* Multimodal Models?

So far: Most models are ***bimodal***, not multimodal
and mostly capable to ***translate*** between two modalities.

→ e.g., CLIP (text ↔ image), LLaVA (image → text), DALL·E (text → image)

Strong language reasoning but no perception.
Architectures treat modalities very independently..

How well do models understand vision?



Ramachandran et al., (2025)

Shift in **GPT-4** *!* First model to showcase strong multimodal general AI capabilities and impressive logical reasoning capability.

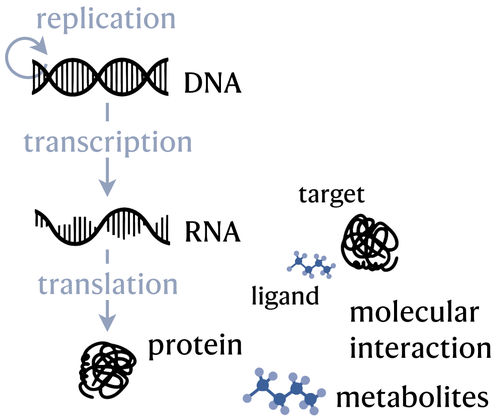
What changed? Native multimodal integration in GPT-4o is achieved through a **unified, end-to-end transformer architecture** that **processes all modalities within a single model.**

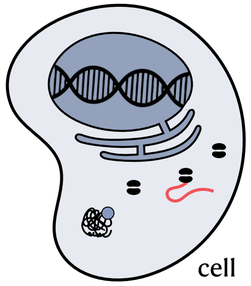
How do we solve the level of multimodality in biology?

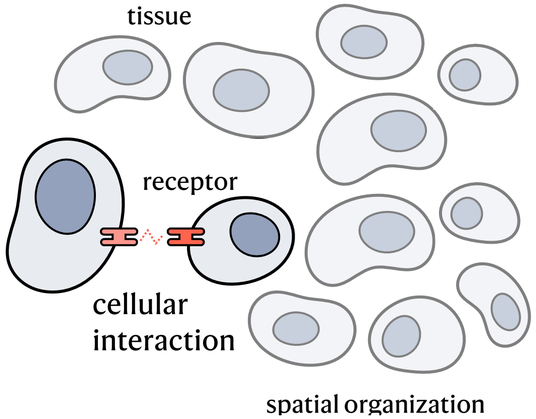
Cellular building blocks, environments, …

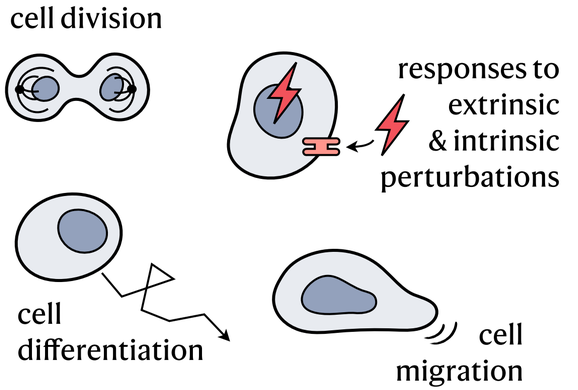
… behavior, and dynamics.

Molecular scale | Cellular scale | Multicellular scale





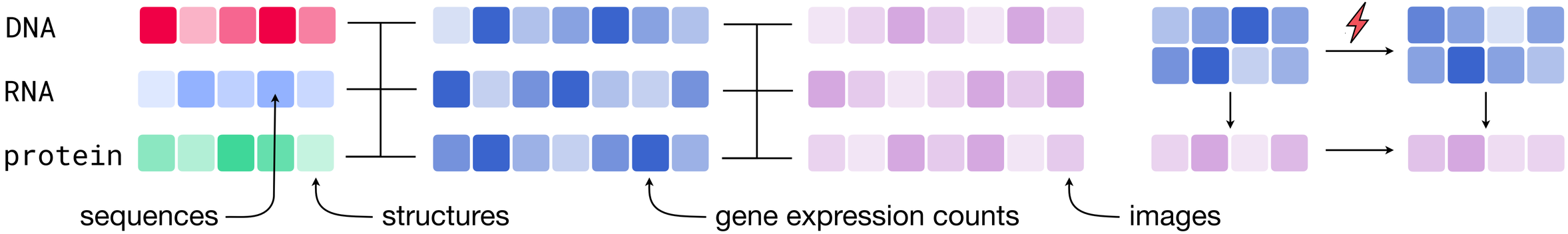




Tokenize across Biological Scales and Modalities …

… and Time

Tokens on Molecular Scale | Tokens on Cellular Scale | Tokens on Multicellular Scale



(Bunne et al., 2024)

Week 10’s Exercise Sheet

Exercise 8 · Task 1

CLIP: Understanding the Concepts

Understand how CLIP is trained through contrastive pretraining, describe how CLIP achieves zero-shot classification, understand effective prompting, and propose applications of CLIP beyond natural images and text.

Exercise 8 · Task 3

LLaVA: Understanding the Concepts

Explain how CLIP and LLaVA differ and how they relate, justify the need for a trainable projection layer between vision and language components and its trade-offs to full fine-tuning, and describe how these architectures can be adapted to “chat” with biological data like single-cell RNA sequences.

Week 10’s Code Demonstration

CLIP: Zero-shot Classification and Latent Space Analysis

Code Notebook 8 · Task 2

Implementing CLIP zero-shot classification, exploring prompt engineering strategies, analysing the embedding space structure, and testing CLIP's compositional understanding and attribute binding capabilities.

LLaVA: Implementation and Attention Evaluation

Code Notebook 8 · Task 4

Implementing and prompting LLaVA 1.6 for visual question answering, extracting and analysing attention patterns from the model, and examining how attention focuses on specific inputs depending on input context or predicted token.

CS-461

Foundation Models and Generative AI

Have a great week!