Unified Multimodal Understanding and Generation

Xu Tan
https://tan-xu.github.io/

Benefits of Unified Model

- Unification
  - Support all tasks in one model, save training/deployment/team cost
- Synergy
  - What I cannot create, I do not understand
  - Share knowledge/capacity, boost understanding and generation
- Context
  - Multi-turn session-based interaction
  - Tasks requiring both understanding and generation
- Future
  - Next generation AI solutions, world models, and embodied AI

Outline

- Part 1: Taxonomy + Overview
- Part 2: Research Topics

Part 1: Taxonomy + Overview

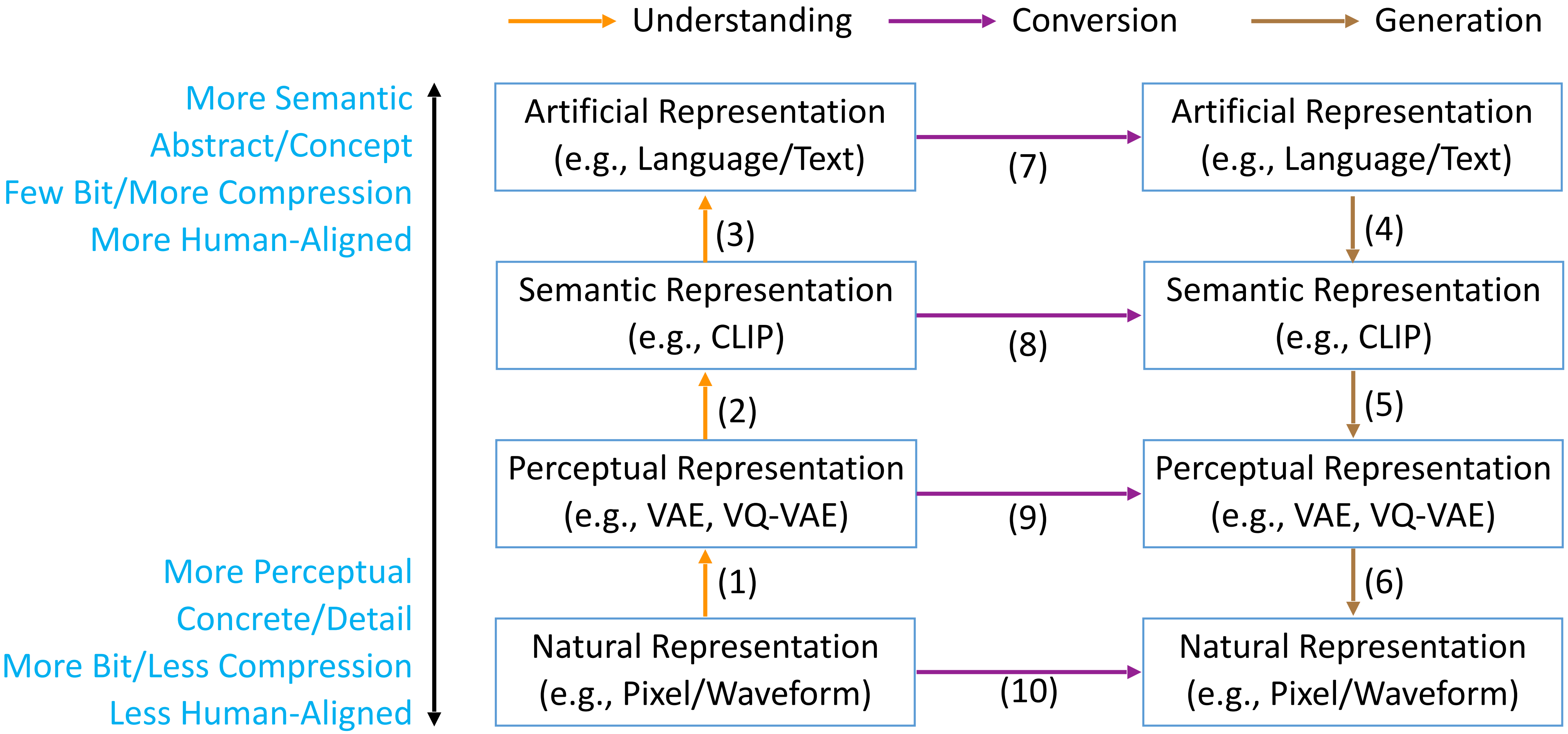
Tokenizer -> Unified Model -> Detokenizer

Representation

+

Modeling

Part 1.1: Representation



(1): VAE encoder
(1,2): CLIP image encoder
(1,2,3): Captioning/OCR
(1,2,3,7): VLM

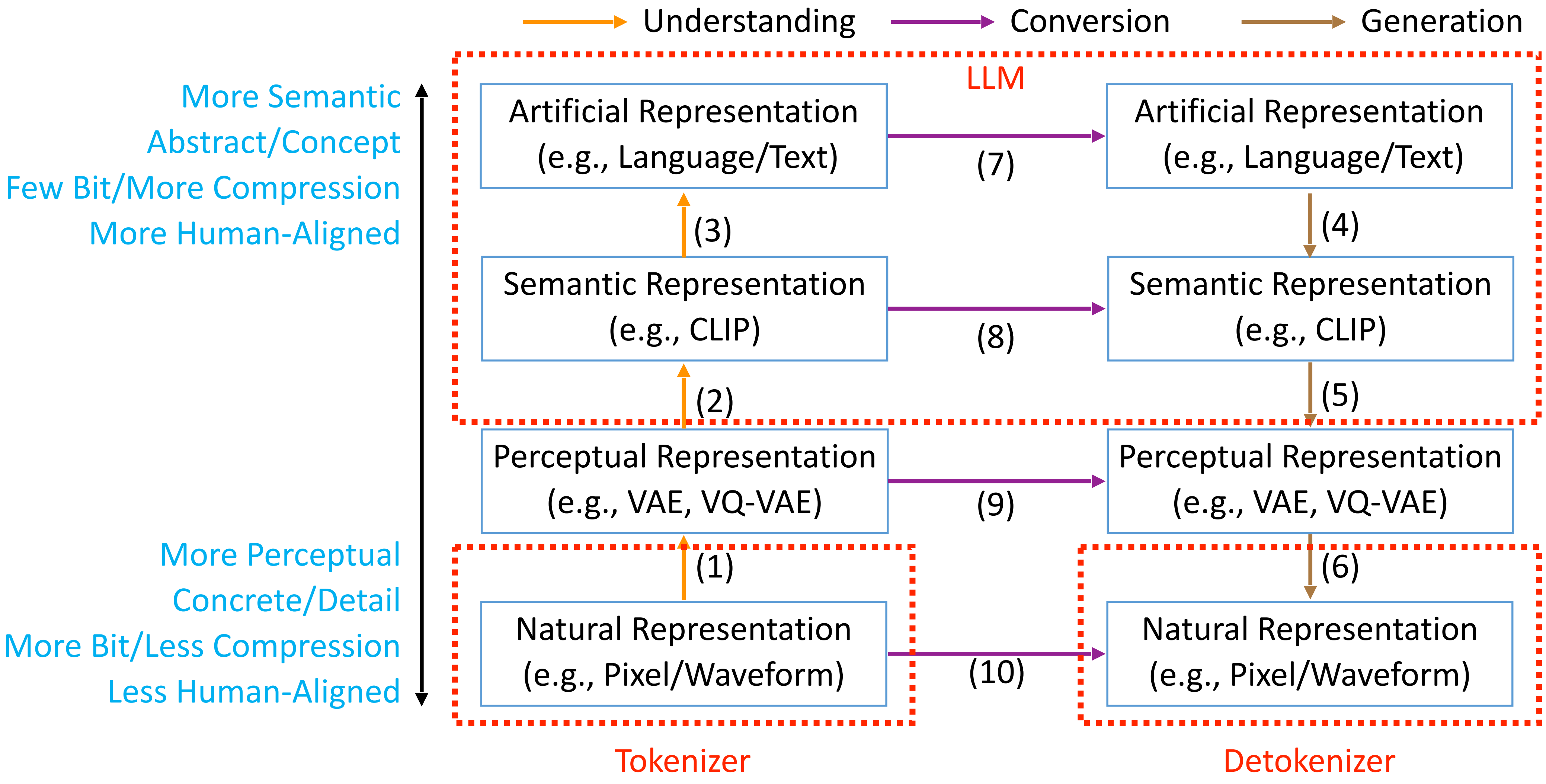
(4): CLIP text encoder
(4,5): Stable Diffusion/DiT
(6): VAE decoder
(4,5,6): ImageGen/VideoGen

(7): LLM
(2,3,7,4,5): Emu3
(3,7,4): BLIP3-o/MetaMorph
(1,2,3,7,4,5,6): GPT-4o service

Representation: Semantic vs Perceptual

- Representation determines the boundary between Tokenizer, Unified Model, and Detokenizer
  - Unified Model: should focus more on semantic information, align with text
  - Tokenizer/Detokenizer: focus more on perceiving and rendering details, compression/decompression

Representation: Perceptual In/Out

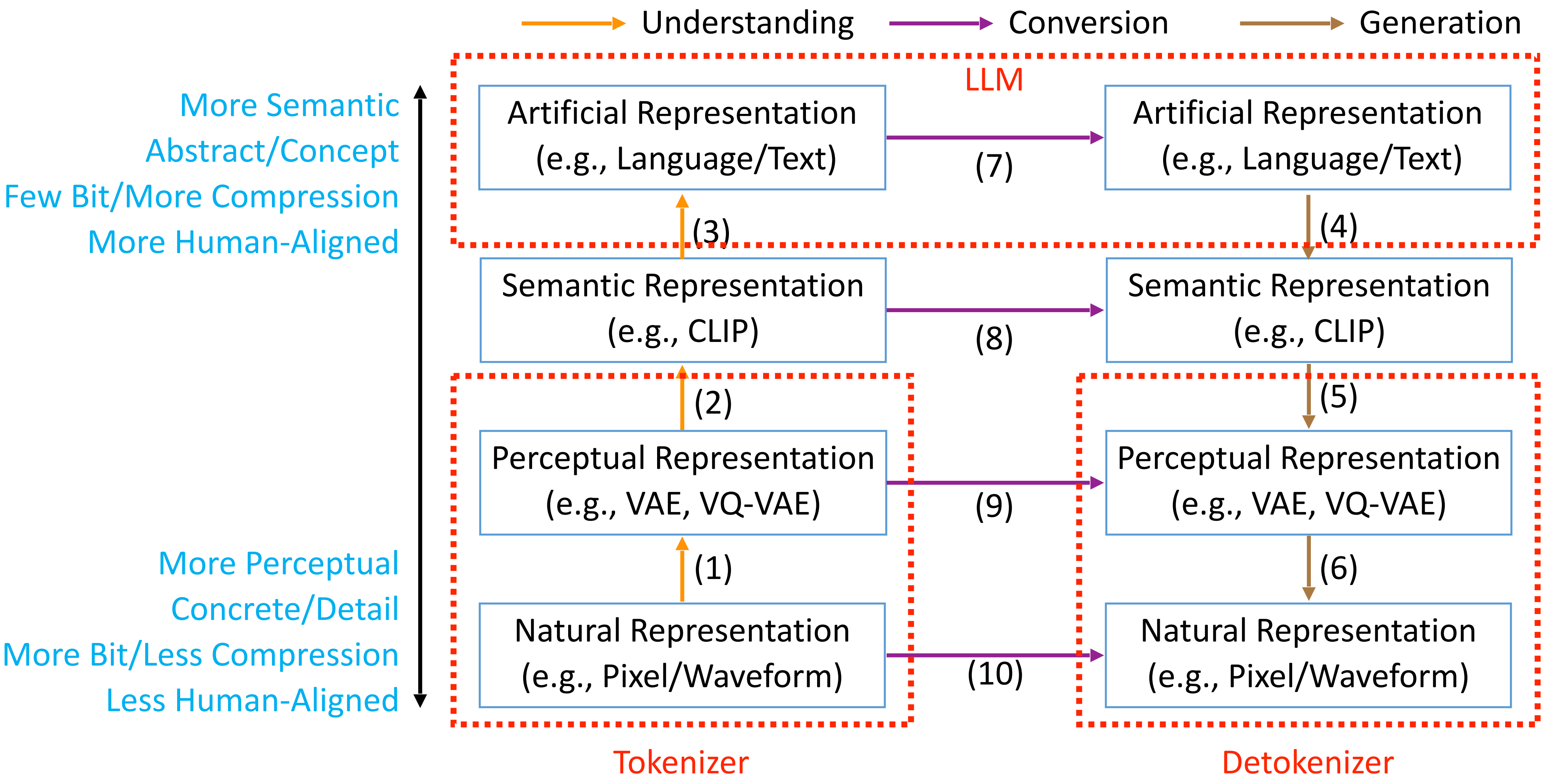


Representation: Perceptual In/Out

- Tokenizer and Detokenizer use VAE/VQ-VAE
- Unified Model handles more complicated task, larger gap to text
  - VAE cannot learn human-aligned semantic feature, not suitable for semantic task
- Easier for generation, but harder for understanding
  - VAE learns details with pixel reconstruction, good for generation

- Related Work
  - Chameleon (arXiv:2405.09818)
  - Transfusion (arXiv:2408.11039)
  - Show-o (arXiv:2408.12528)
  - Emu3 (arXiv:2409.18869)
  - LatentLM (arXiv:2412.08635)

Representation: Semantic In/Out



Representation: Semantic In/Out

- Tokenizer uses CLIP series, Detokenizer uses extra Diffusion Model
- Unified Model handles less complicated task
  - CLIP learns to align with text, with high-level semantic information, smaller gap to text
- Easier for understanding, but harder for generation
  - CLIP lacks details for fine-grained reconstruction/generation

- Related Work
  - Emu/Emu2 (arXiv:2307.05222/arXiv:2312.13286)
  - MetaMorph (arXiv:2412.14164)
  - BLIP3-o (arXiv:2505.09568)
  - LanDiff (arXiv:2503.04606)

Representation: Semantic In / Perceptual Out

- Tokenizer uses CLIP series, Detokenizer uses VAE/VQ-VAE
- Suitable for both understanding and generation
- But mismatch input/output representation, different spaces, harder for LLM

- Related Work:
  - Janus (arXiv:2410.13848)
  - Janus-Pro (arXiv:2501.17811)
  - UniFluid (arXiv:2503.13436)
  - TokLIP (arXiv:2505.05422)

Representation: Semantic + Perceptual In

- Tokenizer uses CLIP + VAE, Detokenizer uses VAE/VQ-VAE or CLIP+VAE/VQ-VAE
- Suitable for both understanding and generation
- Friendly for conversion/editing

- Related Work:
  - Mogao (arXiv:2505.05472)
  - BAGEL (arXiv:2505.14683)
  - ILLUME+ (arXiv:2504.01934)
  - QLIP (arXiv:2502.05178), UniTok (arXiv:2502.20321), UniToken (arXiv:2504.04423)

Representation: Continuous vs Discrete

- Discrete In/Out: align with LLM/Next Token Prediction
  - e.g., Emu3/Chameleon
- Continuous In/Out: better preserve information
  - e.g., Emu/Emu2/MetaMorph/BLIP3-o/BAGEL
- Continuous In/Discrete Out: align with VLM
  - e.g., Janus/Janus-Pro

Part 1: Taxonomy + Overview

Tokenizer -> Unified Model -> Detokenizer

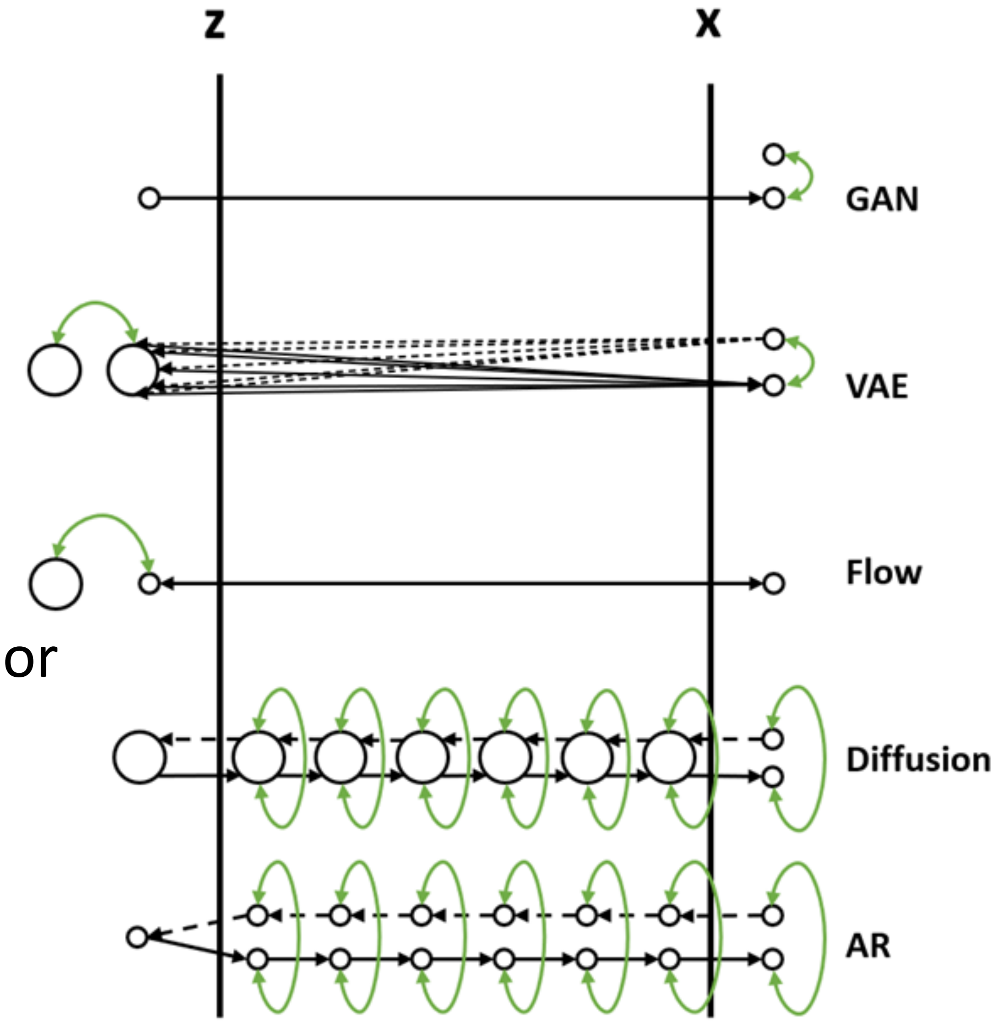
Representation

+

Modeling

Part 1.2: Modeling

- Why AR (LLM) and Diffusion (DiT)?
    - Data factorization, chain-of-thought
    - Compute upscaling
- Which is better?
    - Diffusion
        - Non-causal, iterative generation, no order prior
    - AR
        - Causal, next token prediction, order prior
        - Reduce solutions exponentially!
        - KV cache, compute downscaling



https://zhuanlan.zhihu.com/p/591881660

Modeling: AR + Discrete Tokens

- Related Work: Chameleon (arXiv:2405.09818), Emu3 (arXiv:2409.18869)
- Problems
  - More Compression, Few Bit, Small Entropy
  - Limited Perceptual/Semantic Information
- Solutions
  - Smaller patches, longer sequence, more tokens
  - Multiple tokens for a single patch

Modeling: AR + Continuous Tokens

- Regression loss
  - L1/L2 loss (Emu/Emu2, arXiv:2307.05222/arXiv:2312.13286)
  - Cosine loss (MetaMorph, arXiv:2412.14164)
- Issue of continuous AR (differ from discrete)
  - Error propagation (Nexus-Gen, arXiv:2504.21356)

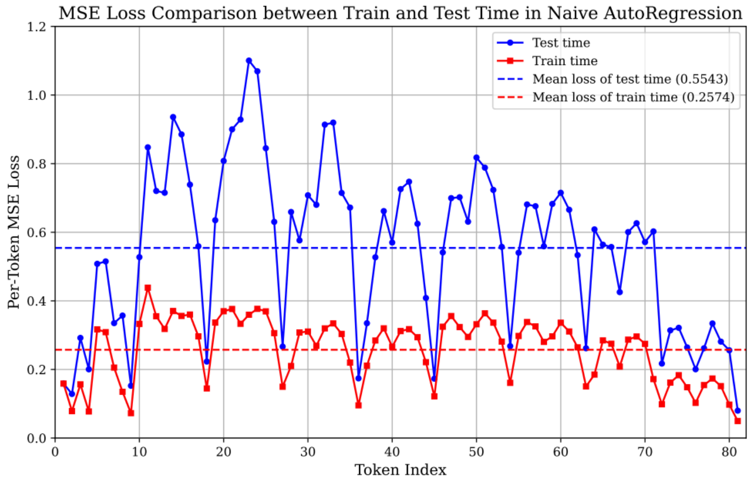


Figure 4: Mean squared error comparison between train and test time in the naive autoregression paradigm.

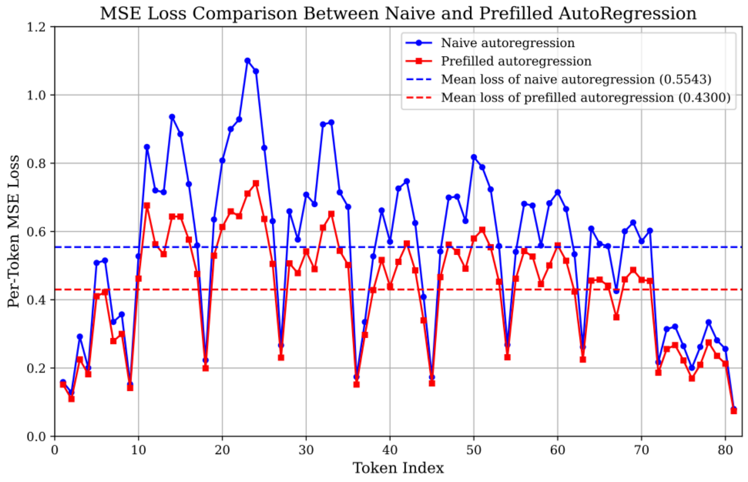
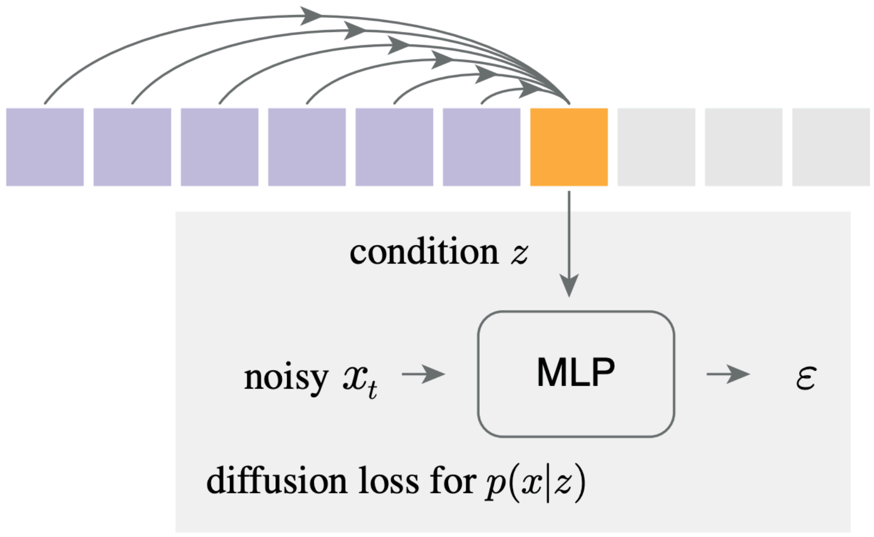


Figure 5: Mean squared error comparison between naive autoregression and prefilled autoregression during inference.

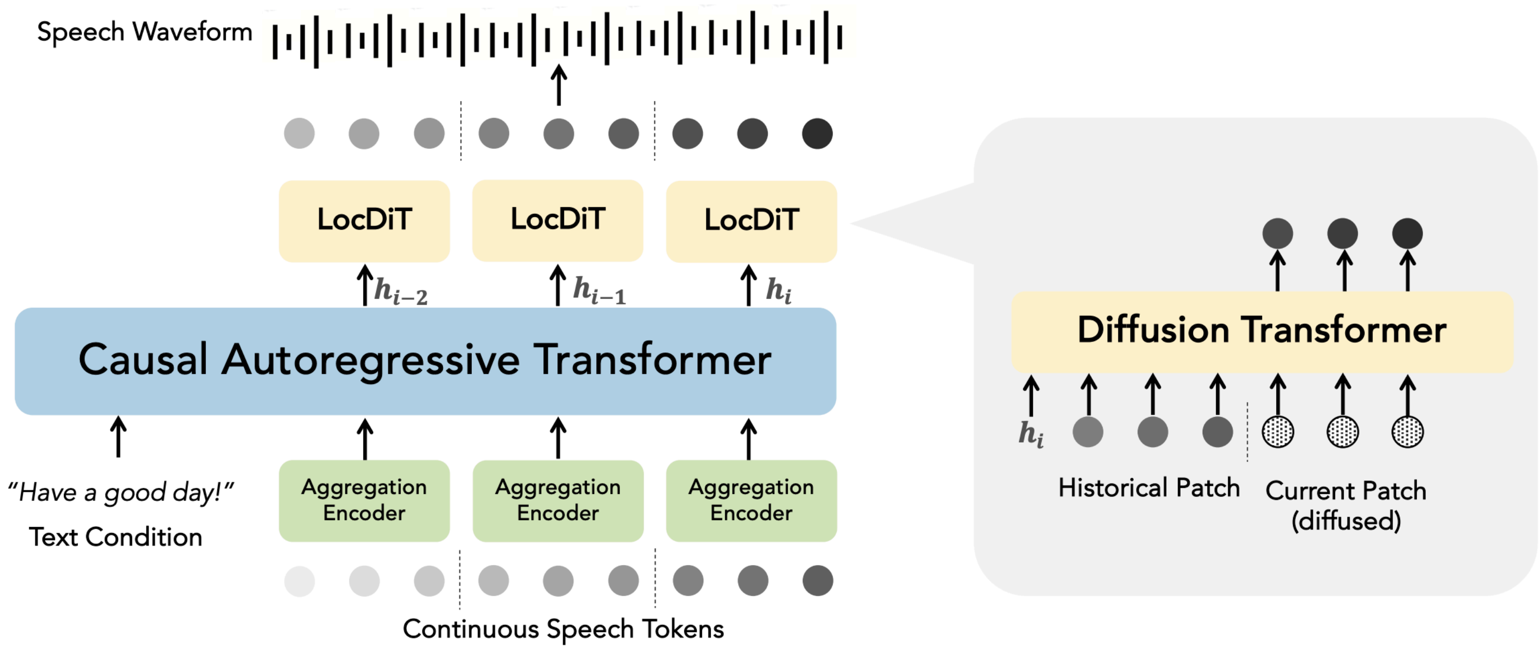
Modeling: AR + Continuous Tokens

- Diffusion head (v1): Per-token diffusion loss
  - Model capacity for generation: mainly in LLM, only MLP in diffusion
  - e.g., MAR (arXiv:2406.11838), UniFluid (arXiv:2503.13436)



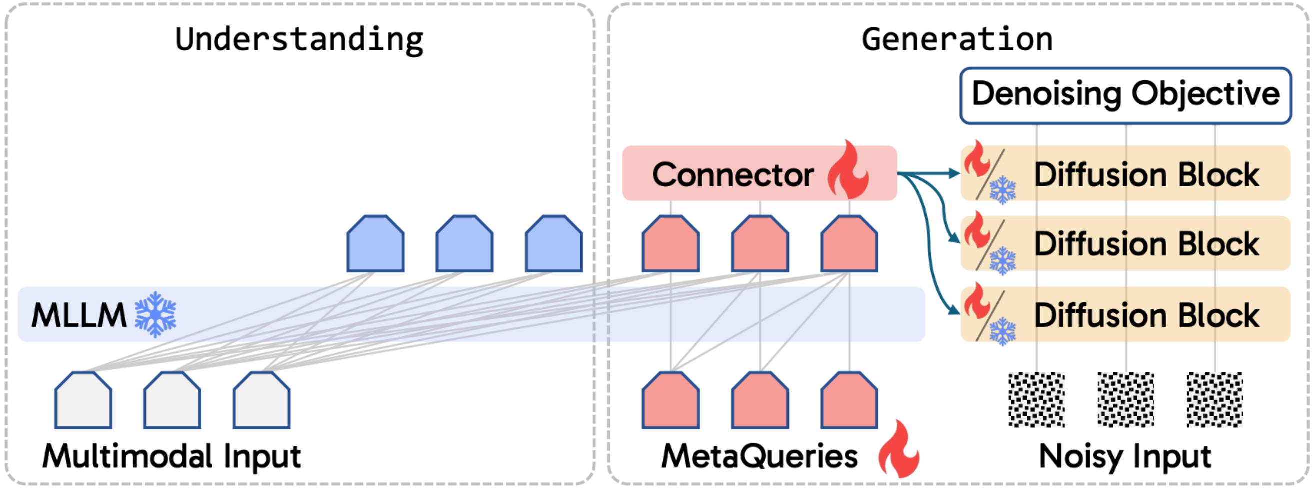
Modeling: AR + Continuous Tokens

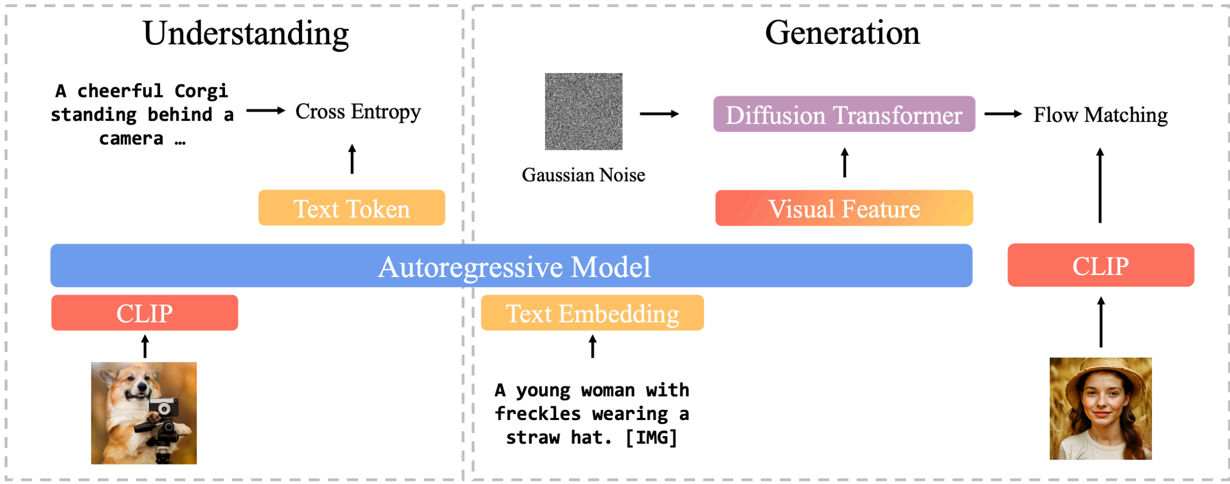
- Diffusion head (v2): Semi-autoregressive + Diffusion Transformer
  - Multiple patches/tokens in an autoregressive step, e.g., DiTAR (arXiv:2502.03930)
  - Model capacity for generation: more in LLM, less in diffusion



Modeling: AR + Continuous Tokens

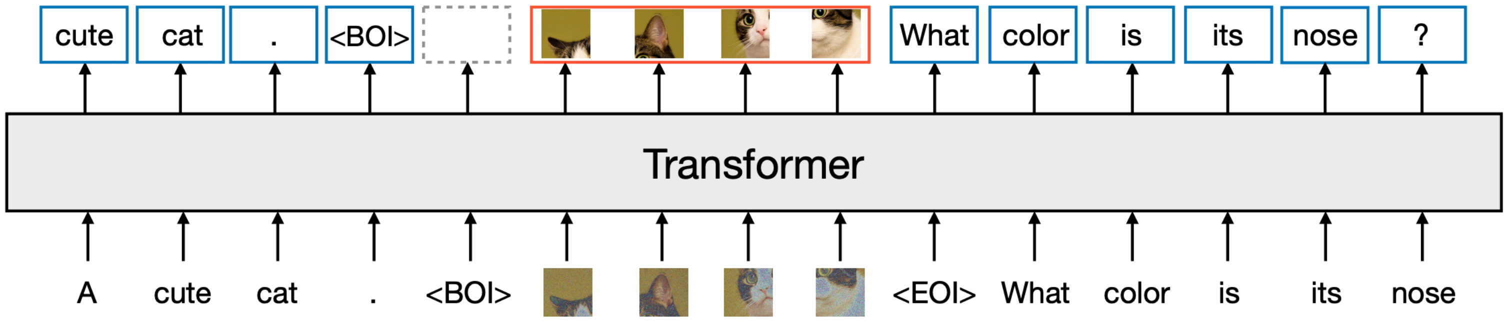
- Diffusion head (v3): Non-autoregressive + Diffusion Transformer
  - All patches/tokens in an autoregressive step
  - Model capacity for generation: less in LLM, more in diffusion
  - e.g., MetaQuery (arXiv:2504.06256), BLIP3-o (arXiv:2505.09568)

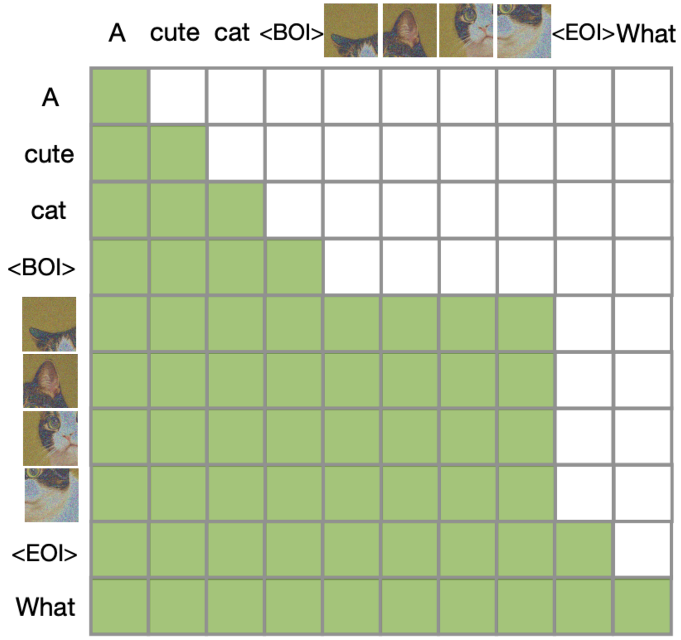




Modeling: AR + Continuous Tokens

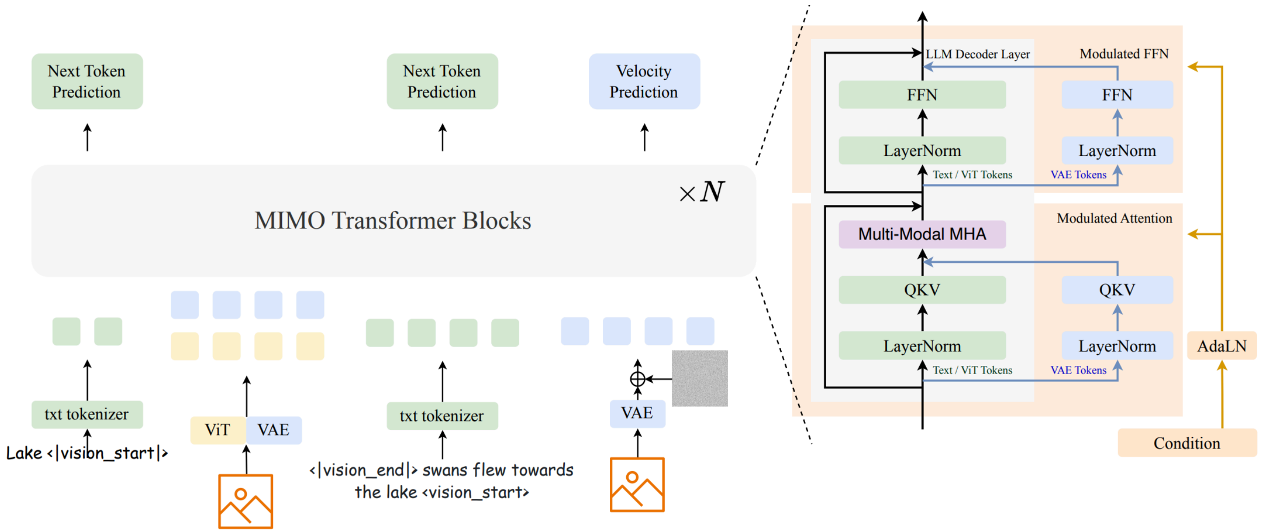
- Diffusion head (v4): In-place non-autoregressive/diffusion (shared)
  - LLM and diffusion share the same parameters
  - Model capacity for generation: all in diffusion, the same as LLM
  - e.g., Transfusion (arXiv:2408.11039), JanusFlow (arXiv:2411.07975)

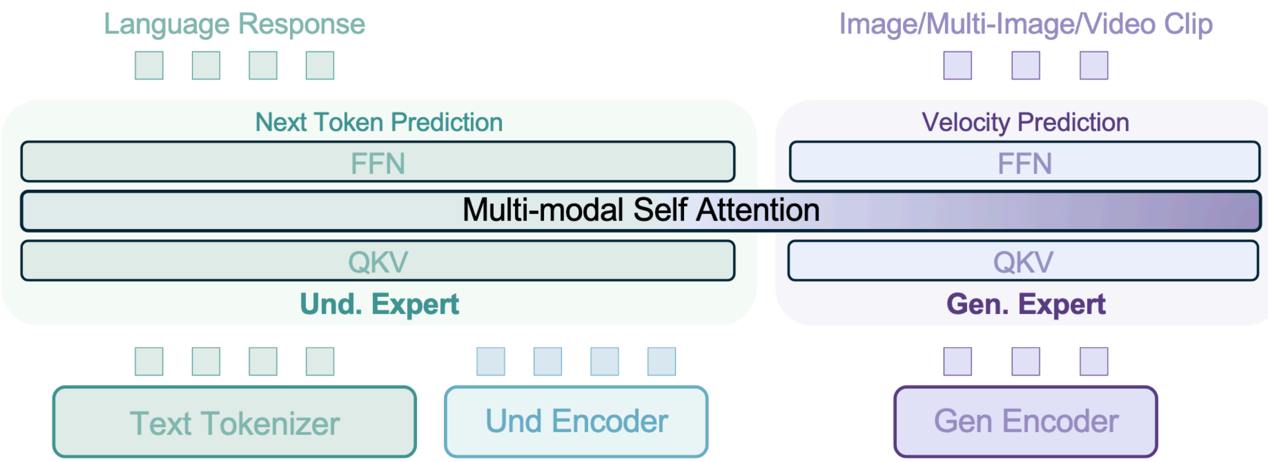




Modeling: AR + Continuous Tokens

- Diffusion head (v5): In-place non-autoregressive/diffusion (non-shared)
  - LLM and diffusion use different parameters: Mixture-of-Transformers (MoT)
  - Model capacity for generation: all in diffusion, the same as LLM
  - e.g., Mogao (arXiv:2505.05472), BAGEL (arXiv:2505.14683)





Modeling: Diffusion

- Text Diffusion
  - Diffusion-LM (arXiv:2205.14217), DiffuSeq (arXiv:2210.08933), DiffusionBERT (arXiv:2211.15029), Differmer (2212.09412)
  - LLaDA (arXiv:2502.09992)
  - Mercury, Gemini Diffusion
- Multimodal Diffusion
  - LLaDA-V (arXiv:2505.16933)
  - MMaDA (arXiv:2505.15809): unified understanding and generation

Outline

- Part 1: Taxonomy + Review
- Part 2: Research Topics

Ideal Paradigm for Unified Understanding/Generation

- For Unification/Synergy/Context, the paradigm should satisfy
  - Requirement 1: Unify representation for multimodal input and output
    - Requirement 1.1: Semantic or Perceptual or Both
    - Requirement 1.2: Discrete or Continuous
  - Requirement 2: Unify modeling for multimodal understanding and generation
    - Requirement 2.1: AR, or Diffusion, or AR + Diffusion
    - Requirement 2.2: Share model parameters
- For good performance, the paradigm should satisfy
  - Requirement 3: Benefit both understanding and generation

Ideal Paradigm for Unified Understanding/Generation

- Why requirement 1?
  - Input and output representation should be in the same space, better for synergy and consistent context
- Why requirement 2?
  - Modeling task for understanding and generation should be the same (e.g., next token prediction),
    - The model only pursue one goal
    - X->Y and Y->X can be the same space under one goal
  - Parameter should be shared for synergy
    - If not sharing parameters, the unified model is similar to two models, no synergy
    - Then it is not unified model, but orchestrated models, like agent

Existing Work and Requirements

Work	Req. 1	Req. 1.1	Req. 1.2	Req. 2	Req. 2.1	Req. 2.2	Req. 3
Chameleon	✅	✅	✅	✅	✅	✅	❌
Emu3	✅	✅	✅	✅	✅	✅	❌
Transfusion	✅	✅	✅	❌	❌	✅	❌
Show-o	✅	✅	✅	❌	❌	✅	❌
LatentLM	✅	✅	✅	❌	❌	✅	❌
Emu/Emu2	✅	✅	✅	✅	✅	✅	❌
MetaMorph	✅	✅	✅	✅	✅	✅	❌
BLIP3-o	✅	✅	✅	❌	❌	❌	❌
Janus/Janus-Pro	❌	❌	❌	✅	✅	✅	?
UniFluid	❌	❌	✅	✅	✅	✅	?
Mogao	❌	❌	✅	❌	❌	❌	?
BAGEL	❌	❌	✅	❌	❌	❌	?
ILLUME+	❌	✅	❌	✅	✅	✅	?
Ideal Paradigm	✅	✅	✅	✅	✅	✅	✅

Topic 1.1——Representation: Semantic or Perceptual

Representation		Pros	Cons
Input	Output		
Perceptual	Perceptual	Good for Generation Tokenizer/Detokenizer Simple	Not Good for Understanding Large Gap to Text
Semantic	Semantic	Good for Understanding Small Gap to Text	Detokenizer Complicated Not Good for Conversion/Edit
Semantic	Perceptual	Good for Understanding and Generation Tokenizer/Detokenizer Simple	Input/Output Mismatch Not Good for Conversion/Edit
Semantic + Perceptual	Perceptual	Good for Understanding and Generation Good for Conversion/Edit	Input/Output Mismatch
Semantic + Perceptual	Semantic + Perceptual	Good for Understanding and Generation Good for Conversion/Edit	Tokenizer/Detokenizer Complicated

Topic 1.1——Representation: Semantic or Perceptual

- The dilemma of representation
  - For understanding, semantic feature is better
    - Align with human-centric understanding, better than perceptual feature (e.g., Janus)
    - However, for edit/conversion task, semantic feature lack details for fine-grained editing and content preservation, perceptual feature is also necessary (e.g., Mogao, BAGEL)

Topic 1.1——Representation: Semantic or Perceptual

- The dilemma of representation
  - For generation, perceptual feature is better
    - Reconstruction quality is better than semantic feature
      - For perceptual feature, directly leverage VAE/VQ-VAE decoder to generate images
      - For semantic feature, usually need additional diffusion for generation
    - However, perceptual feature lacks semantic details
      - The physics/motion in generated images/videos is not good
      - Usually supplement with additional semantic information in generation
        - e.g., VideoJAM (arXiv:2502.02492), REPA (arXiv:2410.06940)

Topic 1.1——Representation: Semantic or Perceptual

- Possible solutions to the dilemma of representation
  - Solution 1: Semanticize perceptual feature, or perceptualize semantic feature
    - Feature should have reconstruction ability, but most importantly with semantics
      - Prefer semantic over perceptual
      - Not necessarily keep every details for reconstruction
    - Align VAE latent with semantic representation
      - e.g., ReaLS (arXiv:2502.00359), REPA-E (arXiv:2504.10483), VA-VAE (arXiv:2501.01423)
    - Train tokenizer with both reconstruction and text alignment objectives
      - e.g., QLip (arXiv:2502.05178), UniTok (arXiv:2502.20321)
    - Semanticize perceptual tokens with semantic supervision
      - e.g., TokLIP (2505.05422)

Topic 1.1——Representation: Semantic or Perceptual

- Possible solutions to the dilemma of representation
  - Solution 2: Use both semantic and perceptual tokens
    - Concatenate channel-wise
      - e.g., MUSE-VL (arXiv:2411.17762)
    - Concatenate sequence-wise in interleaving pattern
      - e.g., ILLUME+ (arXiv:2504.01934)

  - Better solutions?

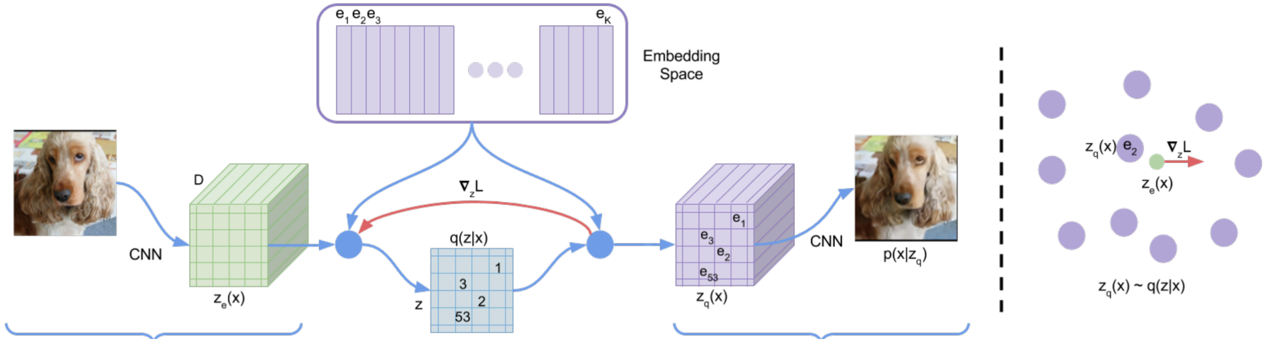
Topic 1.2——Representation: Continuous or Discrete

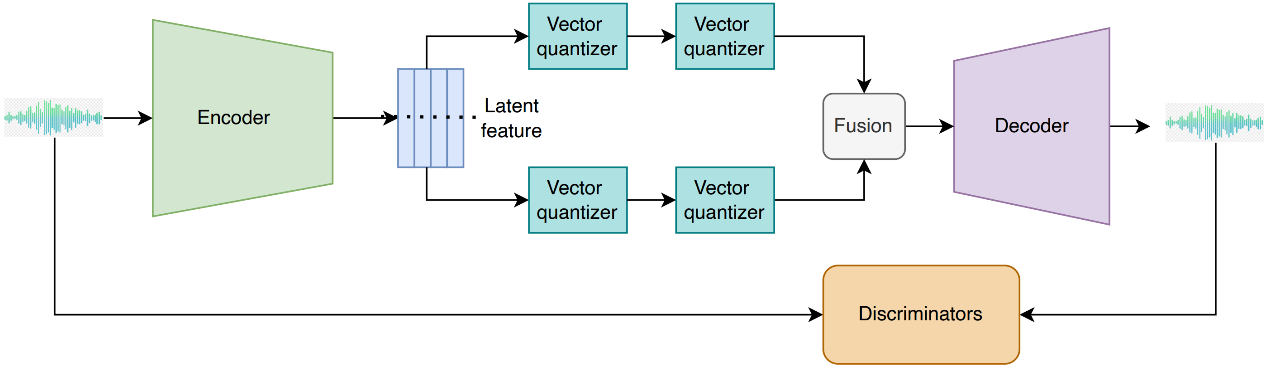
- Either is OK, but input and output should use the same (continuous or discrete)
- Pros and cons
  - Continuous tokens: should find good way for optimization (e.g., diffusion loss)
  - Discrete tokens: should increase entropy

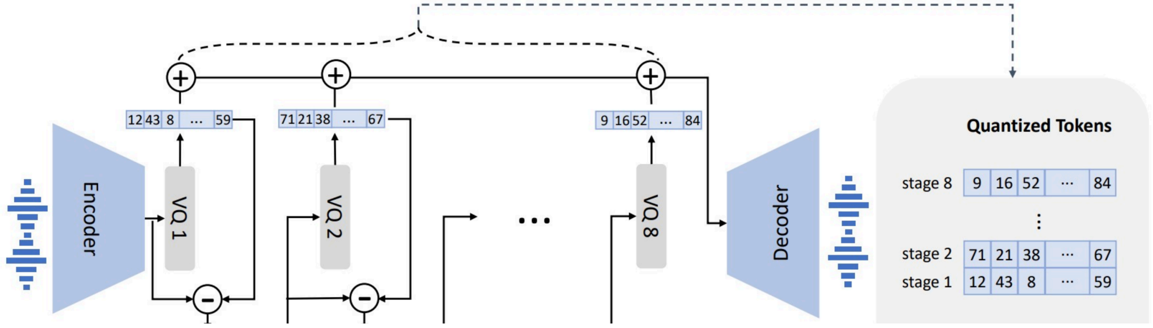
Representation	Pros	Cons
Continuous	Less Compression, More Bit, Larger Entropy Enough Perceptual/Semantic Information	Not Unified with LLM/NTP Hard for Optimization
Discrete	Unified with LLM/NTP Easy for Optimization	More Compression, Few Bit, Smaller Entropy Limited Perceptual/Semantic Information

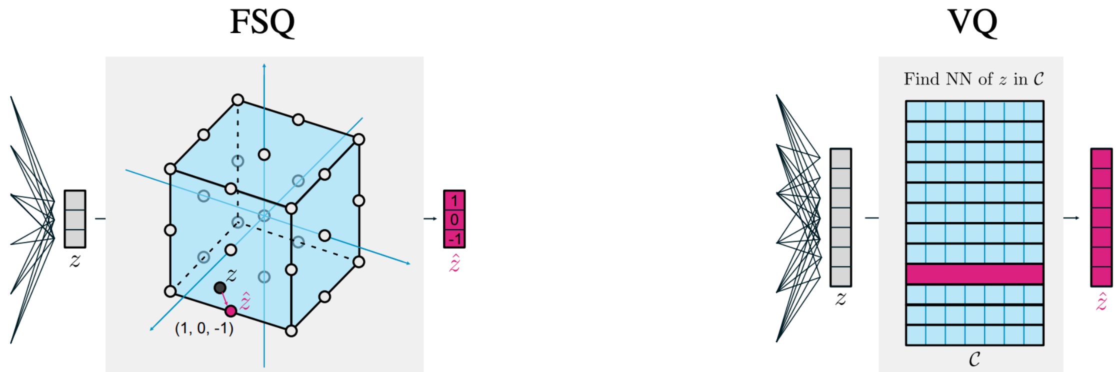
Topic 1.2——Representation: Continuous or Discrete

- How to increase entropy for discrete tokens?
  - Quantize one patch into multiple tokens, exponentially decrease/increase vocab size
  - Residual VQ (arXiv:2107.03312), Product/Group VQ (arXiv:2305.02765)
  - FSQ (Finite Scalar Quantization, arXiv:2309.15505)



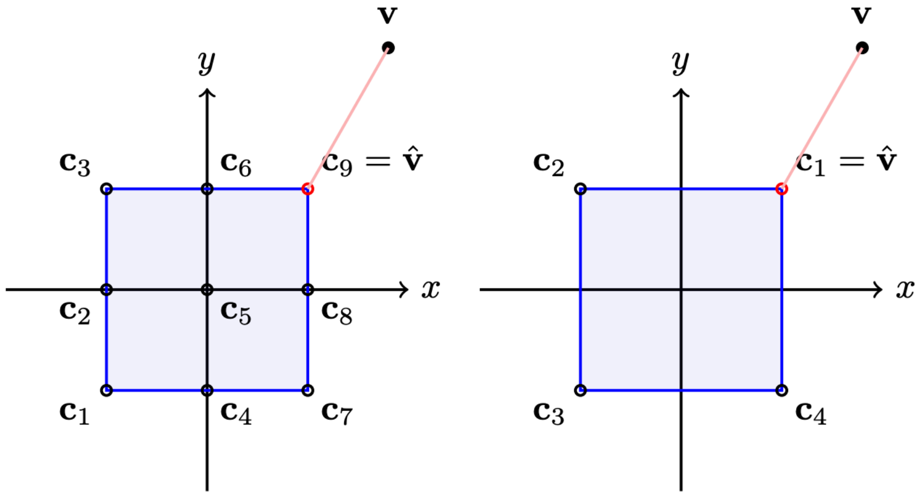




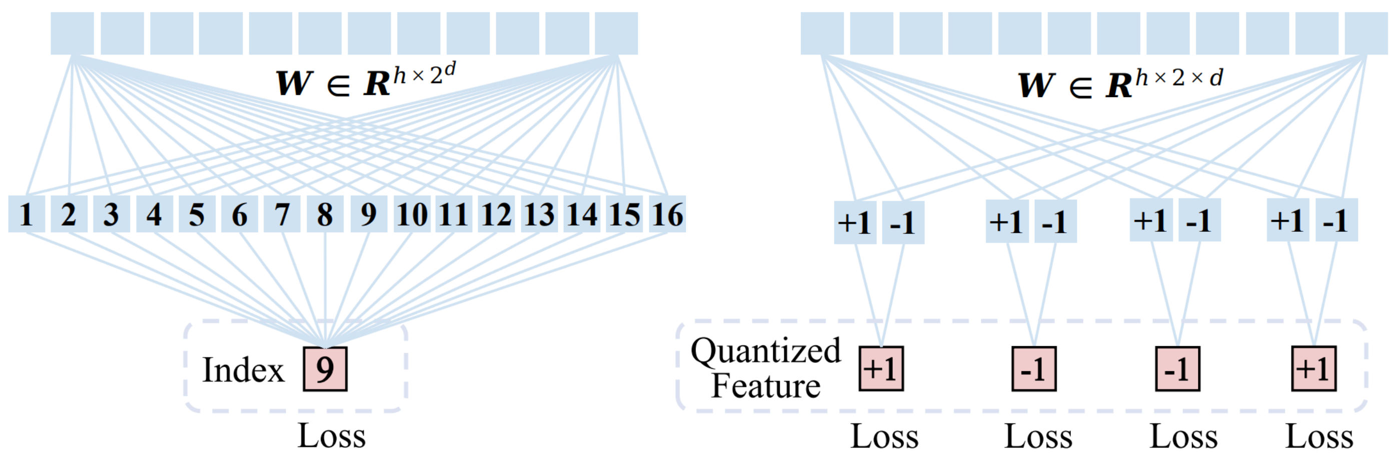


Topic 1.2——Representation: Continuous or Discrete

- How to increase entropy for discrete tokens?
  - Quantize one path into multiple tokens, exponentially decrease/increase vocab size)
  - An extreme case: Bitwise quantization
    - LFQ (Lookup-Free Quantization, arXiv:2310.05737)
    - Infinity (arXiv:2412.04431)



(b) FSQ (c) LFQ

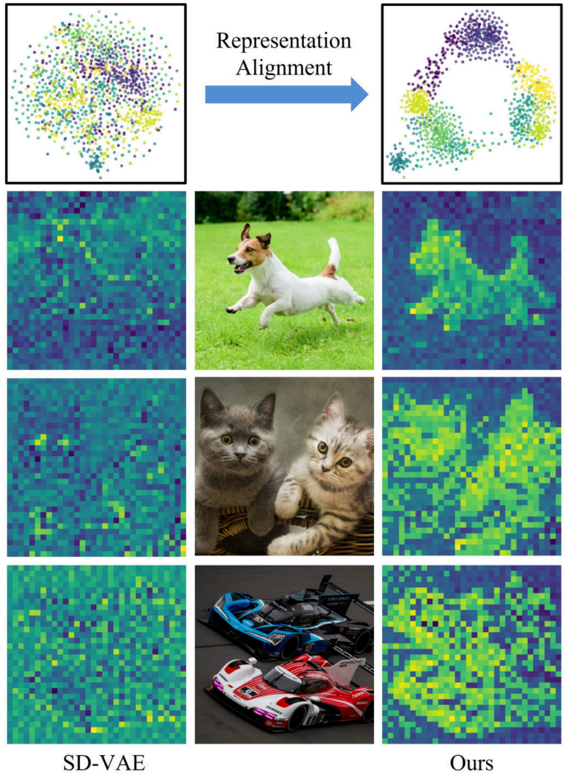


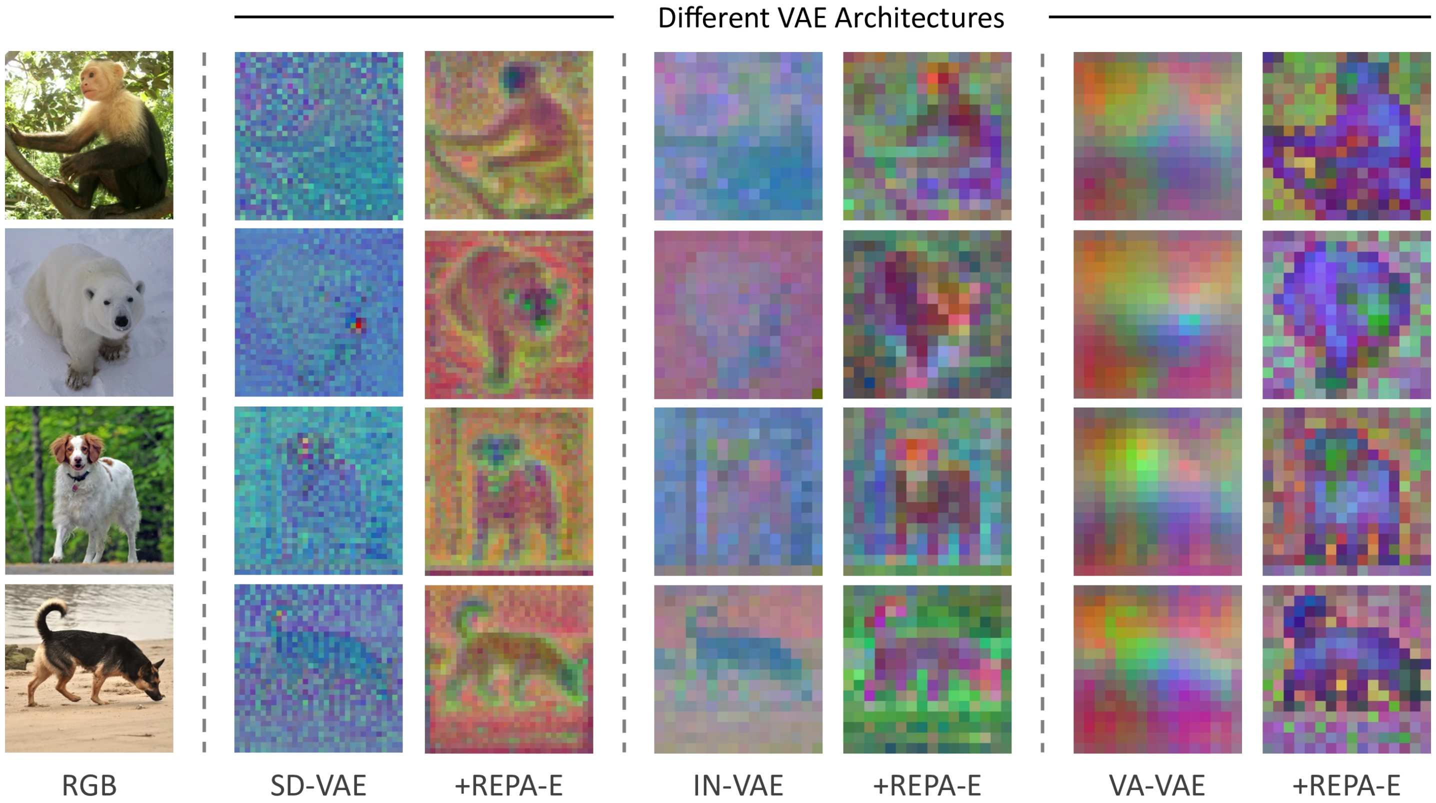
Topic 1.3——Representation: Issue of Pixel Reconstruction

- VAE with pixel/waveform reconstruction cannot learn human-aligned semantic feature
  - VAE with L1/L2 pixel/waveform loss learns too much high-frequency details, hard to differentiate from high-frequency noises
  - Hard to learn human-aligned semantic feature, not suitable for semantic task (both understanding and generation)
- Frequency vs Semantics (https://github.com/JamesCXH/research-ideas/blob/main/Frequency%20vs%20Semantics/Frequency_vs_Semantics.pdf)
  - Pixel-space objectives treat every pixel as equally reliable
  - In practice, this forces them to chase artefacts and sensor noise, yielding brittle features
- Align with Yann LeCun's JEPA
  - Predict in the representation space, instead of the raw data (pixel/waveform) space

Topic 1.3——Representation: Issue of Pixel Reconstruction

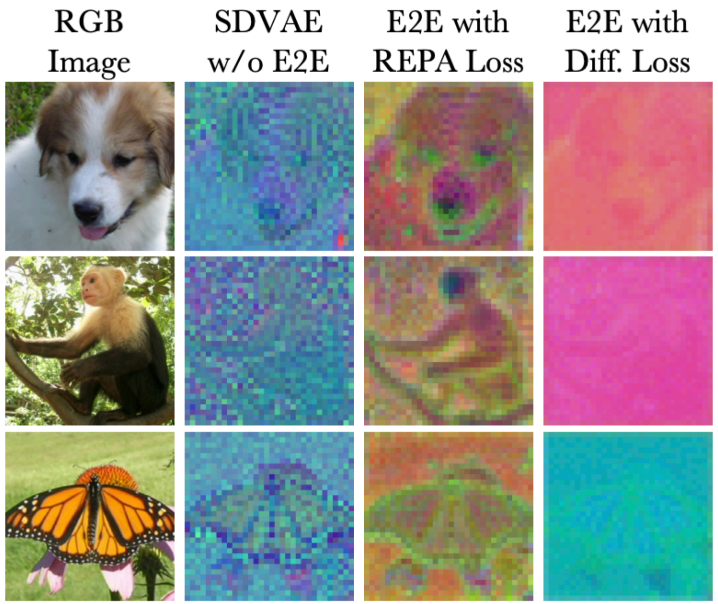
- VAE with pixel reconstruction cannot learn human-aligned semantic feature
  - High-level representation emerges only with explicit supervision
  - e.g., ReaLS (arXiv:2502.00359), REPA-E (arXiv:2504.10483)

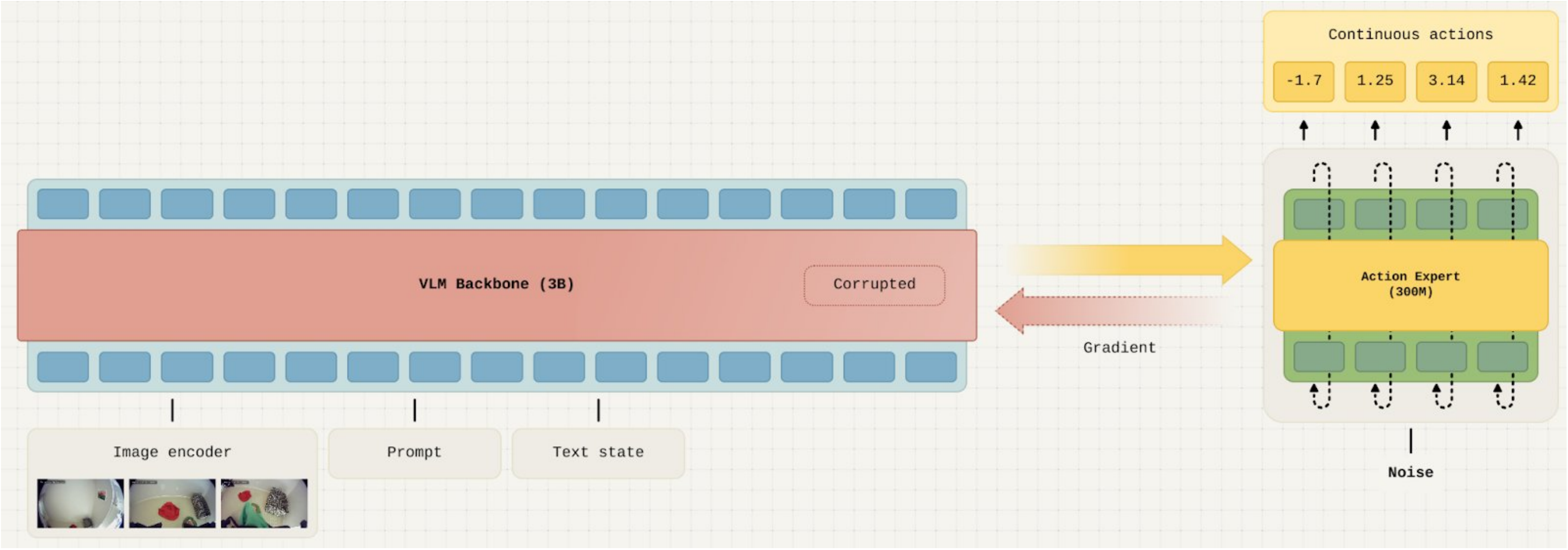




Topic 1.3——Representation: Issue of Pixel Reconstruction

- Diffusion loss corrupts the representation of LLM when jointly optimized
  - Evidences: REPA-E (2504.10483), MetaQuery (arXiv:2504.06256), Knowledge Insulation (arXiv:2505.23705)





Topic 1.3——Representation: Issue of Pixel Reconstruction

- Diffusion loss corrupts the representation of LLM when jointly optimized
  - Why?
    - Reason 1: Diffusion predicts raw data (e.g., pixel, waveform, continuous actions in VLA) or VAE latents (VAE latents are learnt by predicting raw data), which are full of high-frequency low-level details, and conflicts with LLM's high-level semantic information
    - Reason 2: the denoising behavior of continuous diffusion itself
      - Maybe diffusion with discrete token mask prediction will be better
      - The corruption behavior is agnostic of the representation it predicts
        - For perceptual tokens: discrete diffusion better than continuous diffusion
        - For semantic tokens: discrete diffusion better than continuous diffusion
        - But generally semantic is better than perceptual in terms of corruption

Topic 1.3——Representation: Issue of Pixel Reconstruction

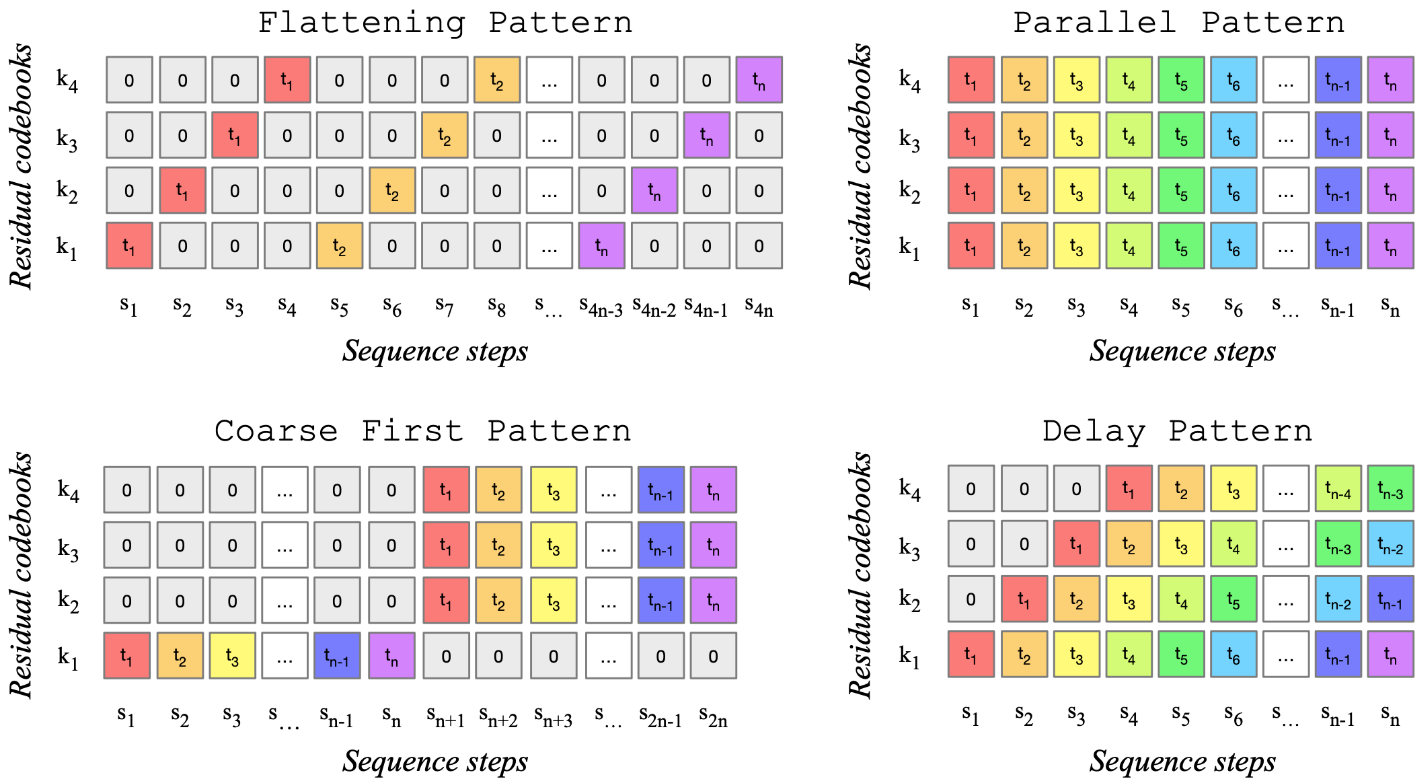
- Diffusion loss corrupts the representation of LLM when jointly optimized
  - Solutions
    - Diffusion predicts in the representation space, not in raw signal space (BLIP3-o, arXiv:2505.09568)
    - Align VAE latents with high-level representations (ReaLS, REPA)
    - Freeze LLM (MetaQuery, arXiv:2504.06256)
    - Discrete diffusion
    - Knowledge Insulation (arXiv:2505.23705)

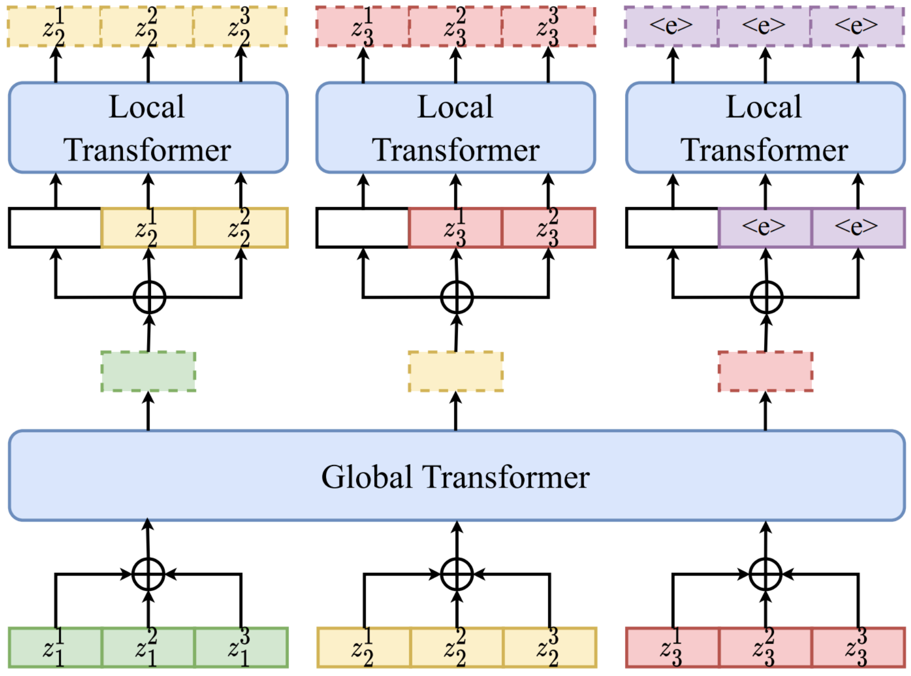
Topic 2.1——Modeling: AR + Discrete Tokens

- AR + discrete tokens
  - Align with LLM
  - Detokenizer uses diffusion to convert discrete tokens into raw data or perceptual feature
- Issues
  - Information not enough for both understanding and generation
  - Need increase entropy (multiple tokens) for representation

Topic 2.1——Modeling: AR + Discrete Tokens

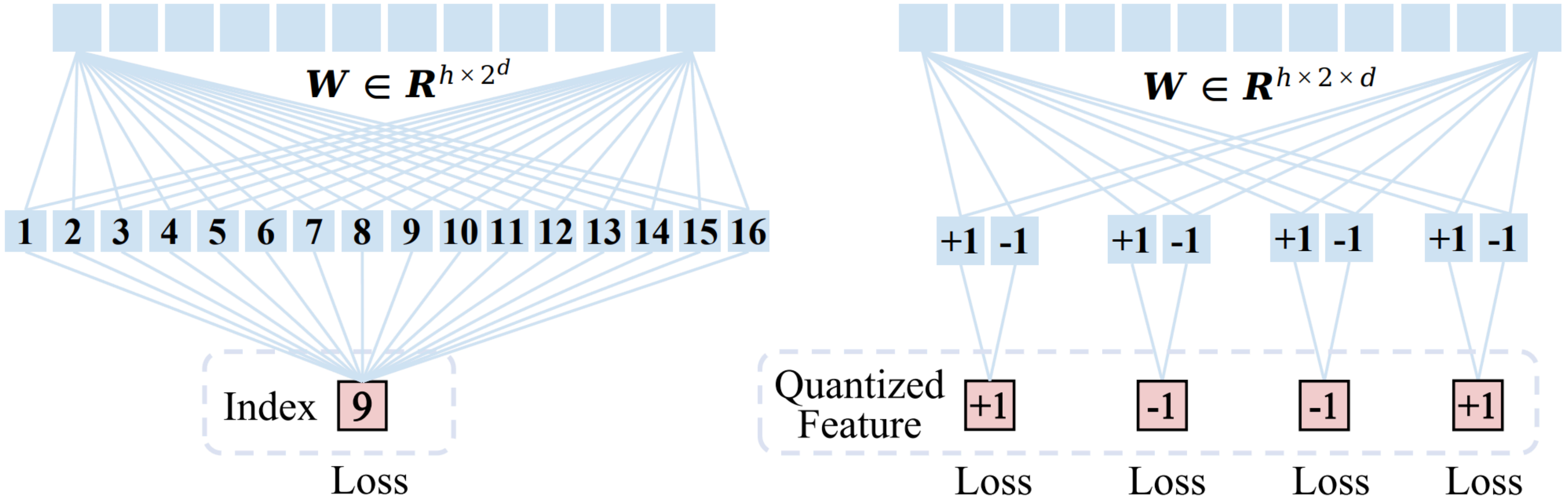
- AR + discrete tokens: predict multiple tokens
  - Interleaving patterns (arXiv:2306.05284)
  - Depth Transformer (UniAudio, arXiv:2310.00704; ViLA-U, arXiv:2409.04429)





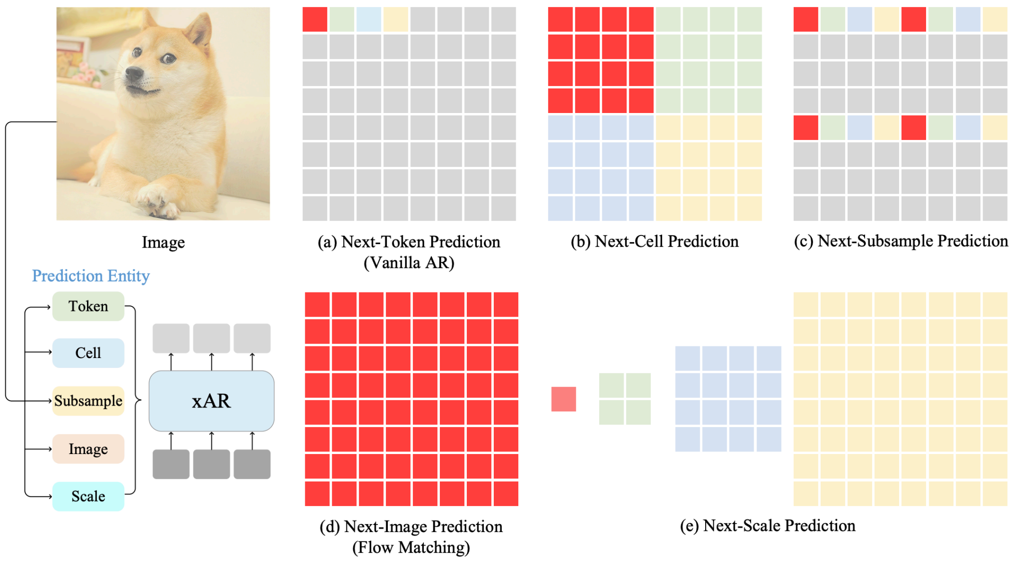
Topic 2.1——Modeling: AR + Discrete Tokens

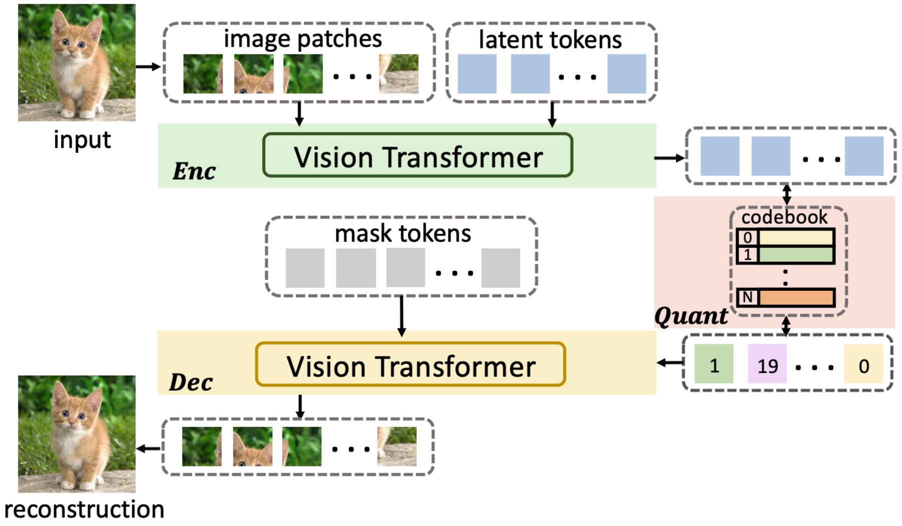
- AR + discrete tokens: predict multiple tokens
  - Extreme case: token as bits, predict next bit (Infinity, arXiv:2412.04431)

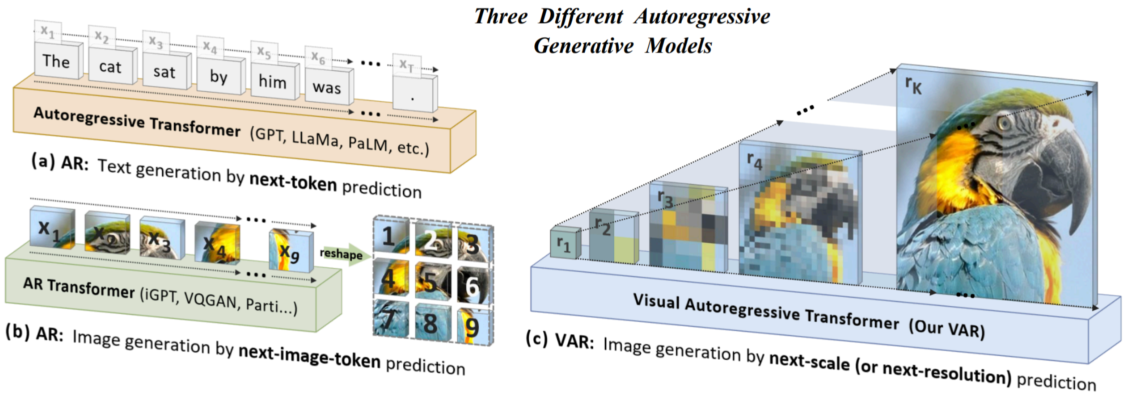


Topic 2.1——Modeling: AR + Discrete Tokens

- AR + discrete tokens: order prior
  - Rasterization
  - Next-X (xAR, arXiv:2502.20388)
  - VAR/Next-Scale (arXiv:2404.02905)
  - Query Tokens (TiTok, arXiv:2406.07550)

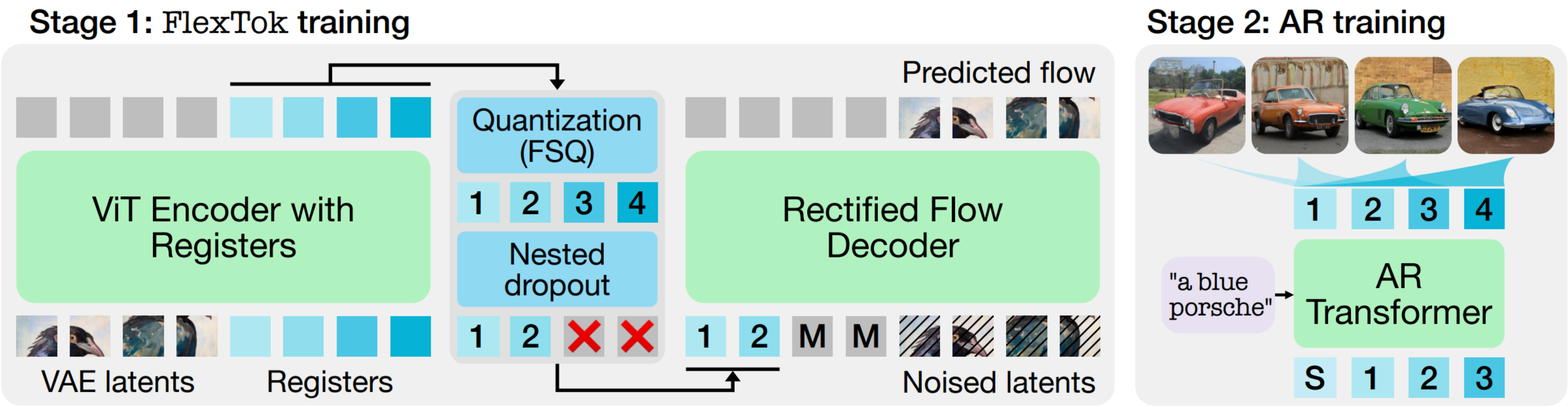






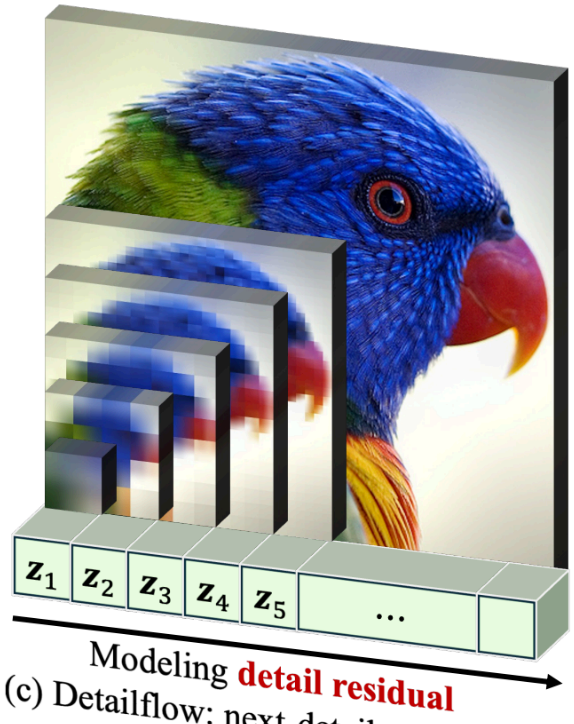
Topic 2.1——Modeling: AR + Discrete Tokens

- AR + discrete tokens: order prior
  - Nested dropout: learn ordered token sequences of flexible length by applying nested dropout (FlexTok, arXiv:2502.13967)
    - Coarse-to-fine: high-level concept first, then low-level details



Topic 2.1——Modeling: AR + Discrete Tokens

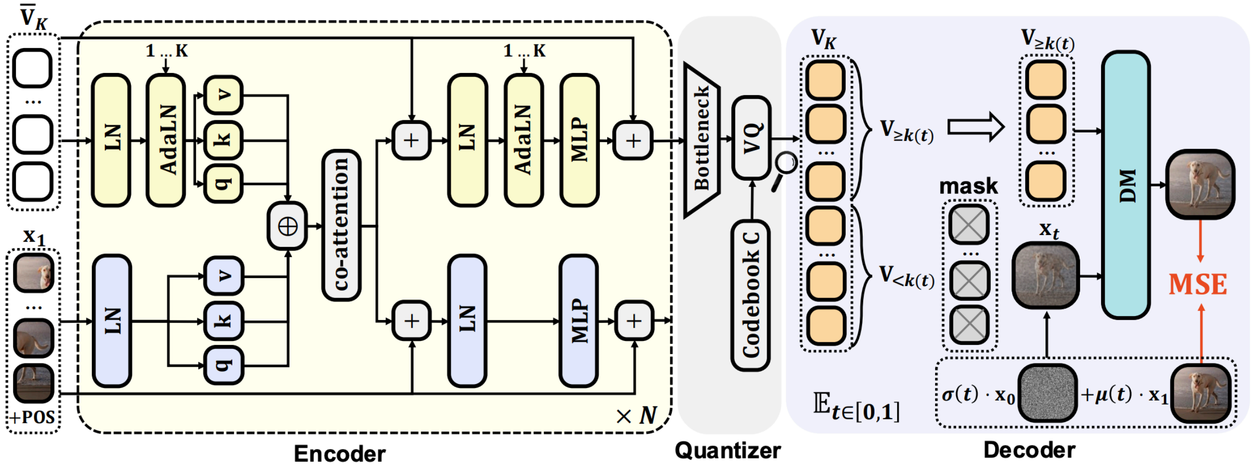
- AR + discrete tokens: order prior
  - Down/up-sampling order prior (DetailFlow, arXiv:2505.21473)
    - Coarse-to-fine: low-resolution first, then high-resolution

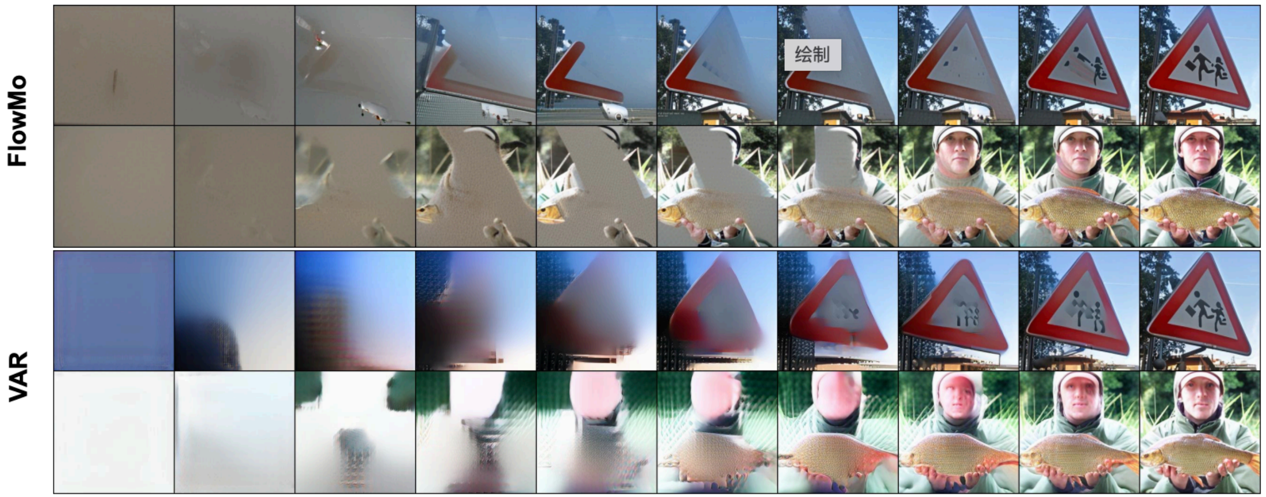


(c) Detailflow: next-detail prediction

Topic 2.1——Modeling: AR + Discrete Tokens

- AR + discrete tokens: order prior
    - Diffusion order prior
    - Coarse-to-fine: diffusion schedule
    - Selftok (arXiv:2505.07538)





Topic 2.1——Modeling: AR + Discrete Tokens

- AR + discrete tokens: order prior
  - Beyond coarse-to-fine?
    - e.g., some kind of semantic order like language? Query token is not enough

Topic 2.2——Modeling: AR + Continuous Tokens

- Pros and cons of current methods from the unified perspective

Version	Pros	Cons
V1 Per-Token Diffusion	Unified Modeling (Causal/NTP) Share understanding/generation parameter	
V2 Semi-AR + Diffusion		AR (understanding) and semi-AR not unified
V3 NAR + Diffusion		AR (understanding) and semi-AR not unified
V4 In-Place NAR/Diffusion (shared)	Share understanding/generation parameter	AR (understanding) and diffusion not unified
V5 In-Place NAR/Diffusion (non-shared)		AR (understanding) and diffusion not unified

Topic 2.2——Modeling: AR + Continuous Tokens

- Pros and cons of current methods from the unified perspective
  - Why V5 (e.g., Mogao, BAGEL) uses separate parameters for understanding and generation?
    - Gap: AR (LLM/understanding) and Diffusion (Generation)
    - Gap: Understanding uses semantic as input, generation use perceptual as output
    - Diffusion loss corrupt LLM if shared parameters
  - This is why some methods (MetaQuery, arXiv:2504.06256; Knowledge Insulation, arXiv:2505.23705) freeze LLM and then train diffusion models

  - However, if separate parameters, understanding and generation only interact in attention context, no synergy between understanding and generation!

Topic 2.2——Modeling: AR + Continuous Tokens

- Ideal Paradigm for AR + Continuous Tokens
  - V1, mainly AR, with per-token diffusion head
    - Satisfy unify modeling for multimodal understanding and generation (Req. 2)
      - Use AR, diffusion only serve as per-token loss (Req. 2.1)
      - Share model parameters, understanding and generation both use AR (Req. 2.2)

Topic 2.3——Modeling: Diffusion

- If use diffusion for unified multimodal understanding and generation, ideal paradigm is
  - Discrete diffusion for text and multimodal generation
    - Align with text, discrete diffusion is better than continuous diffusion
  - Block-wise diffsusion (AR + diffusion)
    - Intra-block use diffusion, inter-block use AR
    - From causal (AR) and non-causal (diffusion) to block-wise causal (block-wise diffusion)

Topic 2.4——Modeling: Input Loss

- Loss for input tokens
  - Towards unified modeling. Learn $P(x, y)$ instead of $P(y|x)$
- Case 1: If use AR + Discrete tokens
  - Input loss is cross-entropy, the same as NTP/LLM
- Case 2: If use AR + Continuous tokens
  - Per-token diffusion loss for continuous tokens
- Case 3: If use block-wise diffusion
  - Input no loss, only serves as condition (last segment with no noises) in block-wise diffusion

- For Case 1 and 2, tokenizer should be causal
- For Case 3, tokenizer can be non-causal

Topic 3.1——Omni-Modal: Lesson from Audio

- Representation
  - Semantic vs Acoustic (Perceptual)
    - Prefer semantic (e.g., CosyVoice, arXiv:2407.05407) over perceptual (e.g., VALL-E, arXiv:2301.02111)
  - Continuous vs Discrete
    - Input continuous, output discrete (e.g., Kimi-Audio, arXiv:2504.18425)
    - or discrete with multiple tokens (e.g., RVQ in Moshi, arXiv:2410.00037)
- Modeling
  - LLM, AR, next token prediction
  - Diffusion as detokenizer

Topic 3.1——Omni-Modal: Lesson from Audio

- Why audio domain adopts unified understanding and generation earlier/quicker than vision?
  - Speech aligns with text explicitly/literally, while vision align with text implicitly
  - Speech is 1D, consistent with text, while vision is 2D or 3D
  - Speech contains less entropy/information than vision, easier for unified modeling
- Unified model in audio domain
  - Satisfy Req. 1.1 (semantic), Req. 2.1 (AR + Diffusion cascaded pipeline), and Req. 2.2 (share parameters)
  - Not satisfy Req. 1.2 (input continuous, output discrete)
    - But discrete is directly quantized from continuous, still in the same space
    - For audio, considering continuous with 6.25Hz is enough for input
      - Option 1: input/output use continuous with 6.25 Hz
      - Option 2: input/output use discrete with higher Hz or more tokens per frame

Topic 3.2——Omni-Modal: Omni Understanding/Generation

- Representation
  - Image/video/audio all use semantic input and output
    - Align with text, but also reconstruct raw data to some extent
      - e.g., in speech, reconstruct text or VAE latent instead of raw waveform
    - No matter text captioning is missing or not, align video and audio (huge amount of internet data)
  - Discrete or continuous
- Modeling
  - LLM with discrete token in/out, with multiple tokens to increase entropy
  - Or LLM with continuous features as in/out, with diffusion head for continuous modeling

Summary of Research Topics

- Topic 1.1——Representation: Semantic or Perceptual
- Topic 1.2——Representation: Continuous or Discrete
- Topic 1.3——Representation: Issue of Pixel Reconstruction
- Topic 2.1——Modeling: AR + Discrete Tokens
- Topic 2.2——Modeling: AR + Continuous Tokens
- Topic 2.3——Modeling: Diffusion
- Topic 2.4——Modeling: Input Loss
- Topic 3.1——Omni-Modal: Lesson from Audio
- Topic 3.2——Omni-Modal: Omni Understanding/Generation

Ideal Paradigm for Unified Understanding/Generation ?

Paradigm	Representation	Modeling
1	Semantic (with some perceptual) Discrete tokens	AR
2	Semantic (with some perceptual) Continuous tokens	AR + Per-token diffusion loss
3	Semantic (with some perceptual) Discrete tokens	Block-Wise Diffusion
new ?	?	?

Opinions are on my own

Welcome discussions and suggestions

Xu Tan
tanxu2012@gmail.com

扫一扫上面的二维码图案，加我为朋友。