Vision Language Models

Himanshu Singh

Github

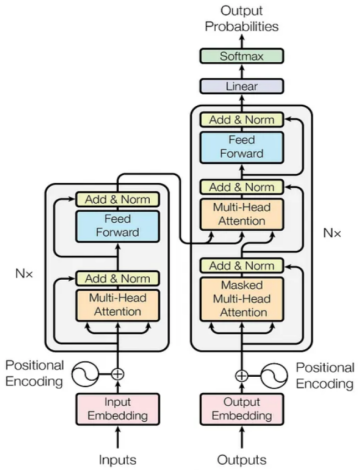
Important Concepts

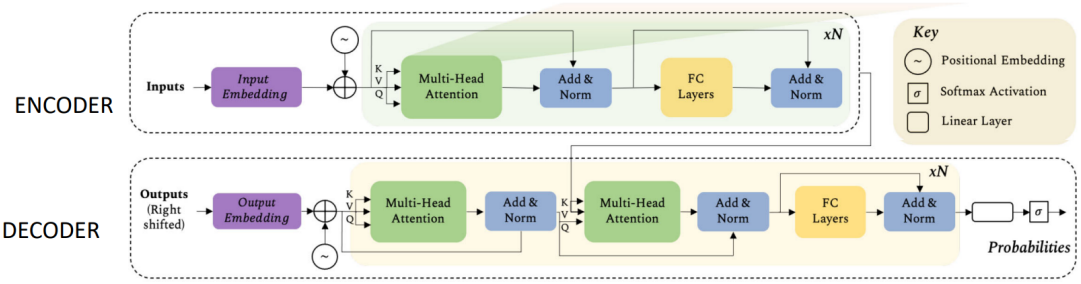
Supervised: Model is trained on a labeled dataset
Self-Supervised: Model generates its own labels from the input data
Zero-Shot: Model makes predictions about new classes that it hasn't seen during training
Few shot: Model learns from only a very small number of training example
Image Encoder: VIT CLIP...
Text Encoder: T5, BERT...
LLM: GPT, LaMA
Loss or Objective function: Contrastive, Captioning, Image-Text Matching
Tasks: Recognition(Accuracy), Retrieval(T2I, I2T : Recall), VQA (Accuracy), Captioning (BLUE,...)

Transformer

Used for modelling long dependencies between input sequence elements. Allows processing multiple modalities (images, videos, text, speech) using similar processing blocks.
Pre-trained using pretext tasks on large-scale (unlabelled) datasets.
Vision Applications:

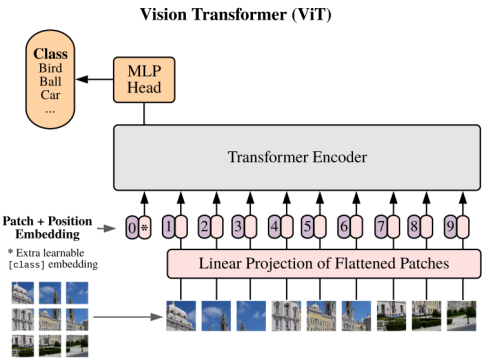
- Recognition Tasks (Image Classification, Object Detection, action recognition and segmentation)
- Generative modelling, multi-modal tasks (Visual Question answering, visual reasoning)
- Video Processing (Activity recognition, Video forecasting)
- Low-level Vision (Image super resolution, Image enhancement)
- 3D Analysis (Point cloud classification)





Vision Transformer

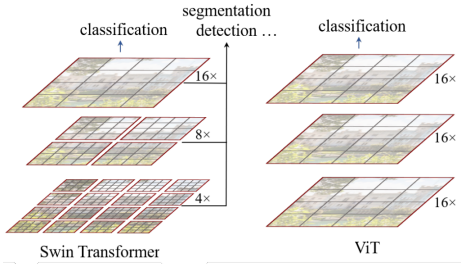
Take 16×16 patch, form an embedding of dimension 768. Can be done with single convolution (in channels = 3, out channels = 768, kernel size = (16,16), stride = (16,16))
For a 224×224 RGB image, the output will be 196×768, add 1 classification token to it. Resultant embedding will be 197×768 dimensional

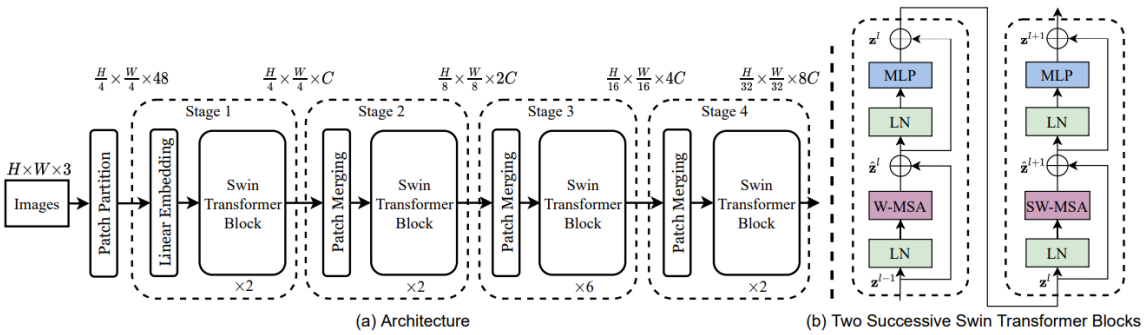


SWIN Transformer

"Shifted Windows"
A hierarchical transformer who representation is computed with shifted windows. Shifted windowing limit attention:

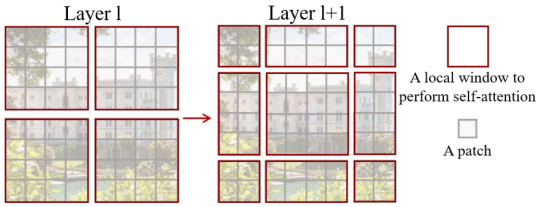
- To local window
- Allowing cross window connections





Initially, you have patches of 4×4 RBG channel, you pass them through linear embedding, which convert them to C-dimensional vector. In patch embedding you merge the 2×2 neighbouring patches into 1 while increasing the embedding dimension by twice.

The paper uses W-MSA instead of MSA, because computing attention of every patch with every other patch will be computationally expensive, as well we don't need that much of global information. Instead in W-MSA/SW-MSA, we compute self-attention within a window. For SW-MSA, the paper proposes cyclic shift of patches.

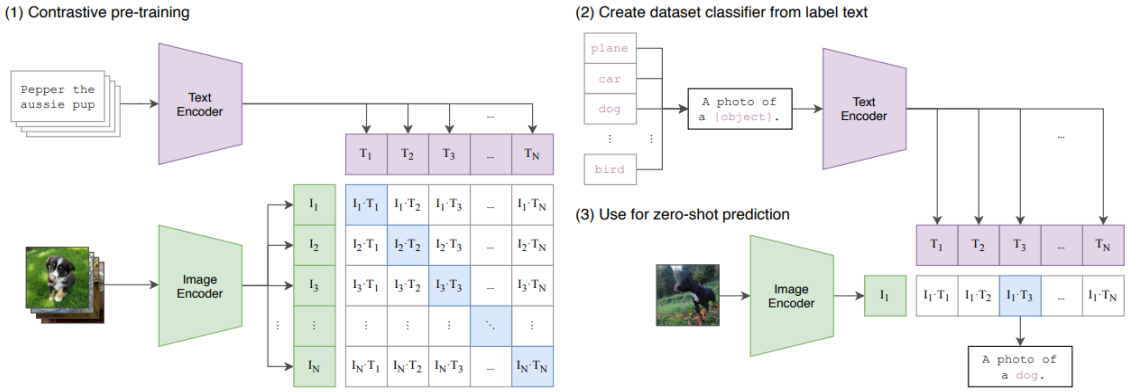


$$\hat{z}^{l} = \text{W-MSA}\left(\text{LN}\left(\hat{z}^{l-1}\right)\right) + \hat{z}^{l-1}$$
$$z^{l} = \text{MLP}\left(\text{LN}\left(\hat{z}^{l}\right)\right) + \hat{z}^{l}$$

$$\hat{z}^{l+1} = \text{SW-MSA}\left(\text{LN}\left(\hat{z}^{l}\right)\right) + \hat{z}^{l}$$
$$z^{l+1} = \text{MLP}\left(\text{LN}\left(\hat{z}^{l+1}\right)\right) + \hat{z}^{l+1}$$

Contrastive Language Image Pre-Training

Pair an image with its caption using contrastive learning.



m_i = one-hot encoded label vector for i^{th} image sample

y_i^m = cosine similarities vector for i^{th} image sample

t_i = one-hot encoded label for i^{th} text sample

y_i^t = cosine similarities vector for i^{th} text sample

ϕ = cross entropy loss

$$\mathcal{L}_m = \frac{\sum_{i=1}^{N} \phi\left(y_i^m, m_i\right)}{N} \qquad \mathcal{L}_t = \frac{\sum_{i=1}^{N} \phi\left(y_i^t, t_i\right)}{N}$$

$$\mathcal{L} = \frac{\mathcal{L}_m + \mathcal{L}_t}{2}$$

Supervised Learning	Zero Shot Learning
Labelled Data	Data can be labelled/unlabelled
Training Phase	No Training
Final Prediction on Labelled data	Accuracy on the final result

Motivation: Image classification models are limited with fixed number of labels and Generalization. CLIP overcomes these limitations.

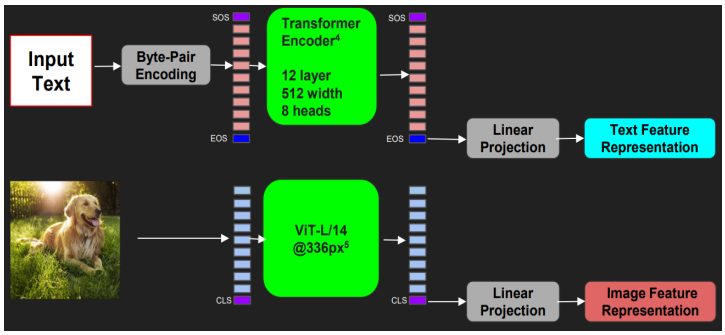
Training:

- Trained on 400M image-text pairs from internet.
- Batch size 32768
- 32 epochs over the dataset

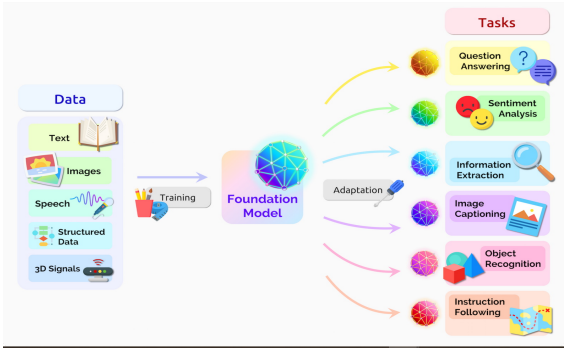
- Cosine learning rate decay

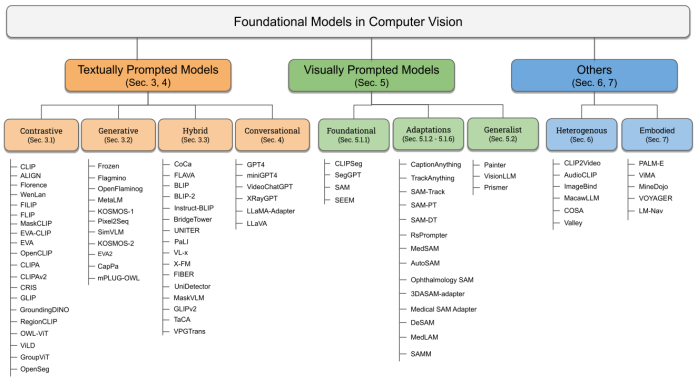
Architecture:

- ResNet-based or ViT-based image encoder
- Transformer based text encoder



Foundation Models: AI Neural network trained on mountain of raw data generally with unsupervised learning.





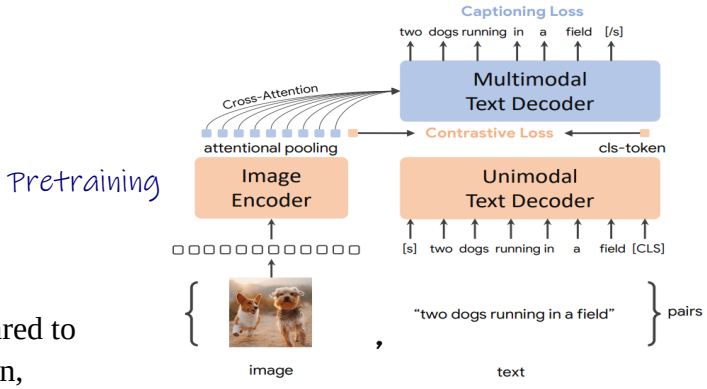
Contrastive Captioners (CoCa)

Contrastive Captioners Pre-training

$$\mathcal{L}_{CoCa} = \lambda_{con}\mathcal{L}_{con} + \lambda_{cap}\mathcal{L}_{cap}$$

Encode Decoder Captioning

$$\mathcal{L}_{cap} = -\sum_{t=1}^{T} \log P_\theta(y_t | y_{<t}, x)$$

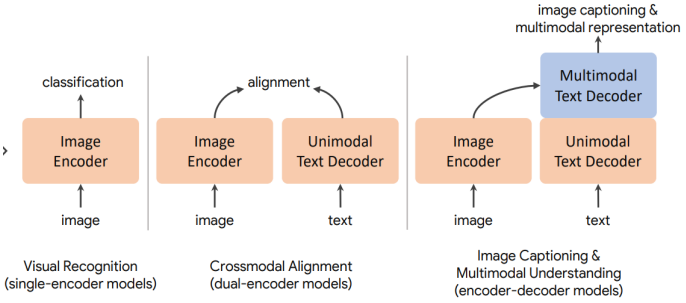


Dual Encoder Contrastive Learning: Compared to pretraining with single-encoder classification, which requires human-annotated labels and data cleaning, the dual-encoder approach exploits noisy web-scale text descriptions and introduces a learnable text tower to encode free-form texts

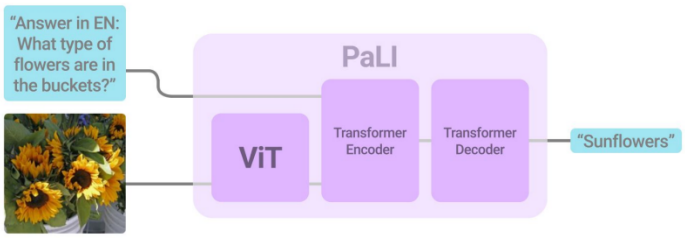
$$\mathcal{L}_{CON} = -\frac{1}{N}\left(\underbrace{\sum_{i=1}^{N}\log\frac{\exp(x_i^T y_i/\sigma)}{\sum_{j=1}^{N}\exp(x_i^T y_j/\sigma)}}_{\text{image-to-text}} + \underbrace{\sum_{i=1}^{N}\log\frac{\exp(y_i^T x_i/\sigma)}{\sum_{j=1}^{N}\exp(y_i^T x_j/\sigma)}}_{\text{text-to-image}}\right)$$

where x_i and y_j are normalized embeddings of the image in the i^{th} pair and that of the text in the j^{th} pair

The pretrained CoCa can be used for downstream tasks including visual recognition, vision-language alignment, image captioning and multimodal understanding with zero-shot transfer, frozen-feature evaluation or end-to-end finetuning.



Pathways Language and Image (PaLI)

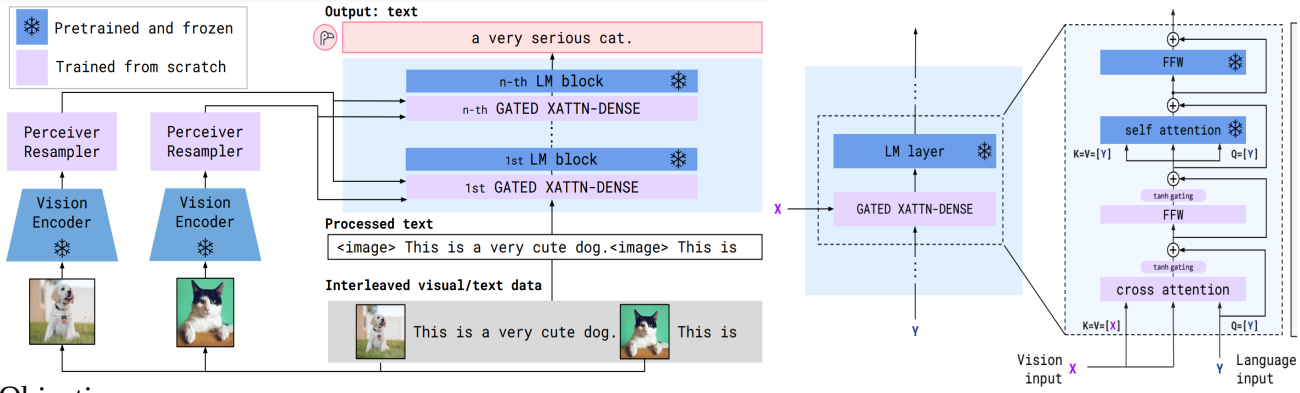


Training Mixture:

- Span corruption on text only data
- Split captioning on WebLI alt-text data
- Captioning on CC3M-35L
- OCR on WebLI OCR-text data
- English and Cross lingual VQA
- English and Cross lingual VQG(visual question generation)
- English-only Object-Aware (OA) VQA
- Object Detection

Model	Components	Image Encoder	Mutimodal Encoder-Decoder	Total
PaLI-3B	ViT-G, mT5-L	1.8B	1.2B	3B
PaLI-15B	ViT-G, mT5-XXL	1.8B	13B	14.8B
PaLI-17B	ViT-e, mT5-XXL	3.9B	13B	16.9B

Flamingo



Objective:

$p(y|x)=\prod_{l=1}^{L} p(y_l|y_{<l},x_{\leq l})$ y_l is the l[th] language token of input text; $y_{<l}$ is set of preceding token; $x_{\leq l}$ set of images/videos preceding token y_l in the interleaved sequence and p is parametrized by Flamingo model.

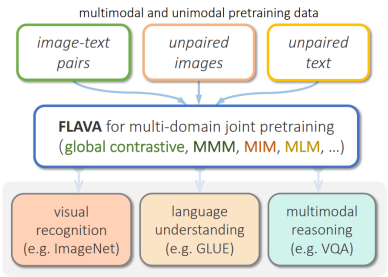
Dataset:

- Training on mixture of vision and language datasets
  MutltiModal MassiveWeb (M3W) : 43M webpages
- Pairs of image/video and text
  Long Text and Image Pair (LTIP) consist 312M image & text pair. Video and Text Pair (VTP) consist 27M short video paired with sentence description.

FLAVA

Foundational Language and Vision Alignment Model

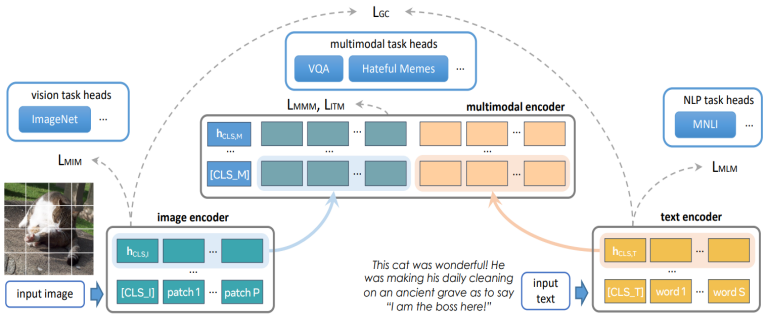
The model involves an image encoder to extract unimodal image representations, a text encoder to obtain unimodal text representations, and a multimodal encoder to fuse and align the image and text representations for multimodal reasoning.



Loss functions:

1) Masked Image Modeling (MIM) Loss: This loss function is used for pretraining the vision model by predicting masked regions in an image.

2) Masked Language Modeling (MLM) Loss: This loss function is used for pretraining the language model by predicting masked words in a sentence.



3) Multimodal Masked Modeling (MMM) Loss: This loss function combines both MLM and MIM to ensure that the model learns to understand and generate both text and images together.

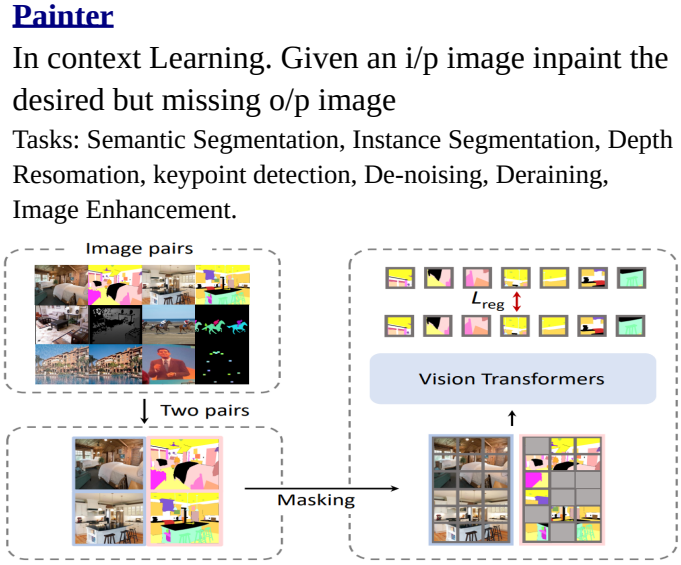
4) Image-Text Matching (ITM) Loss: This loss function is used to ensure that the model can correctly match corresponding pairs of images and texts.

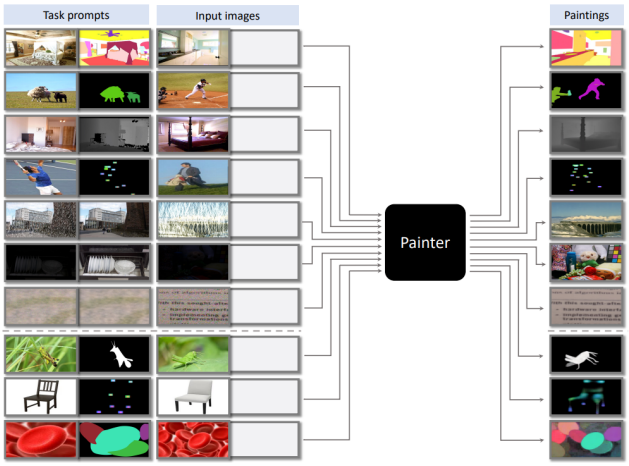
5)Contrastive Loss: This loss function helps in aligning the representations of different modalities (e.g., text and image) by bringing similar pairs closer and pushing dissimilar pairs apart2.

Painter

In context Learning. Given an i/p image inpaint the desired but missing o/p image

Tasks: Semantic Segmentation, Instance Segmentation, Depth Resomation, keypoint detection, De-noising, Deraining, Image Enhancement.

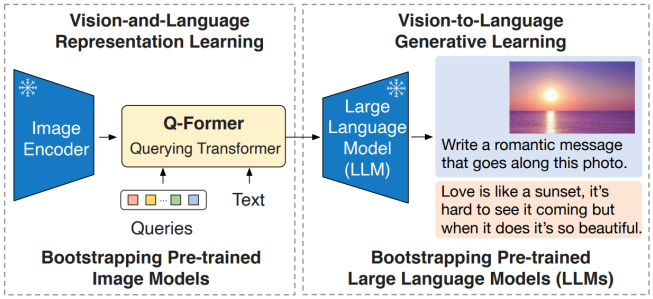




BLIP-2 Bootstrapping Language-Image Pre-training with Frozen Image Encoders and Large Language Models

Used lightweight Querying transformer to serve as bottleneck between frozen image and text encoder.

Q-Former uses a set of learnable querying vectors and is pre-trained in 2 stages:

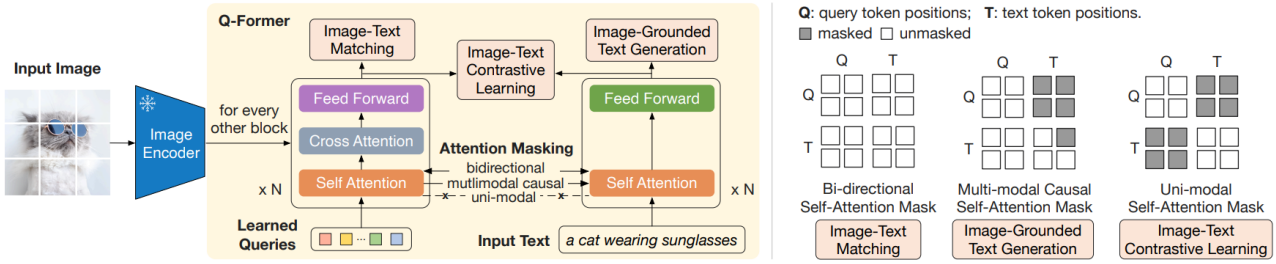


1) Vision-Language representation learning with froze image encoder.

2) Vision-Language generative learning with frozen text encoder.

<u>Vision-Language representation learning</u>

- Image-Text Contrastive Learning: Align image representation with text representation
- Image-Text Matching: Binary classification of image-text pair
- Image-grounded Text generation: condition Q-Former to generate text by accepting images as input



Vision-Language Generative Learning

Two variants of LLM architectures. Q-Former is connected to frozen LLM and pre-trained s.t. LLM generative ability is fully harnessed.

The output query embeddings from image transformer are linearly projected to same dimension as text embedding the language model expects. The projected query embedding are then concatenated to text embedding to serves as soft visual prompts forcing LM to focus on visual features extracted by Q-Former.

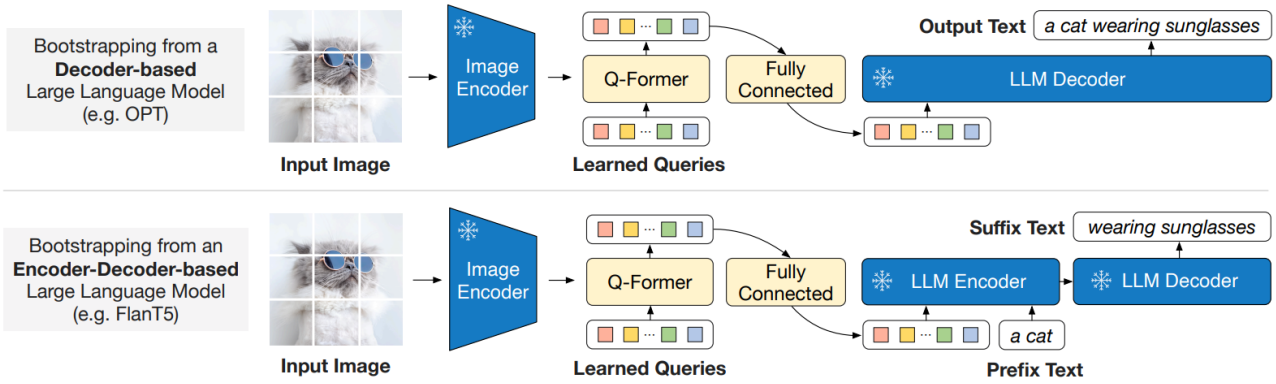
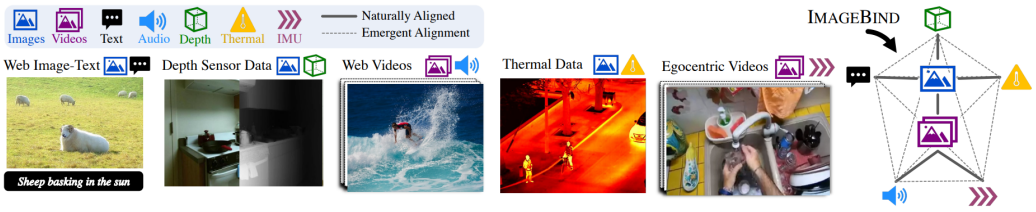


Image-Bind

Learn a joint embedding across six different modalities:

- Image
- Text
- Audio
- Depth
- Thermal
- IMU



Naturally Paired Datasets

- Video, audio pairs from Audioset dataset
- Image, depth pairs from SUN RGB-D datatset
- Image, thermal pairs from the LLVIP dataset
- Video IMU pairs from Ego4D datasets

ViT-H for Images/Videos

OpenCLIP for Text

ViT-B for Audio

ViT-S for Thermal

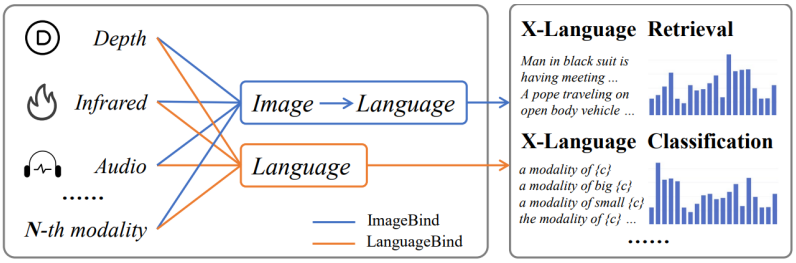
Images and Text Encoder are frozen during training; Audio, depth, thermal, IMU encoder are updated

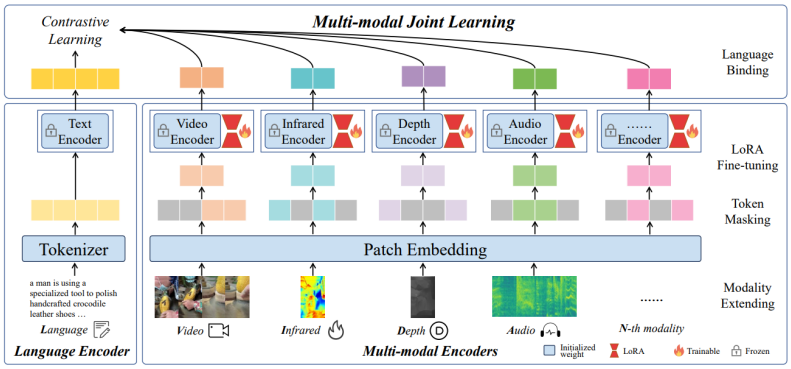
InfoNCE loss:

$$L_{I,M} = -\log \frac{\exp(q_i^T k_i)/\tau}{\exp(q_i^T k_i)/\tau + \sum_{j \neq i} \exp(q_i^T k_j)/\tau}$$

q_i: Normalized Embedding for Image, k_i: Normalized embedding for other modality

Language-Bind





1) 24 layer, 1024-dim ViT with patch size of 14
2) Initialize with OpenCLIP-large
3) Depth and infrared are treated as RGB image
4) Audio data is transformed to spectrogram of duration 10sec
3) Low Rank Adaptation (LoRA) for FineTunig, maintain weight W_0 frozen while learning new weight matrix BA.

In case of modality-agnostic encoder h(.) and x, forward process can be represented as:

$h(x)=W_0 x+BAx$

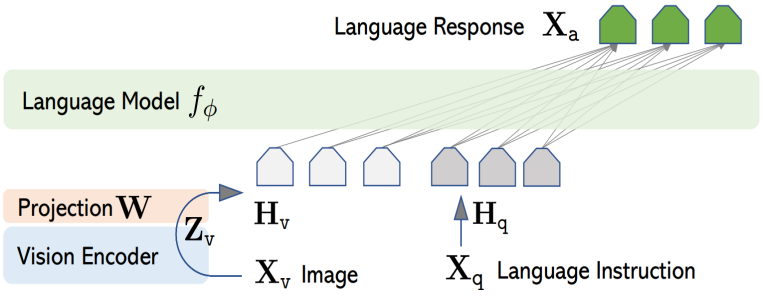
Contrastive learning to bind individual modalities to language

$$L_{M2T}=-\frac{1}{K}\sum_{i=1}^{K}\log\frac{\exp(x_i^T y_i/\tau)}{\sum_{j=1}^{K}\exp(x_i^T y_j/\tau)}$$

$$L_{T2M}=-\frac{1}{K}\sum_{i=1}^{K}\log\frac{\exp(y_i^T x_i/\tau)}{\sum_{j=1}^{K}\exp(y_i^T x_j/\tau)}$$

LlaVa Visual Instruction Tuning

$H=W.Z_v$ with $Z_v=g(X_v)$



Stage 1: Pretaining for feature Aligment

Filter CC3M to 595K image-text pairs, keep both the visual encoder and LLM weight frozen, train for W (projection matrix)

Stage 2: Fine-tuning End-to-End

Train $\theta=\{W,\phi\}$

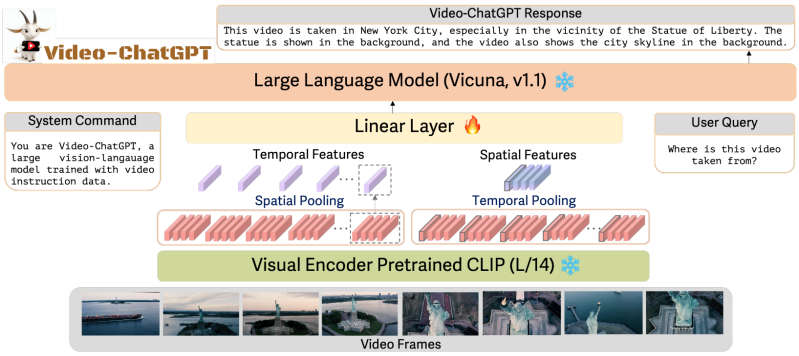
Instruction Tuning For each image X_v, generate multi-turn conversation data $(X_q^1, X_a^1, \ldots, X_q^T, X_a^T)$. Organise them as sequence, by treating all answers as the assistant's response. Instruction at the t[th] turn:

$$X_{instruct}^t=\begin{cases}\text{Randomly choose } [X_q^1, X_v^1] \text{ or } [X_v, X_q^1] & \text{the first turn } t=1\\ X_q^t, & \text{the remaining turn } t>1\end{cases}$$

Perform instruction tuning of LLM on the prediction tokens, using original auto-regressive training objective.

$$p(X_a|X_v, X_{instruct})=\prod_{i=1}^{L} p_\theta(x_i|X_v, X_{instruction,<i} X_{a,<i})$$

Video ChatGPT



Video Sample $V_i \in \Re^{T \times H \times W \times C}$, using CLIP generate $x_i \in \Re^{T \times h \times w \times D}$.

No. of tokens $N = h \times w$

Temporal representation $t_i \in \Re^{T \times D}$

Spatial representation $z_i \in \Re^{N \times D}$

Both concatenated to form video-level feature

$v_i = [t_i \quad z_i] \in \Re^{(T+N) \times D}$

Trainable layer g, projects these video-level features into language decoder's embedding space

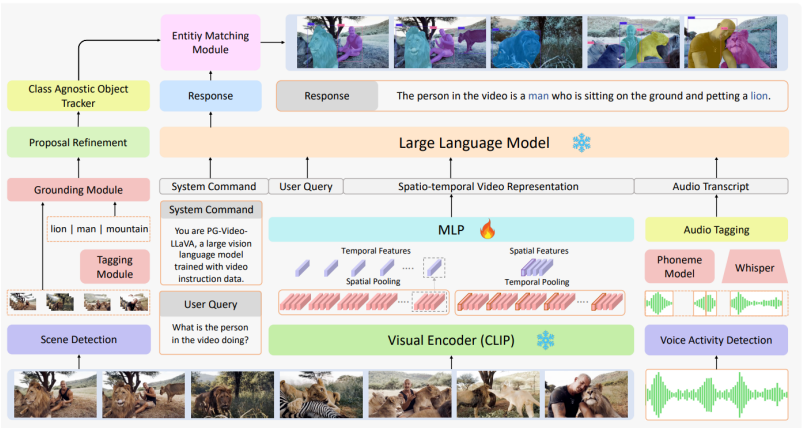
$Q_v = g(v_i) \in \Re^{(T+N) \times K}$

Text Queries $Q_t \in \Re^{L \times K}$ is concatenated with Q_t

Video Instruction Set creation

- Human Assisted Annotations: Videos and captions from ActivityNet-200. Annotators enrich original captions by adding information about: Physical appearance and Spatial and temporal localization
- Semi-Automatic Annotation Framework: Use BLIP-2 and GriT for key frame analysis. Also use Tag2Text to generate tags fr each key-frame.
- GPT-Assisted PostProcessing

PG-Video-LlaVa



Incorporates audio context that enhances video understanding.

1)Frozen LLM: Vicuna-13b-v1.5

2)Voice activity Detection: Identifies speech segments, filter out noise, cuts and merges audio into batches.

3)Whisper models converts speech to transcript. Audio Tagging produce audio-tagging o/p from sliced original audio, consider top 3 audio classes predicted. If speed is not among them segment is ignored. Phoneme classifies the smallest unit of speech, temporary aligns transcript with matching phonemes.

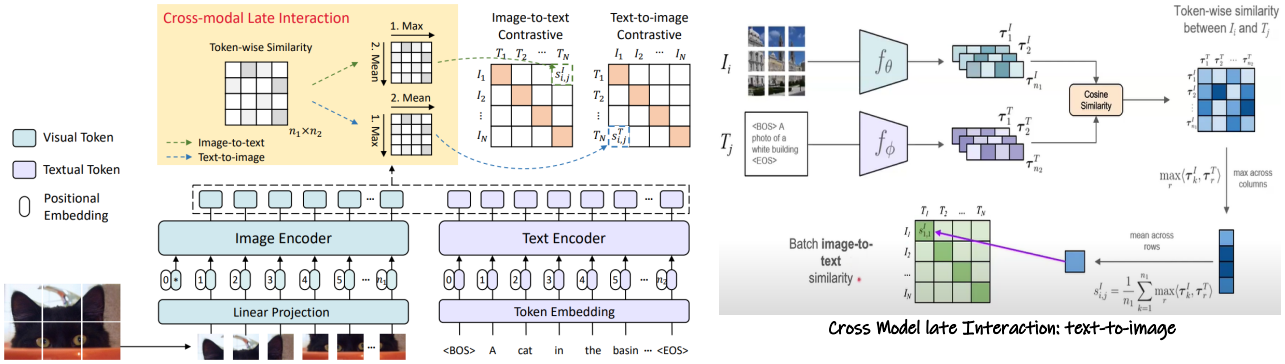
4)Spatial Temporal Video Representation same as Video-Chat GPT

5)Scene Detection (PyScene Detect): to group frames from same scene; Tagging Module (RAM SWIN) generate objects in that image; Grounding Module (GroundingDINO-T) Produces Bounding boxes in the image; Proposal Refinement (SAM): o/p segmentation; Class Agnostic Object Tracker (DEVA Tracker) decouples segmentation and tracking tasks to be class agnostics

6) Entity Matching Module (Vicuna 1.5) takes LLM response and list of objects to filetr out objects

FILIP **F**ine-Grained **I**nteractive **L**anguage-**I**mage **P**re-Training

CLIP is not able to captures fine-grained interactions, also cannot capture relationship between image patches and textual words.

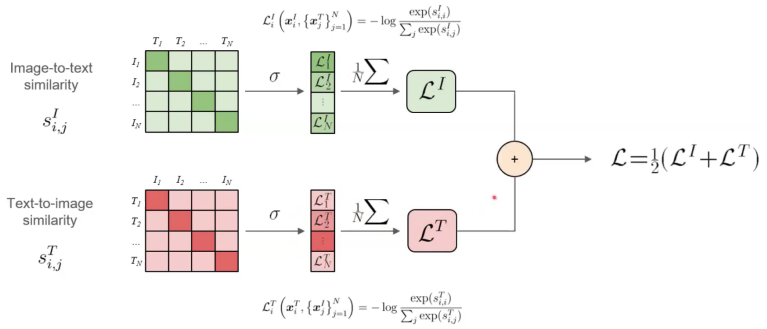


Cross Model late Interaction: text-to-image

n_1 and n_2 be the no. of non-padded tokens in ith image and jth text, corresponding encoded feature τ_1^I and τ_j^T. Compute the similarity matrix.

For image-to-text similarity, take max along columns and mean it to get s_{ij}^I.

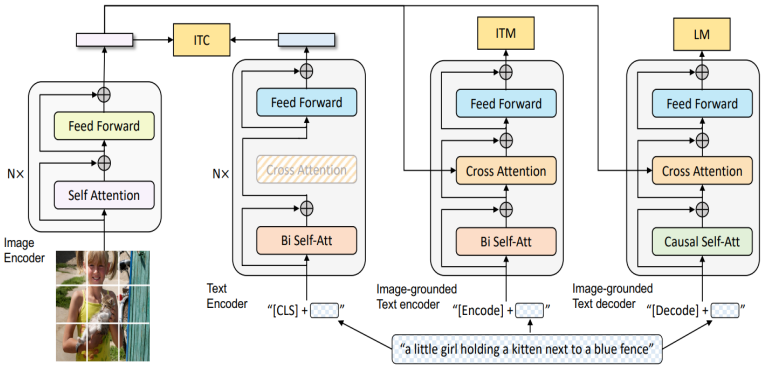
Similarly for text-to-image similarity, take max along row and mean it to get s_{ij}^T



Contrastive Loss

BLIP

Pre-training model architecture and objectives of BLIP (same parameters have the same colour).



1) Unimodal encoder (ViT-B/16 and ViT-L/16; BERT base) is trained with an image-text contrastive (ITC) loss to align the vision and language representations.

$$L_{ITC}=\frac{1}{2}\underset{(I,T)\sim D}{E}\left[H\left(y^{i2t}(I)\right)+H\left(y^{t2i}(T)\right)\right]$$

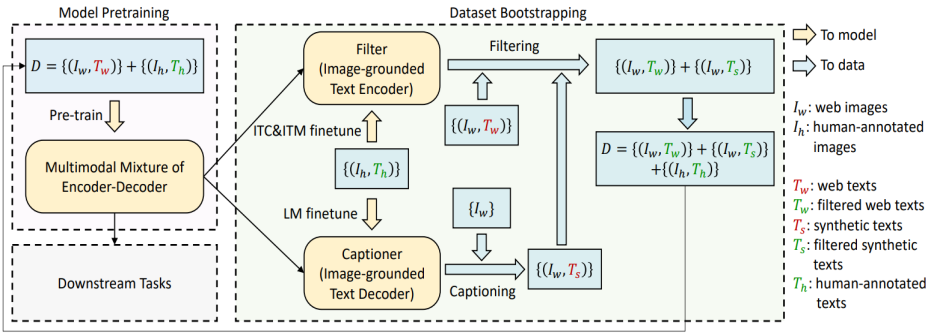
2) Image-grounded text encoder uses additional cross-attention layers to model vision-language interactions, and is trained with a image-text matching (ITM) loss to distinguish between positive and negative image-text pairs.

$$L_{ITM}=\underset{(I,T)\sim D}{E}H\left(y^{itm},p^{itm}(I,T)\right)$$

3) Image-grounded text decoder replaces the bi-directional self-attention layers with causal self-attention layers, and shares the same cross-attention layers and feed forward networks as the encoder. The decoder is trained with a language modeling (LM) loss to generate captions given images.

$$H(t,p)=-\sum_{s\in S}t(s).\log\left(p(s)\right)$$

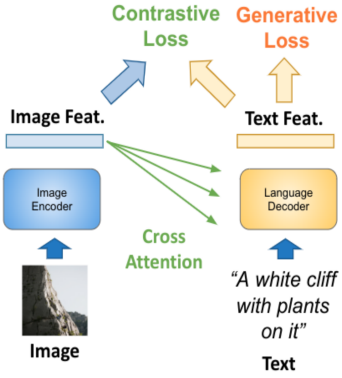
The goal of CapFilt is to take noisy web data, filter out unusable captions and provide higher quality synthetic captions,



Captioner to produce synthetic captions for web images, and a filter to remove noisy image-text pairs. The captioner and filter are initialized from the same pre-trained model and finetuned individually on a small-scale human-annotated dataset. The bootstrapped dataset is used to pre-train a new model.

MaMMUT

A Simple Architecture for Joint Learning for MultiModal Tasks



ViT-based Vision Encoder and a single text decoder

- M cross Attention layer into N text decoder layers
- No tasks-specific head required

Uses a two pass learning strategy approach to unify

- contrastive learning
- Autoregressive captioning
- Localization Awareness (with cropped positional embedding)

Allows maximal weight sharing for generative and contrastive tasks

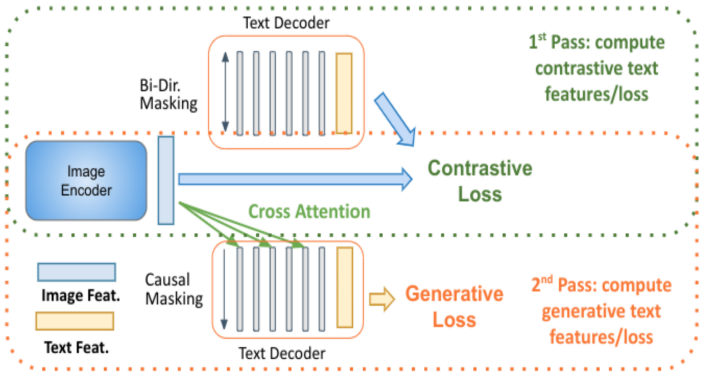
1st pass :: Focal Contrastive Loss: Contrastive loss require larger batch size. Goal is to learn from more challenging and informative examples

v_i and l_j represent normalized image and text embeddings. τ is temperature

$$p_i = \begin{cases} \sigma(v_i l_j / \tau) & \text{if } i = j \\ 1 - \sigma(v_i l_j / \tau) & \text{if } i = j \end{cases}$$

$$L_{\text{focal\_constrastive}} = -\frac{1}{B}\sum_{i=1}^{B}\sum_{j=1}^{B}(1-p_i)^{\gamma}\log(p_i)$$

$$L_{constarstive} = L_{I2T} + L_{T2I}$$



2nd pass :: Generative Loss

$$L_{captioning} = -\sum_{t=1}^{T}\log P_{\theta}(y_t | y_{1,2,\ldots,t-1}, x)$$

$$L_{total} = \lambda_{cap} L_{captioning} + \lambda_{focal} L_{\text{focal\_contrastive}}$$

MERLOT RESERVE

Multimodal Event Representation Learning Over Time, with RE-entrant SupERVision of Events

Neural Script Knowledge through Vision and Language and Sound

Script Knowledge is a body of knowledge that describes a typical sequence of actions people do in a particular situation.

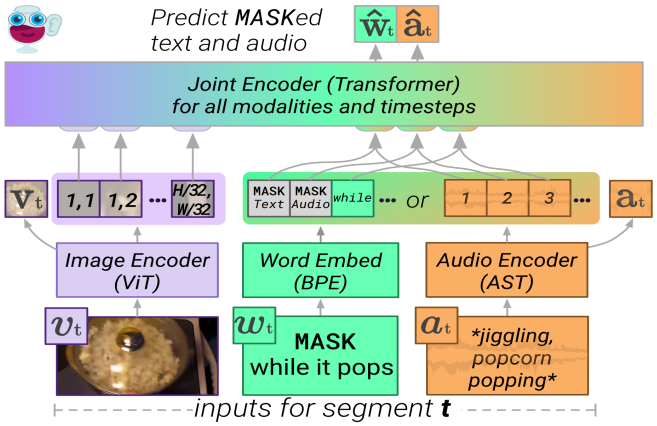
Model learns from Video frames, Subtitles, Audio. Given a video, replace subtitles and audio with MASK token. The model predicts by choosing correct masked-out snippet.

Divide the video into segments of 5 sec. For each segment s_t

- Acquire middle frame v_t
- Acquire ASR token w_t
- Acquire audio a_t

Image Encoder: Use ViT to encode each frame independently, used patch size of 16 and apply 2x2 query key value attention pool after transformer, converting image of HxW into H/32xW/32 feature map of dimension d_h

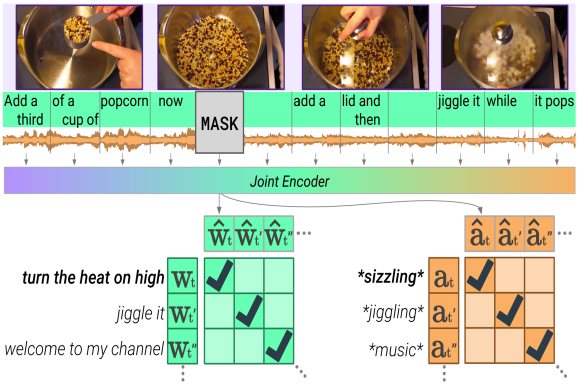
Audio Encoder: Split audio in each segment at into 3 equal-sized subsegment, use Audio Spectrogram Transformer to encode each subsegment. The three feature maps are concatenated, the result is of 18xd_h for every 5 second audio.



Joint Encoder: Jointly encode all modalities using bidirectional Transformer. We use a linear projection of final layer's hidden states for all objective.

Avoiding shortcut Learning Train two types of masked videos

- Audio only as target: Provide video frames and text, infer text and audio representation in MASKed tokens
- Audio as input : Provide video frames and (text or audio), infer only text representations in MASKed tokens



Contrastive Masked Span

$$L_{mask \to text} = \frac{1}{|W|} \sum_{w_t \in W} \left(\log \frac{\exp(\sigma \hat{w}_t . w_t)}{\sum_{w \in W} \exp(\sigma \hat{w}_t . w)} \right)$$

$L_{text \to mask}$ is the transpose if $L_{mask \to text}$

$L_{text} = L_{mask \to text} + L_{text \to mask}$

Similarly, we get the audio loss L_{audio}.

$L_{constrastive\ masked\ span} = L_{text} + L_{audio}$

Contrastive Frame-Transcript Matching

$L_{constrastive\ frame-transcipt\ matching} = L_{frame}$

$L_{frame} = CE_{transcript \to frame} + CE_{frame \to transcript}$

Total Loss

$L_{constrastive\ span\ training} = L_{text} + L_{audio} + L_{frame}$

Shikra

Unleashing Multimodal LLM's Referential Dialogue Magic