模式分析与机器智能
工业和信息化部重点实验室
MIIT Key Laboratory of
Pattern Analysis & Machine Intelligence

ParNeC 模式识别与神经计算研究组
PAttern Recognition and NEural Computing

Multimodal Models

3. Multi-modal Large Language Model

Background

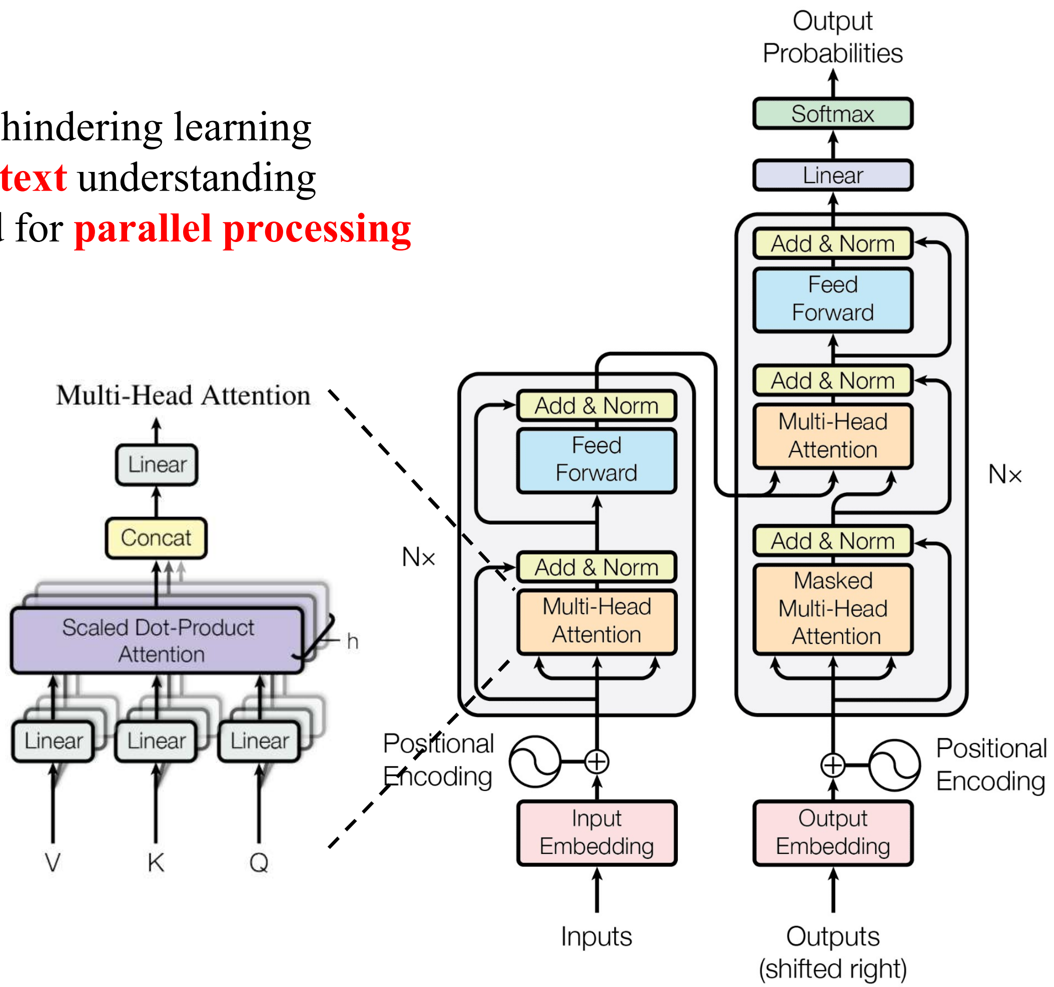
Transformer

Drawbacks of RNN and LSTM:

- Prone to vanishing and exploding gradient problems, hindering learning
- More biased toward recent data instead of **global context** understanding
- Not able to take advantage of modern GPUs designed for **parallel processing**

Main contributions of the Transformer paper:

- Encoder-decoder structure
  Self attention and cross attention
- Multi-head attention
  Each head could attend to different feature
- Causal modeling
  Causal mask in the decoder prevents positions from attending to future positions
- Positional embedding
  Adding perturbation to the sequence to differentiate the order of tokens

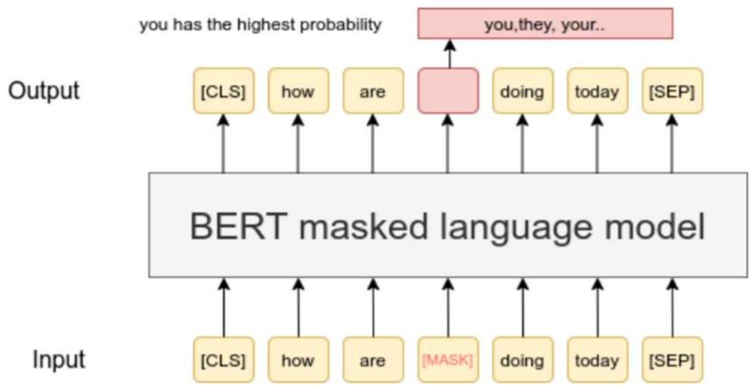


Attention is All You Need (June 2017 Google)

Background

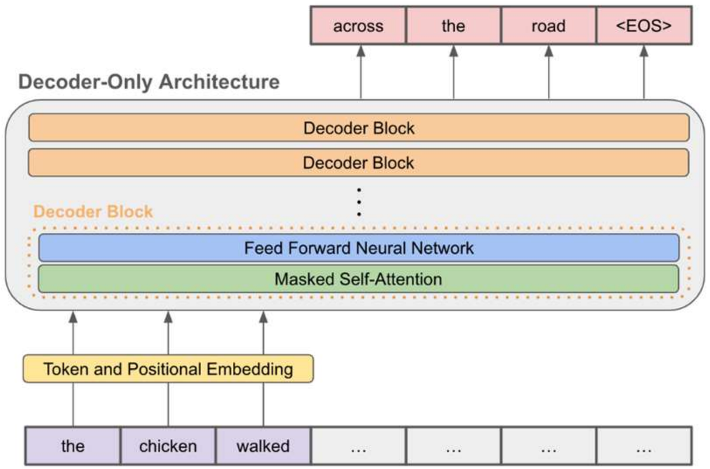
Different Architecture for Different NLP Tasks

BERT – Masked Language Modeling (Encoder only)



Pros:
Better understanding for global information. Suitable for almost every NLP task like sentiment analysis, token classification etc.
Cons:
Fixed context window size.
Not suitable for generative tasks.
High annotation cost.

GPT – Causal Language Modeling (Decoder only)



Pros:
Best choice for generative tasks.
Extrapolable context window size.
Cons:
Limited understanding for global context.

BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding (October 2018)
GPT: Improving Language Understanding by Generative Pre-Training (June 2018)

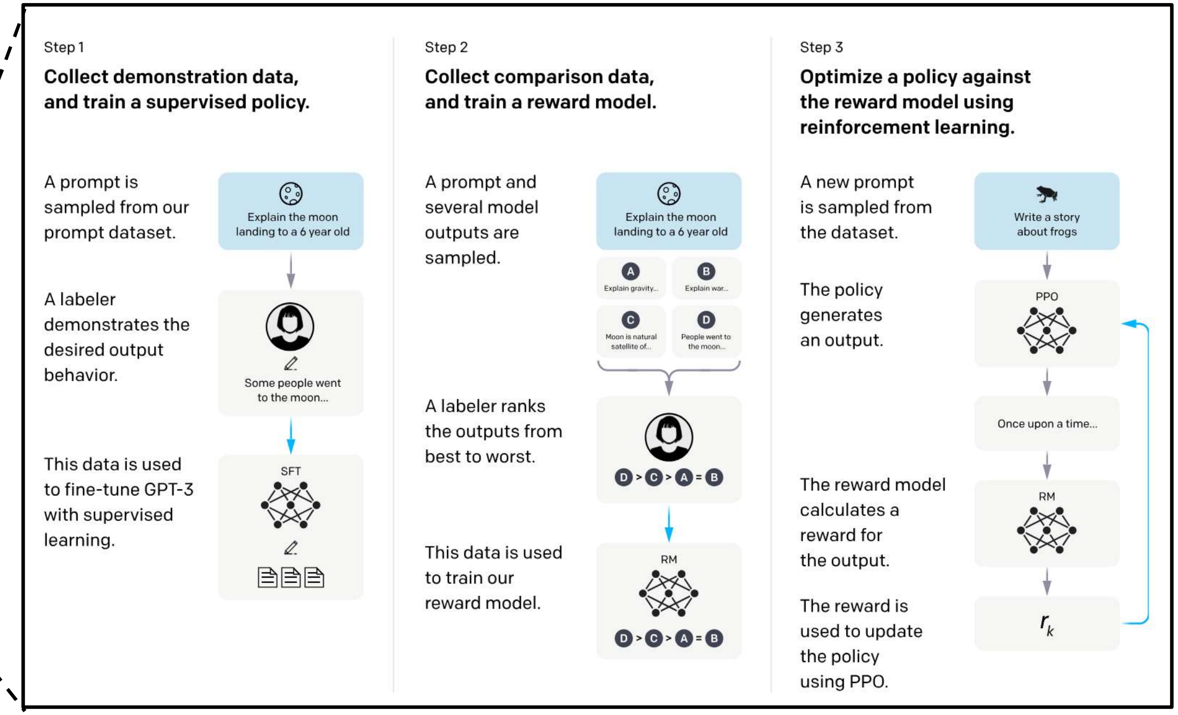
Background

Training Steps for LLMs

- Pretraining
  Use causal loss to perform next-token prediction training on large corpus of data
- Instruction tuning
  Supervised training on instruction tuning dataset to make LLM better follow users' instructions

$$L_1 = -\sum_t \log P_\theta(x_{t+1} \mid x_{t:1}),$$

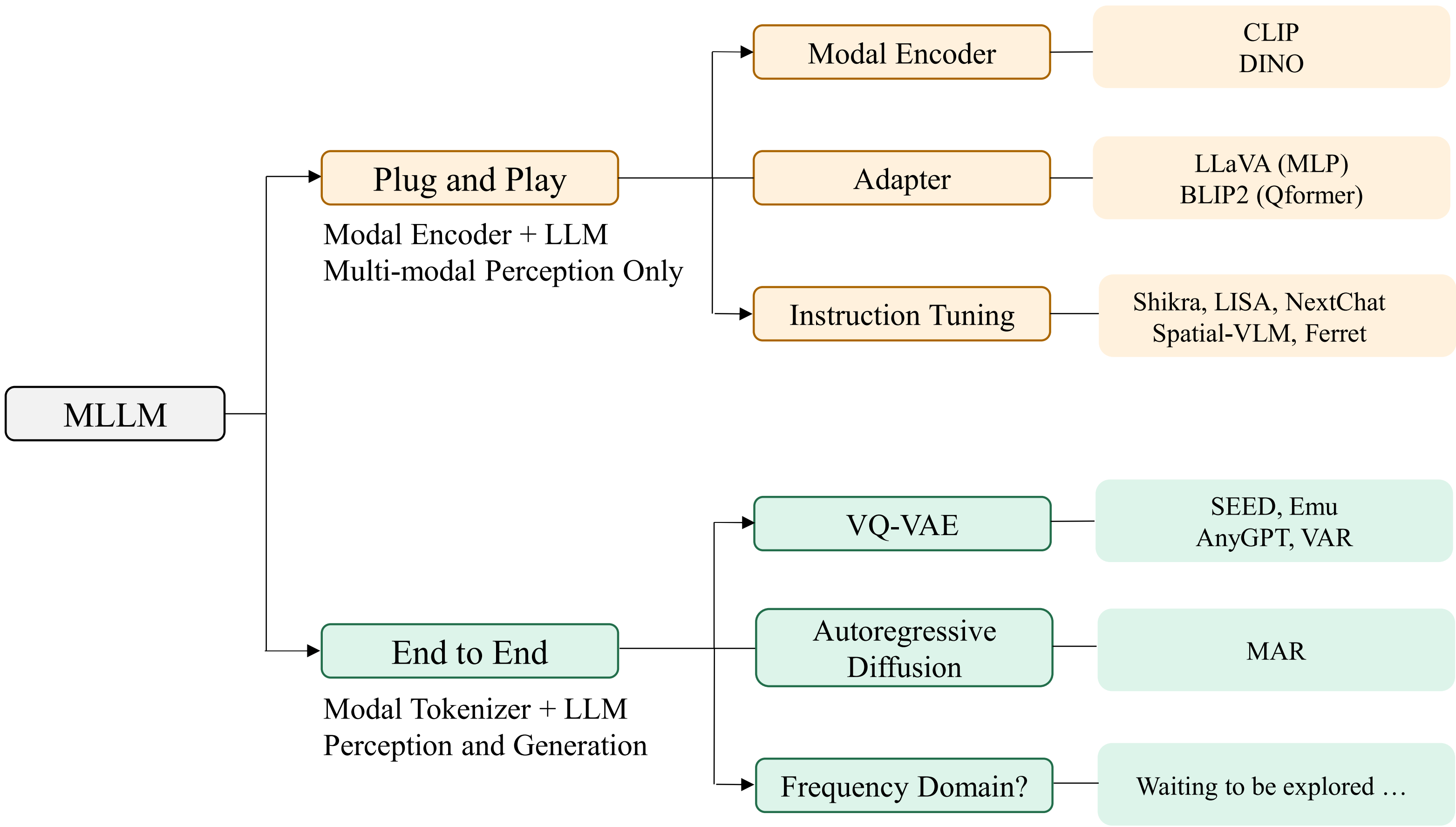
- Alignment. LLM should generate contents that align with human's preference (safety…)
  RLHF (PPO)
  DPO
  …



Training language models to follow instructions with human feedback (Mar 2022)

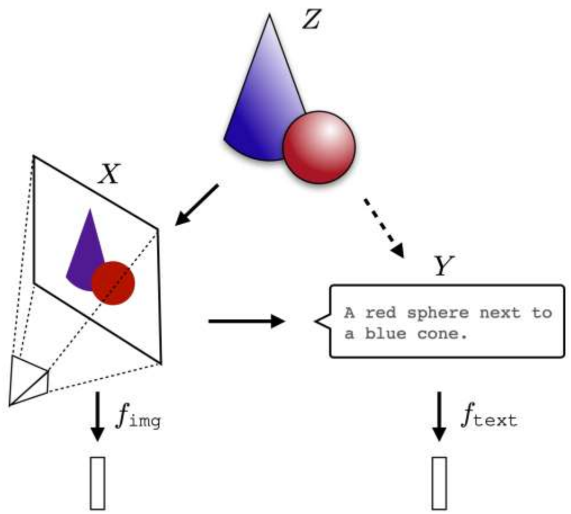
Introduction

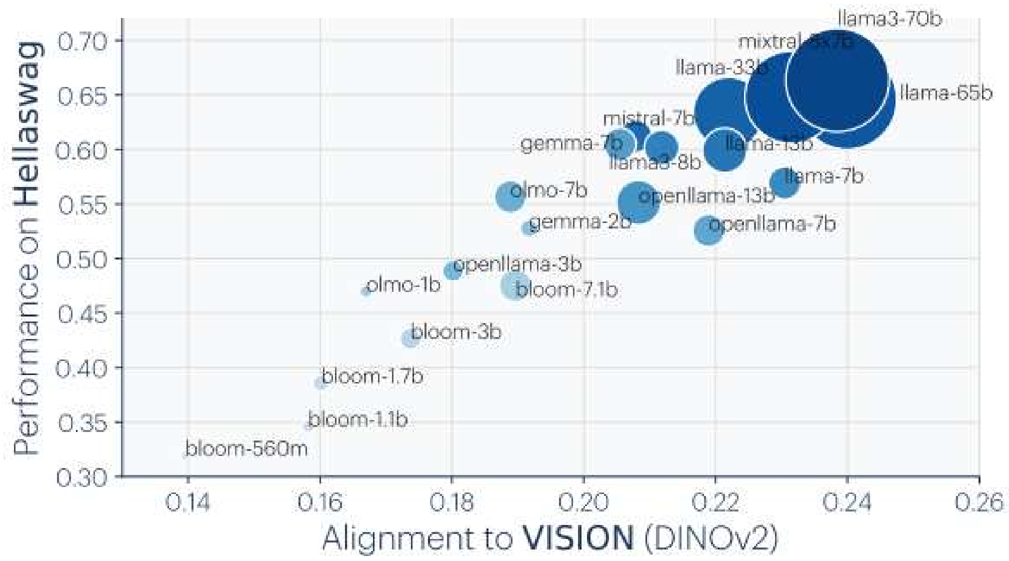
Content



Prerequisite

The Meta Question: Why Could Multi-modal Reasoning Exist?





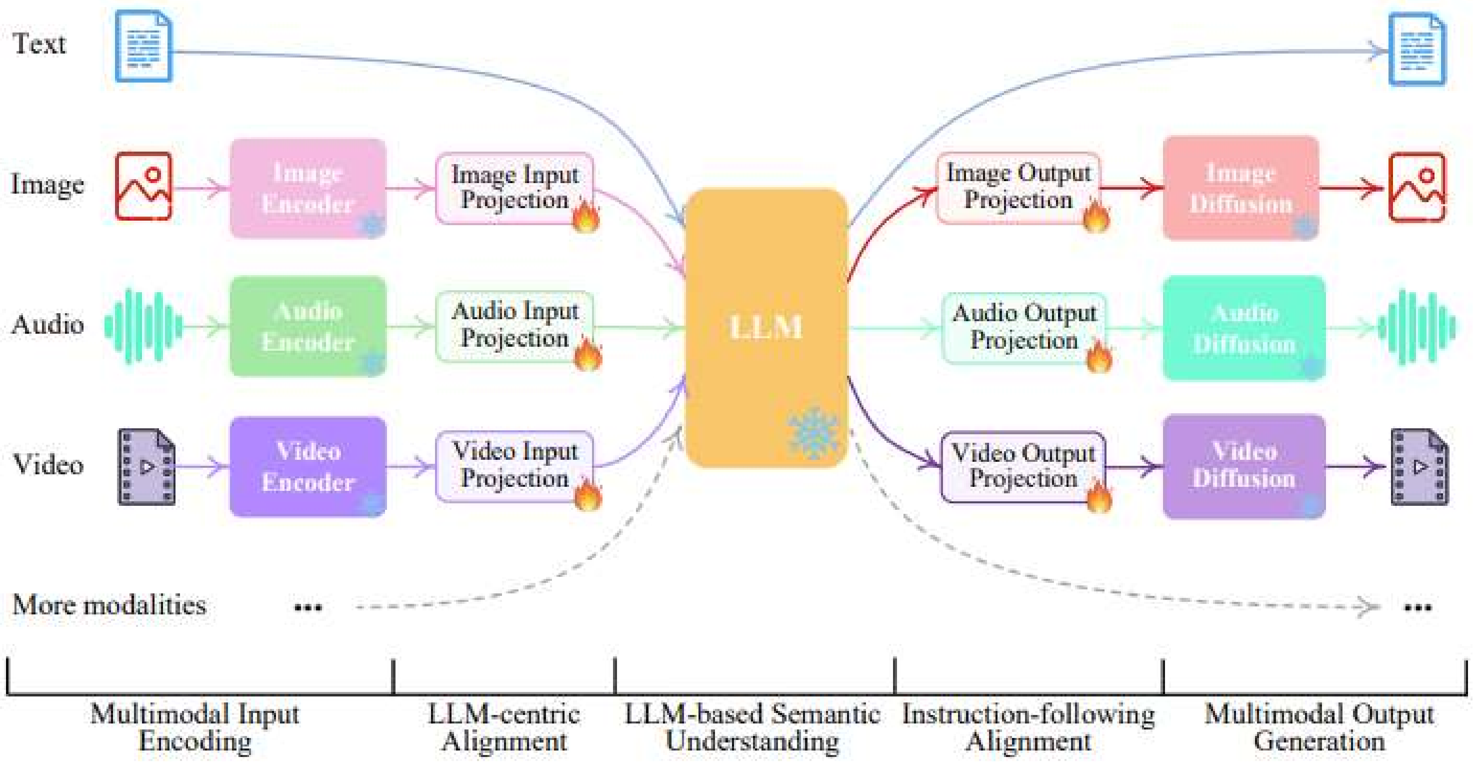
The representations across multiple modalities will converge on a shared representation of *Z*, and scaling model size, as well as data and task diversity, drives this convergence.

We can build MLLM using existing **visual encoder** and **LLM** with suitable adapter to deal with modal alignment.

The Platonic Representation Hypothesis (May 2024)

Plug and Play

ParNeC 模式识别与神经计算研究组 PAttern Recognition and NEural Computing

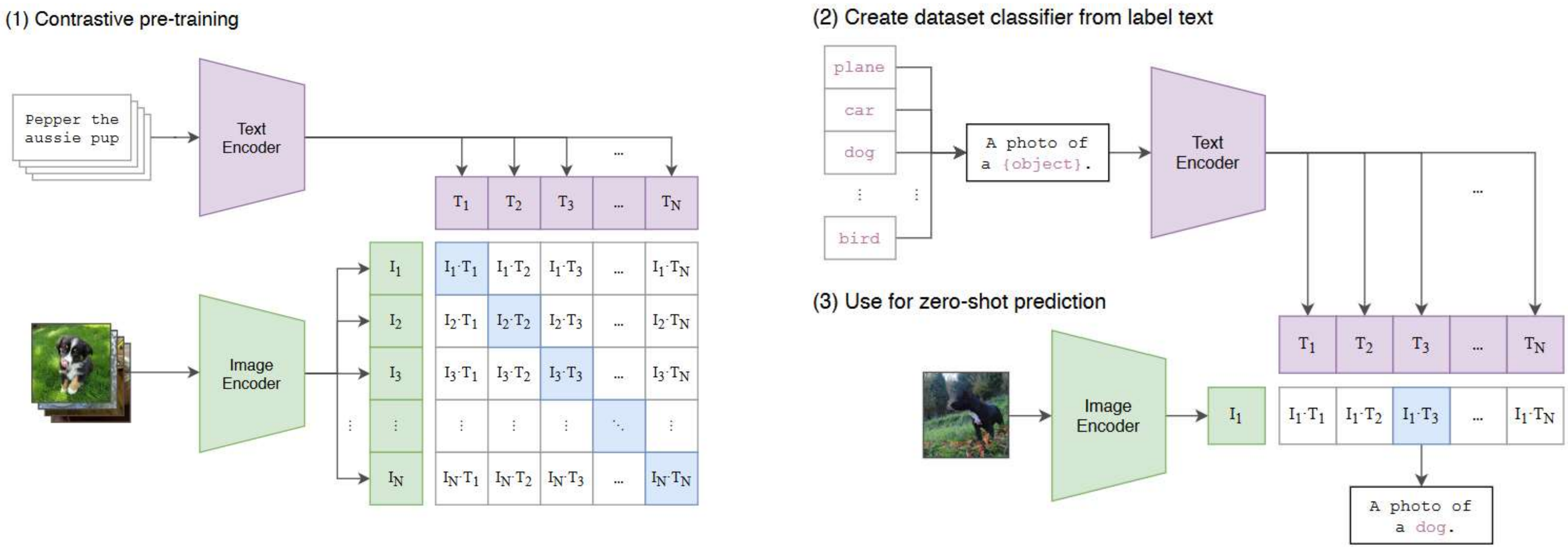


Connect an LLM with multimodal adaptors and different diffusion decoders, enabling it to perceive inputs and generate outputs in arbitrary combinations of **text**, **image**, **video**, and **audio**

NExT-GPT: Any-to-Any Multimodal LLM (Sep 2023)

Modal Encoder

CLIP



- ➢ Pretrained on 400 million image-text pairs just using **contrastive** loss
- ➢ Powerful **zero-shot** capability
- ➢ The most popular visual encoder for MMLM

Learning Transferable Visual Models From Natural Language Supervision (Feb 2021 OpenAI)

Modal Encoder

DINO

Motivation: The success of Transformers in NLP was the use of self-supervised pretraining (BERT, GPT)

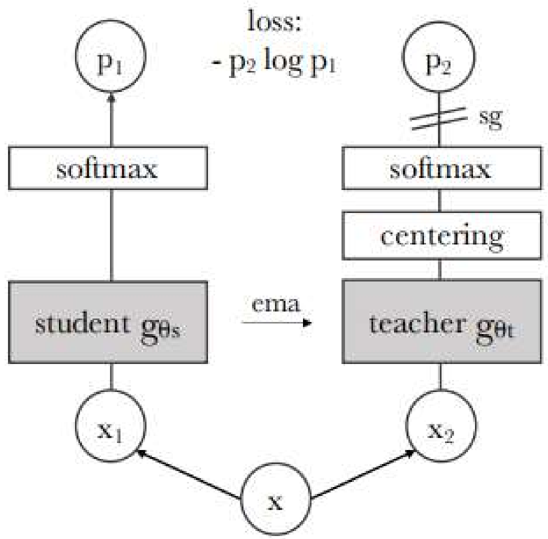


Figure 1: **Self-attention from a Vision Transformer with 8×8 patches trained with no supervision.** We look at the self-attention of the [CLS] token on the heads of the last layer. This token is not attached to any label nor supervision. These maps show that the model automatically learns class-specific features leading to unsupervised object segmentations.

From a given image, generate a set V of different views. This set contains two global views, x_1^g and x_2^g and several local views of smaller resolution. All **crops** are passed through the **student** while only the **global views** are passed through the **teacher.** Align the predictions of two networks on feature dimension K.

$$\text{loss\_v1} = \sum_{x\in\{x_1^g, x_2^g\}} \sum_{\substack{x'\in V \\ x'\neq x}} H(P_t(x), \boxed{P_s(x')}) \longrightarrow P_s(x)^{(i)} = \frac{\exp(g_{\theta_s}(x)^{(i)}/\tau_s)}{\sum_{k=1}^{K}\exp(g_{\theta_s}(x)^{(k)}/\tau_s)}$$

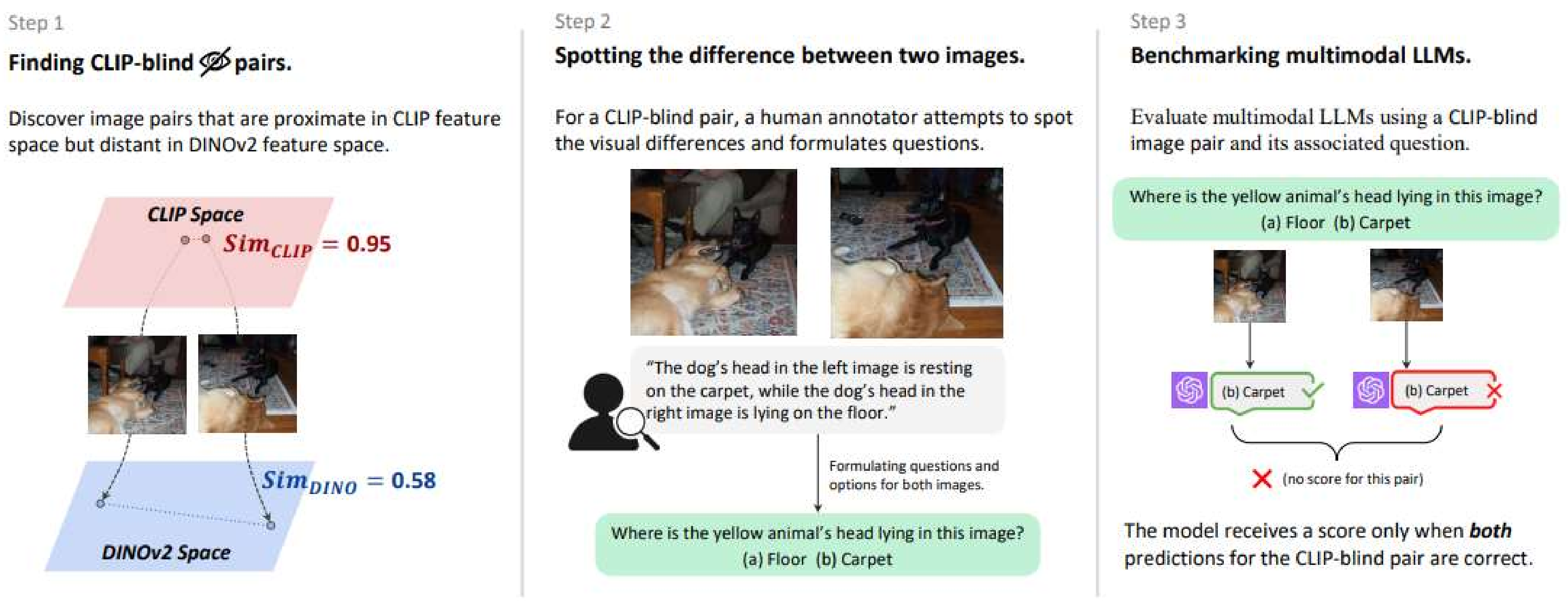
$$\text{loss v2} = \underbrace{\boxed{\text{loss\_v1}}}_{\text{Image-level}} + \underbrace{\boxed{\sum_{i=1}^{N} m_i \cdot P_{\theta'}^{\text{patch}}(\boldsymbol{u}_i)^{\mathrm{T}} \log P_{\theta}^{\text{patch}}(\hat{\boldsymbol{u}}_i)}}_{\text{Patch-level}}$$

DINO: Emerging Properties in Self-Supervised Vision Transformers (May 2021 Meta)
DINOv2: Learning Robust Visual Features without Supervision (Feb 2024 Meta)

Modal Encoder

CLIP v.s. DINO   How to Choose Between These Two?

Motivation: *CLIP-blind pairs* – images that CLIP perceives as similar despite their clear visual differences, is the main source of incorrect answers and hallucinated explanations.



DINO is more suitable for fine-grained level perception, object detection, semantic segmentation
Merging visual representations from CLIP and DINO t leads to improved performance in visual grounding tasks.

Eyes Wide Shut? Exploring the Visual Shortcomings of Multimodal LLMs (Apr 2024)

Adapter

LLaVA (MLP)

Motivation: Use a two-layer MLP to project visual feature into language space.

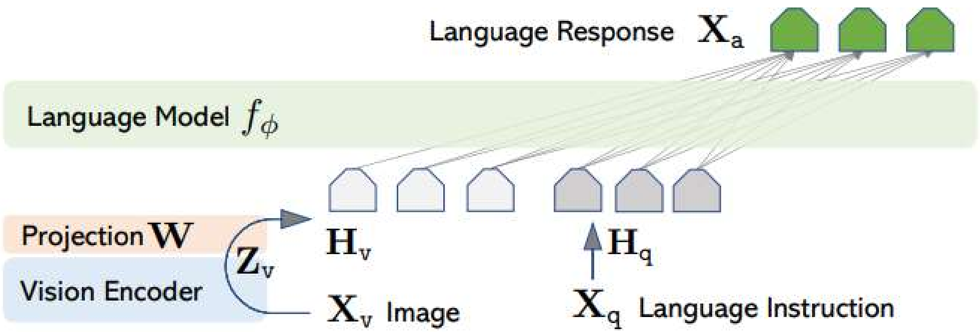
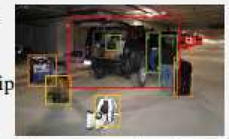


Figure 1: LLaVA network architecture.

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?
Answer: In the image, a group of people is standing outside a black SUV in a parking area, surrounded by various pieces of luggage, including suitcases and backpacks. They are facing the challenge of fitting all their luggage into the black SUV. There are multiple suitcases and backpacks to be packed, which suggests that the group has a significant amount of belongings ...<omitted>

Stage 1: Pre-training for Feature Alignment. Optimize MLP only on short VQA pairs.

Stage 2: Fine-tuning End-to-End. Use 150K GPT-generated multimodal instruction-following data, plus around 515K VQA data from academic-oriented tasks, to teach the model to follow multimodal instructions.

Visual Instruction Tuning (Dec 2023)

Adapter

BLIP2 (QFormer)

Motivation: Optimize learnable queries (Query2Label, Perceiver, finetuning CLIP) to merge and down-sample visual feature

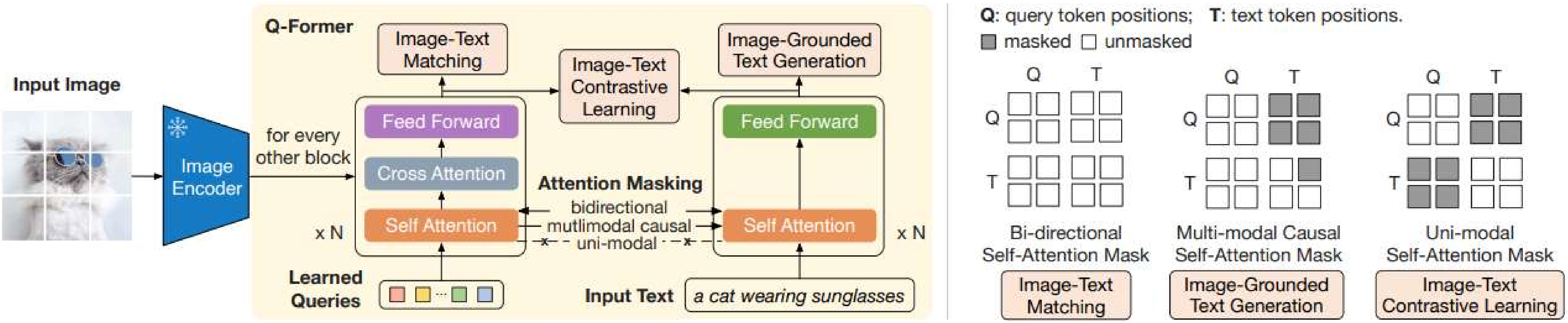


Image-Text Contrastive Learning (ITC) learns to align image representation and text representation by contrasting the image-text similarity of a positive pair against those of negative pairs in a batch.

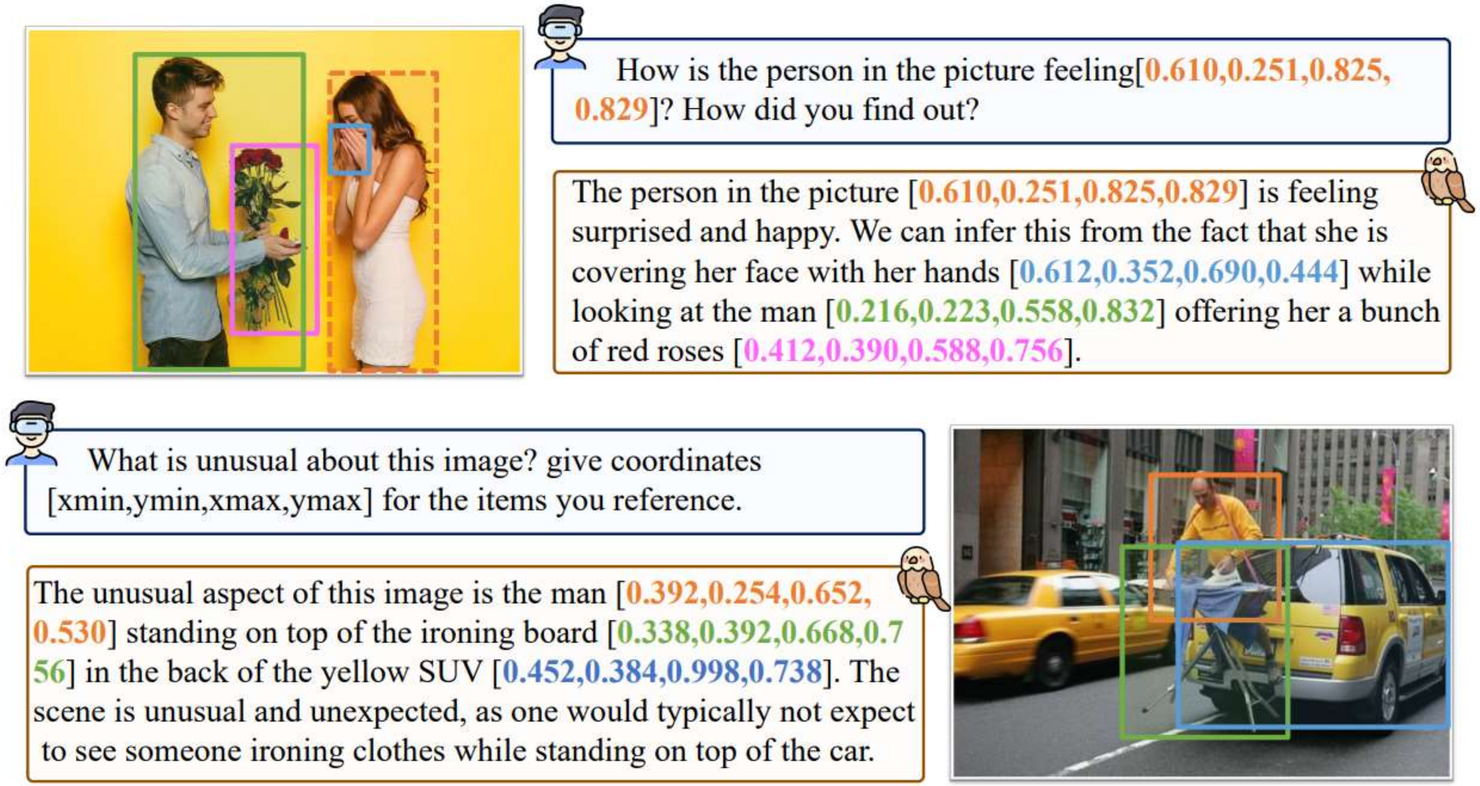
Image-grounded Text Generation (ITG) loss trains the Q-Former to generate texts, given input images as the condition.

Image-Text Matching (ITM) aims to learn fine-grained alignment between image and text representation. It is a binary classification task where the model is asked to predict whether an image-text pair is positive (matched) or negative (unmatched).

Instruction Tuning

Shikra

Motivation: Endow LLM with fine-grained perception capability like object detection, semantic segmentation



Add more object detection datasets in pretraining stage

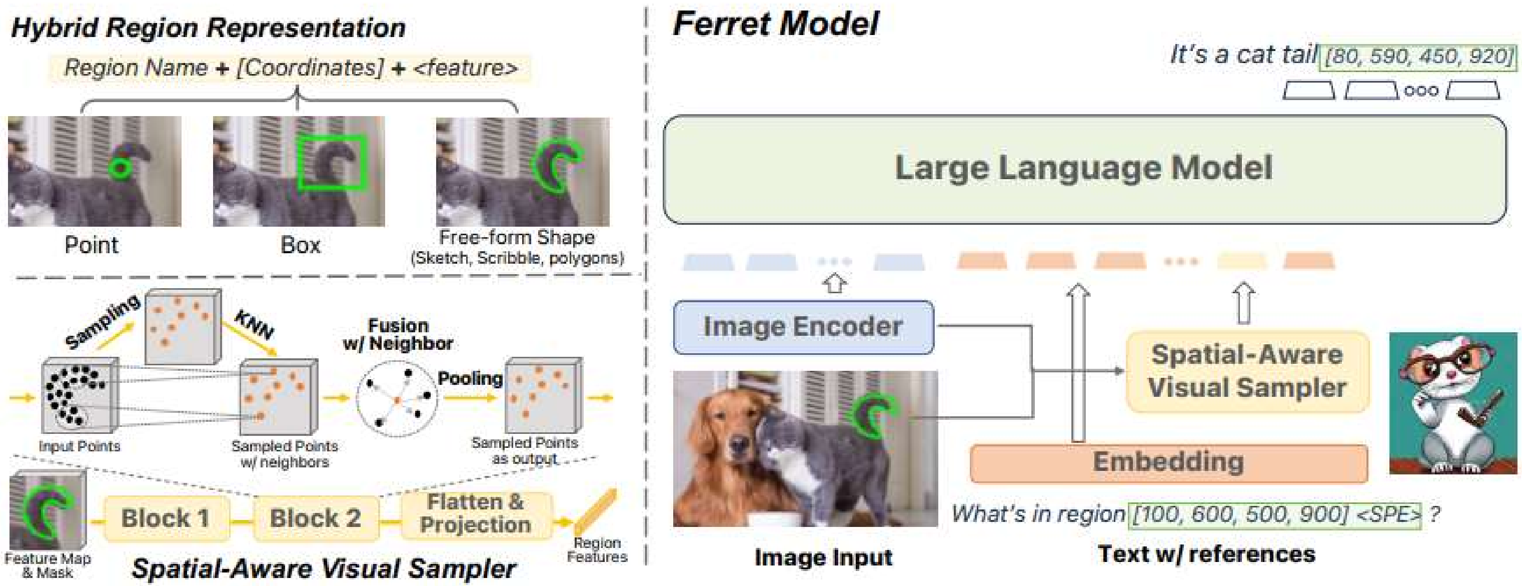
Use GPT4 to generate QA pairs with 2d bounding box to perform instruction tuning

Shikra: Unleashing Multimodal LLM's Referential Dialogue Magic (Jul 2023)

Instruction Tuning

Ferret

Motivation: Endow LLM with fine-grained perception capability like object detection, semantic segmentation



Spatial-aware visual sampler. The shape of the referred regions can be quite varied, not limited to just points or rectangle boxes. Given extracted image feature map Z and the binary region mask M, we first randomly sample N positive points inside M. For each point, its feature is obtained by bilinear interpolation.

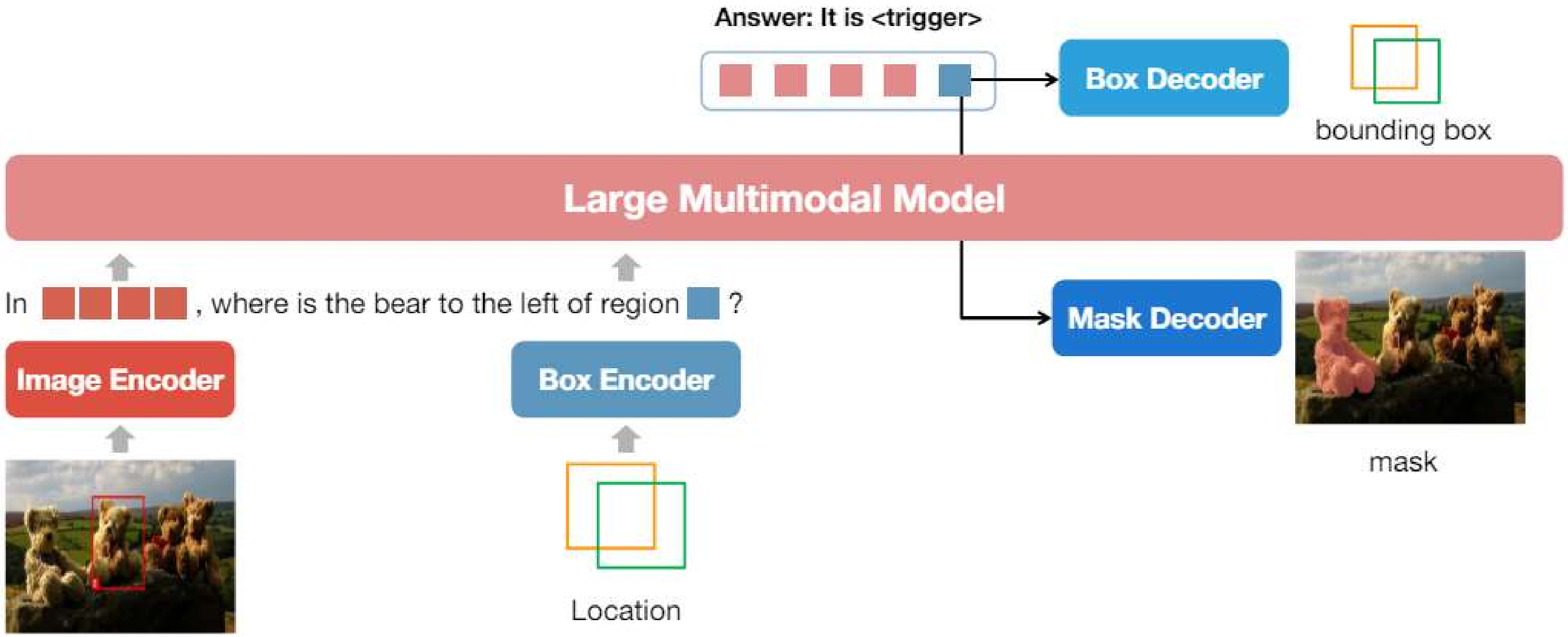
$$h_{ik} = \sigma([\theta([\mathbf{Z}(x_{ik}) - \mathbf{Z}(x_i); \overset{\text{2D coordinates of point x}}{\boxed{C(x_{ik})}} - C(x_i)]); \mathbf{Z}(x_i); C(x_i)]) \,,$$

Ferret: Refer And Ground Anything Anywhere at Any Granularity (Oct 2023)

Instruction Tuning

NeXT-Chat

Motivation: Endow LLM with fine-grained perception capability like object detection, semantic segmentation



Introduce a special token, denoted as `<trigger>` , which serves to trigger the localization (detection/segmentation)
Incorporate box encoder and decoder to extract fine-grained localization information
Connect with SAM to perform semantic segmentation

NExT-Chat: An LMM for Chat, Detection and Segmentation (Dec 2023)

End to End

One Unified MLLM Architecture

Text $\xrightarrow{\text{Tokenize}}$ Index in vocab $\xrightarrow{\text{Embedding}}$ Token vectors $\xrightarrow[\text{generation}]{\text{Auto-regressive}}$ $P(x_{i+1}|x_{0:i})$ over vocab $\xrightarrow[\text{technique}]{\text{Sampling}}$ Next token

How about using light-weight **modal tokenizer** to discrete feature in other modalities as well?

- Throw away modal adapter. LLM will process all modal data in one feature space.
- Generate image, text, audio… all in auto-regressive manner

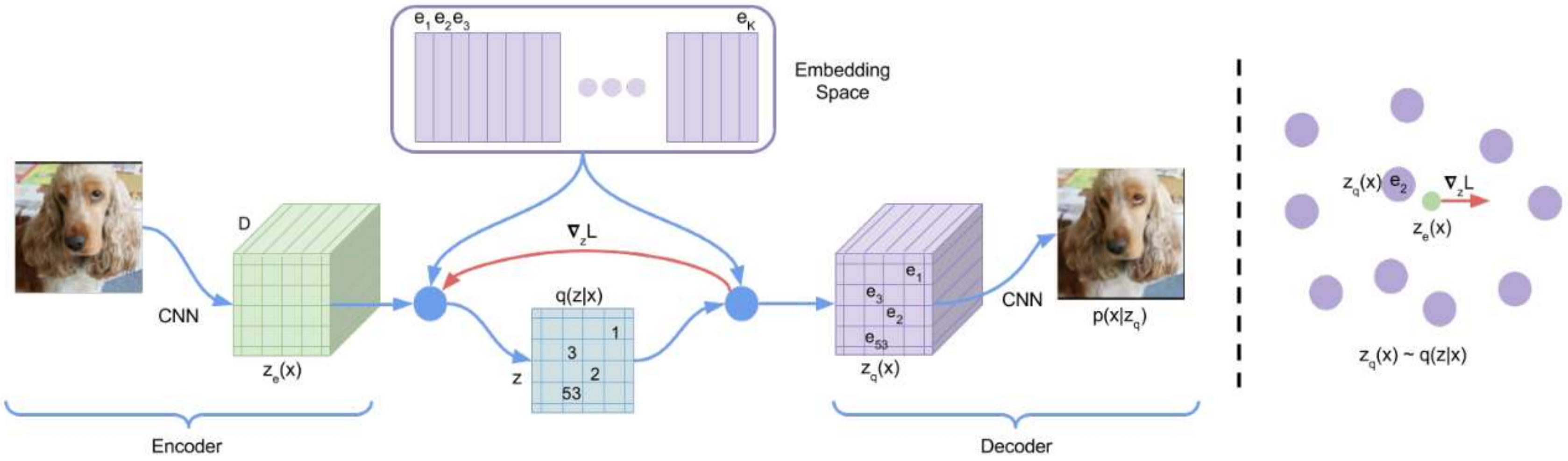
Tone perception, real-time conversation, change any voice type you like, image generation…

May 13 2024 GPT-4o Release

End to End

How to Tokenize Image? Pixel by Pixel? Slow and high cost!

Motivation: Vector Quantized-Variational-AutoEncoder (VQ-VAE), a simple yet powerful **generative** model that learns visual discrete representations, differs from VAEs in two key ways: the encoder network outputs discrete feature



Define a latent embedding space $e \in R^{K \times D}$ where K is the size of the discrete latent space (i.e., a K-way categorical), and D is the dimensionality of each latent embedding vector e_i. Take an image input x to calculate $z_e(x)$ from encoder, and then find the nearest e_i in codebook

$$z_q(x) = e_k, \quad \text{where} \quad k = \boxed{\text{argmin}_j \|z_e(x) - e_j\|_2} \xrightarrow{\text{No gradient}} \|x - decoder(z + sg[z_q - z])\|_2^2 + \beta\|sg[z] - z_q\|_2^2 + \gamma\|z - sg[z_q]\|_2^2$$

Neural Discrete Representation Learning (May 2018)

Can we use VQ-VAE as image tokenizer? Yes and No

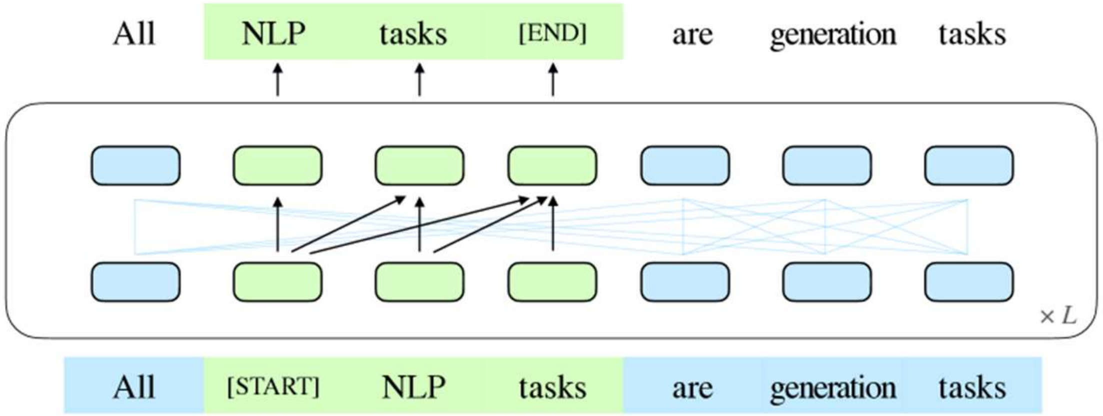
What Stands in The Way of A Unified MLLM Architecture?

Causal modeling for image

What's causality?

The system will only laugh if you tickle it…

Text/audio generation: the future token is dependent on past tokens. That's why we can generate text/audio sequence in a auto-regressive manner. (1D causal system)



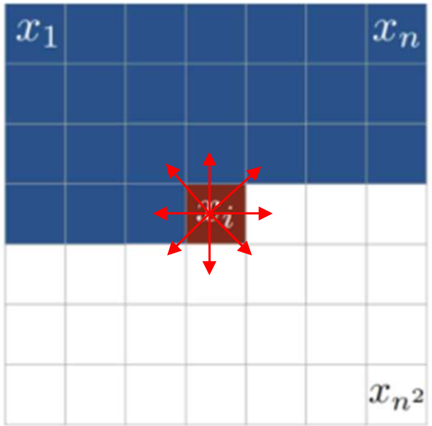


Image generation: A pixel in image space has relation with other 8 pixels in its neighborhood. So 1D causal modeling is a wrong representation for image. (2D semi-causal system)

How to define causality for image?

End to End

Causal Modeling for Image

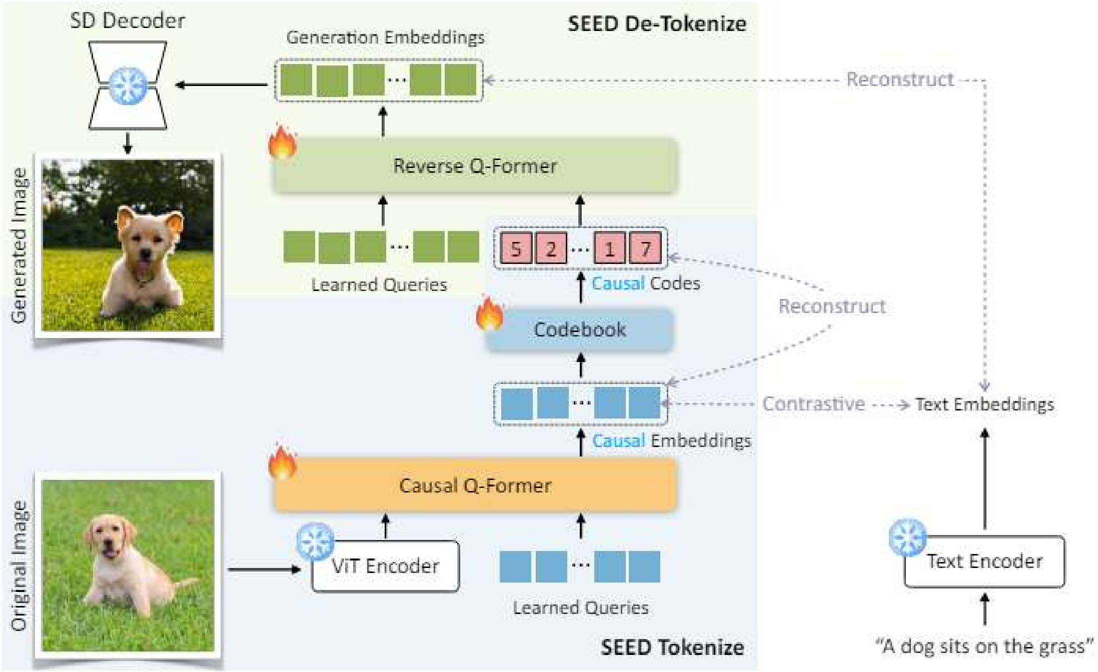
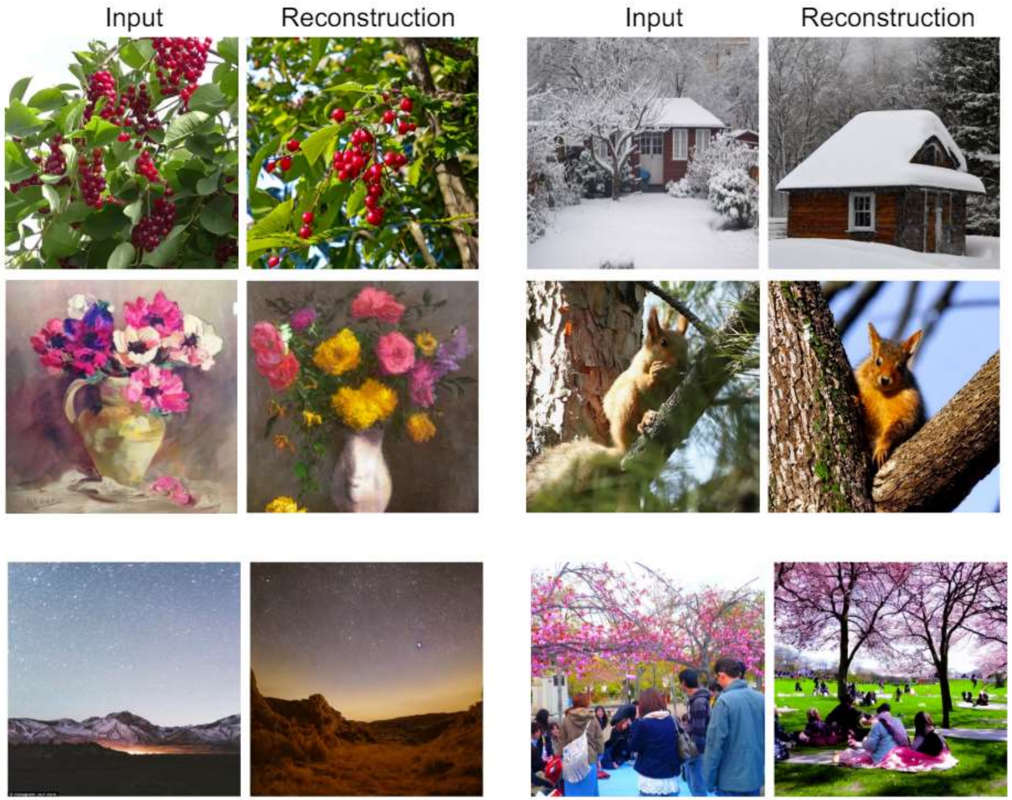


Figure 2: Overview of our **SEED** tokenizer, which produces discrete visual codes with causal dependency and high-level semantics.



Training Stage I: Causal Q-Former. Use causal mask in Q-Former to optimize learnable queries by contrastive loss (final token only).
Training Stage II: Visual Quantization and De-tokenization. Train a VQ codebook to discretize the causal embeddings and then employ a Reverse Q-Former to reconstruct the textual features of a frozen stable diffusion model from discrete codes.

Planting a SEED of Vision in Large Language Model (Aug 2023)

Encode image feature into 1D sequence, losing 2D spatial information!

End to End

Causal Modeling for Image

Motivation: autoregressive image generation can be viewed as coarse-to-fine "next-scale prediction" or "next-resolution prediction", diverging from the standard raster-scan "next-token prediction".

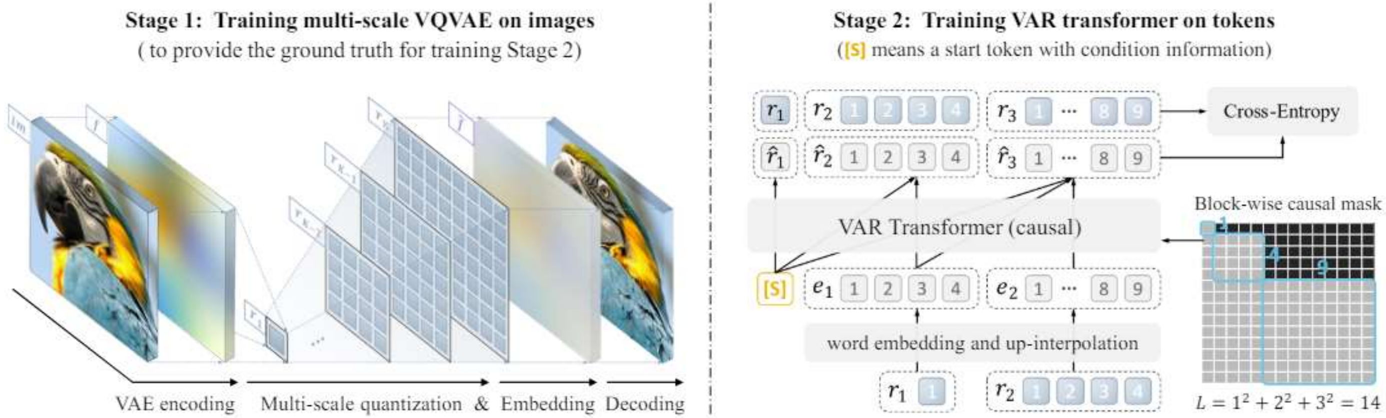


Figure 4: **VAR involves two separated training stages. Stage 1:** a multi-scale VQ autoencoder encodes an image into K token maps $R = (r_1, r_2, \ldots, r_K)$ and is trained by a compound loss (5). For details on "Multi-scale quantization" and "Embedding", check Algorithm 1 and 2. **Stage 2:** a VAR transformer is trained via next-scale prediction (6): it takes $(\texttt{[s]}, r_1, r_2, \ldots, r_{K-1})$ as input to predict $(r_1, r_2, r_3, \ldots, r_K)$. The attention mask is used in training to ensure each r_k can only attend to $r_{\leq k}$. Standard cross-entropy loss is used.

The autoregressive unit should be an entire token map, rather than a single token. Quantize a feature map $f \in \mathbb{R}^{h \times w \times C}$ into K multi-scale token maps $(r_1, r_2, \ldots, r_K)$, each at a increasingly higher resolution $h_k \times w_k$, culminating in r_K matches the original feature map's resolution $h \times w$.

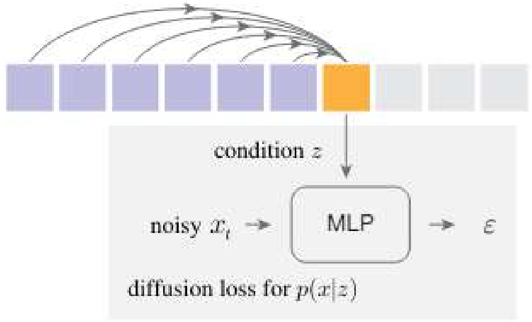
$$p(r_1, r_2, \ldots, r_K) = \prod_{k=1}^{K} p(r_k \mid r_1, r_2, \ldots, r_{k-1}),$$

Visual Autoregressive Modeling: Scalable Image Generation via Next-Scale Prediction (Jun 2024)

End to End

Does Image Modeling Have to Be Discrete?

Motivation: model the per-token probability distribution using a diffusion procedure, which allows us to apply autoregressive models in a continuous-valued space using *Diffusion Loss*.



$$\mathcal{L}(z, x) = \mathbb{E}_{\varepsilon,t}\left[\|\varepsilon - \varepsilon_\theta(x_t|t, z)\|^2\right]$$

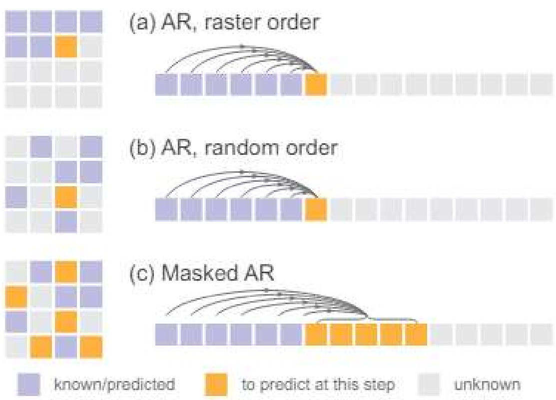


Figure 3: **Generalized Autoregressive Models.** (a) A standard, raster-order autoregressive model predicts one next token based on the previous tokens. (b) A random-order autoregressive model predicts the next token given a random order. It behaves like randomly masking out tokens and then predicting one. (c) A Masked Autoregressive (MAR) model predicts multiple tokens simultaneously given a random order, which is conceptually analogous to masked generative models [4, 29]. In all cases, the prediction of one step can be done by causal or bidirectional attention (Figure 2).

Instead of using diffusion models for representing the joint distribution of all pixels or all tokens, in our case, the diffusion model is for representing the distribution for **each token**. Similar to MAE, predict multiple tokens based on previous tokens.

$$p(x^1, ..., x^n) = p(X^1, ..., X^K) = \prod_k^K p(X^k \mid X^1, ..., X^{k-1}).$$

$X^k = \{x^i, x^{i+1} ..., x^j\}$ is a set of tokens to be predicted at the k-th step

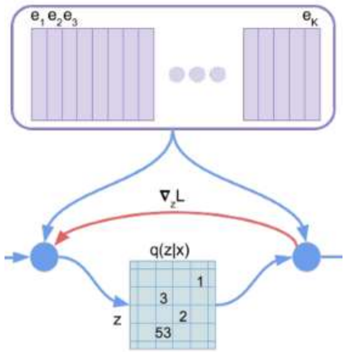
MAE + Diffusion

Use diffusion to replace MAE decoder

Autoregressive Image Generation without Vector Quantization (Jun 2024)

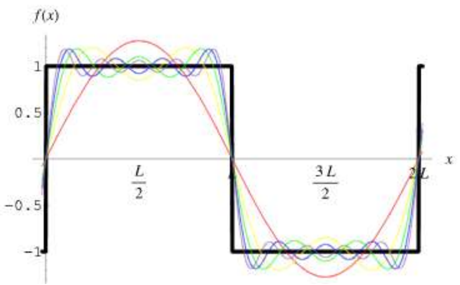
End to End

How About Frequency Domain?



Hard to ensure the independence among vectors (bases) in codebook
Not an efficient information encoding technique, leading to detail loss in image
Still unclear how to define causality in image space

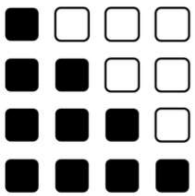
Use FFT to transform images into frequency domain.



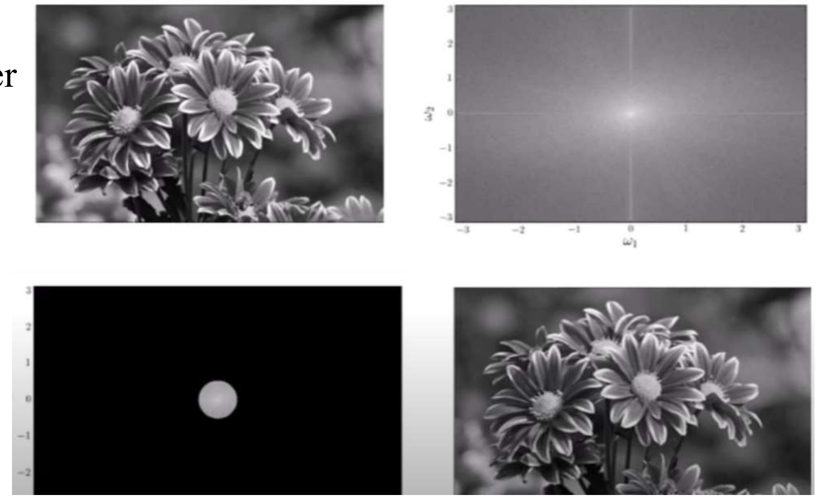
$$F(k,l) = \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} f(i,j)\, e^{-\iota 2\pi\left(\frac{ki}{N}+\frac{lj}{N}\right)}$$

- Sinusoidal bases are linear independent to each other
- In 2D image space, there is a **causal** relationship between low frequency basis (blur) and high frequency (clear) basis.

$$f_q = e_1 f_1 + e_2 f_2 + e_3 f_3 + \cdots + e_n f_n$$

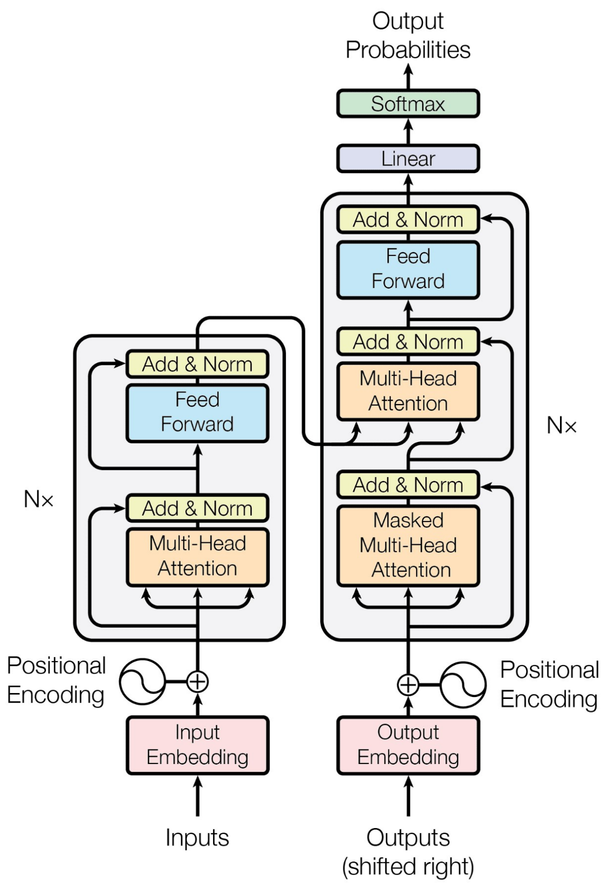


Codebook only consists of coefficients



Alternative

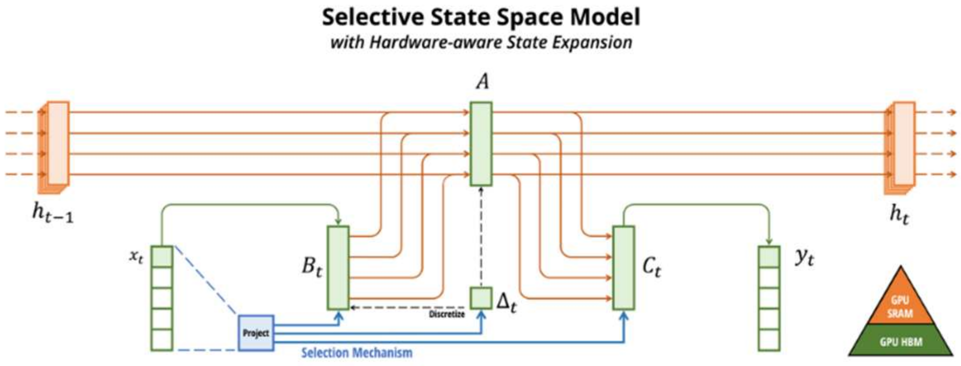
How Long Will Transformer Dominate?



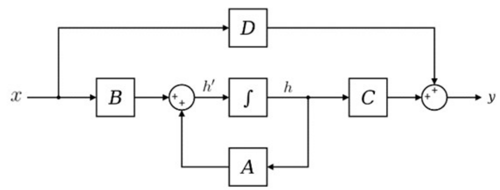
- Simple but efficient. The basic operation is matrix multiplication, making parallel computing possible. (ZeRO, Megatron…)
- Suitable for almost any modal data.

$O(L^2)$ inference cost. Struggle to achieve long-context window understanding. (KV-Cache, GQA, RoPE, NTK-aware interpolation …)

Any challenger?



Technical structure of Mamba



$$h'(t) = Ah(t) + Bx(t) \quad (1a)$$
$$y(t) = Ch(t) \quad (1b)$$

$$h_t = \overline{A}h_{t-1} + \overline{B}x_t \quad (2a)$$
$$y_t = Ch_t \quad (2b)$$

$$\overline{K} = (C\overline{B}, C\overline{AB}, \dots, C\overline{A}^k\overline{B}, \dots) \quad (3a)$$
$$y = x * \overline{K} \quad (3b)$$

Mamba: Linear-Time Sequence Modeling with Selective State Spaces (May 2024)

Linear-Time Sequence Modeling

The Bitter Lesson

Next-Token Prediction: The Path to AGI?

In computer vision, there has been a similar pattern. Early methods conceived of vision as searching for edges, or generalized cylinders, or in terms of SIFT features. But today all this is discarded. Modern deep-learning neural networks use only the notions of **convolution** and certain kinds of invariances, and perform much better.

- AI researchers have often tried to build knowledge into their agents
- this always helps in the short term, and is personally satisfying to the researcher,
- in the long run it plateaus and even inhibits further progress,
- breakthrough progress eventually arrives by an opposing approach based on scaling computation by search and learning.

We should build in only the meta-methods that can find and capture this arbitrary complexity. Essential to these methods is that they can find good approximations, but the search for them should be by our methods, not by us. We want AI agents that can discover like we can, not which contain what we have discovered.

Thanks