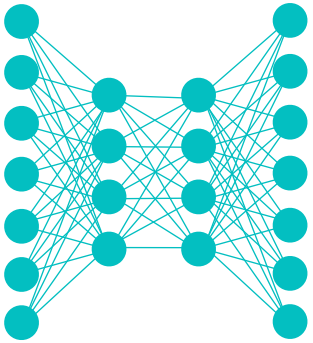
Lecture Notes for

Neural Networks and Machine Learning

Generative Models
Stable Diffusion



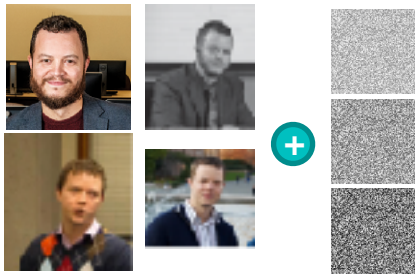
Logistics and Agenda

- Logistics
  - Grading Update
  - New due date Lab 4
  - Team members!
- Agenda
  - *VAEs (done)*
  - *Stable Diffusion Basics (mostly done)*
  - Student Paper Presentation
  - Final Project Town Hall
  - Stable Diffusion 3

Stable Diffusion

The Diffusion Process, Simplified

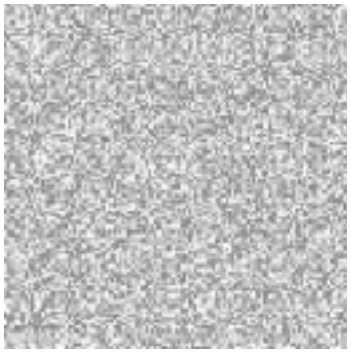
- ***Guiding*** Example: Predict noise sample in an image



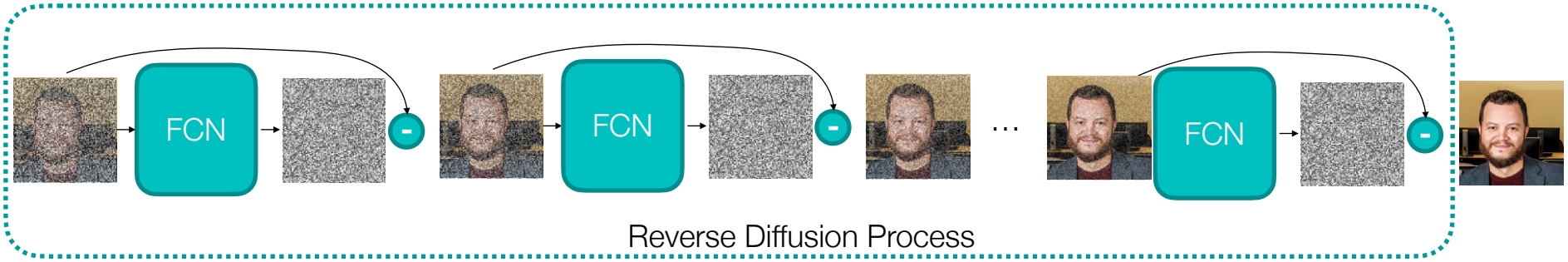
Create Dataset

Add in "t" steps of noise

Full Convolutional Network, Noise Predictor



Train Network on Synthetic Examples, various values of "*t*"



Reverse Diffusion Process

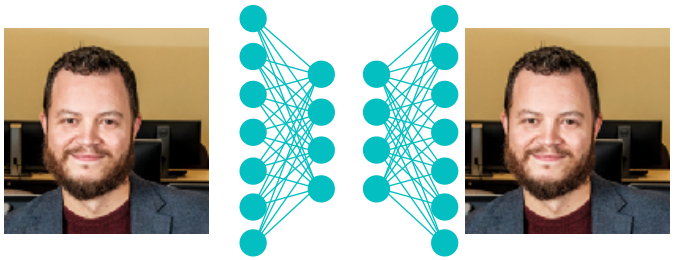
- Now we could generate great looking images from noise!!

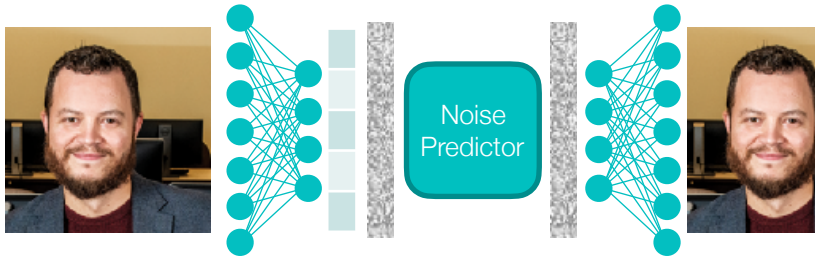
Departure to Latent Space

Departure to Latent Space Our approach starts with the analysis of already trained diffusion models in pixel space: Fig. 2 shows the rate-distortion trade-off of a trained

Rombach et al., 2022, https://arxiv.org/pdf/2112.10752.pdf

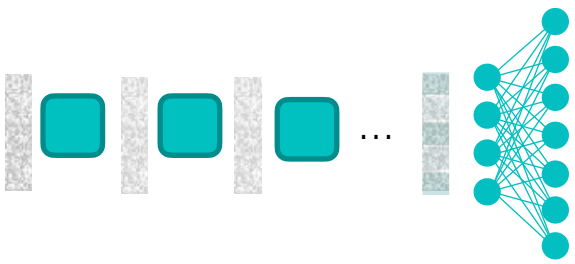
- Start with a nice VAE
- Train noise prediction in latent space
- Perform diffusion in latent Space
- Generate from VAE decoder



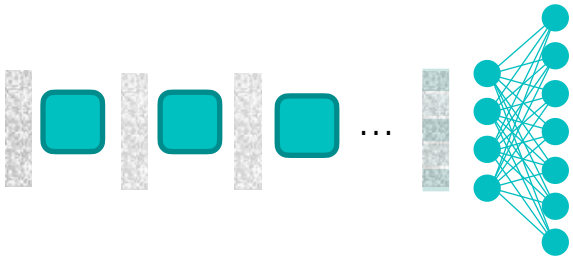


Random Noise

$z \sim \mathcal{N}(0,I)$



$z \sim \mathcal{N}(0,I)$



Eric Larson, Brown University

Eric Larson, Disney Animator

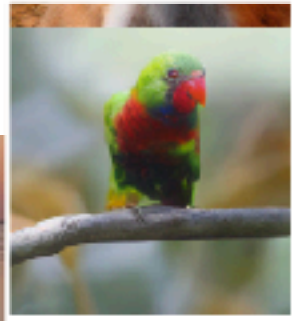
Examples of Diffusion in Latent Space

CelebA-HQ 256 × 256				FFHQ 256 × 256			
Method	FID ↓	Prec. ↑	Recall ↑	Method	FID ↓	Prec. ↑	Recall ↑
DC-VAE [63]	15.8	-	-	ImageBART [21]	9.57	-	-
VQGAN+T. [23] (k=400)	10.2	-	-	U-Net GAN (+aug) [77]	10.9 (7.6)	-	-
PGGAN [39]	8.0	-	-	UDM [43]	5.54	-	-
LSGM [93]	7.22	-	-	StyleGAN [41]	4.16	0.71	0.46
UDM [43]	7.16	-	-	ProjectedGAN [76]	**3.08**	0.65	0.46

Method	FID↓	IS↑	Precision↑	Recall↑	N_{params}	
BigGan-deep [3]	6.95	203.6±2.6	**0.87**	0.28	340M	-
ADM [15]	10.94	100.98	0.69	**0.63**	554M	250 DDIM steps
ADM-G [15]	4.59	186.7	0.82	0.52	608M	250 DDIM steps
LDM-4 (ours)	10.56	103.49±1.24	0.71	0.62	400M	250 DDIM steps
LDM-4-G (ours)	**3.60**	**247.67**±5.59	**0.87**	0.48	400M	250 steps, c.f.g [32], $s = 1.5$

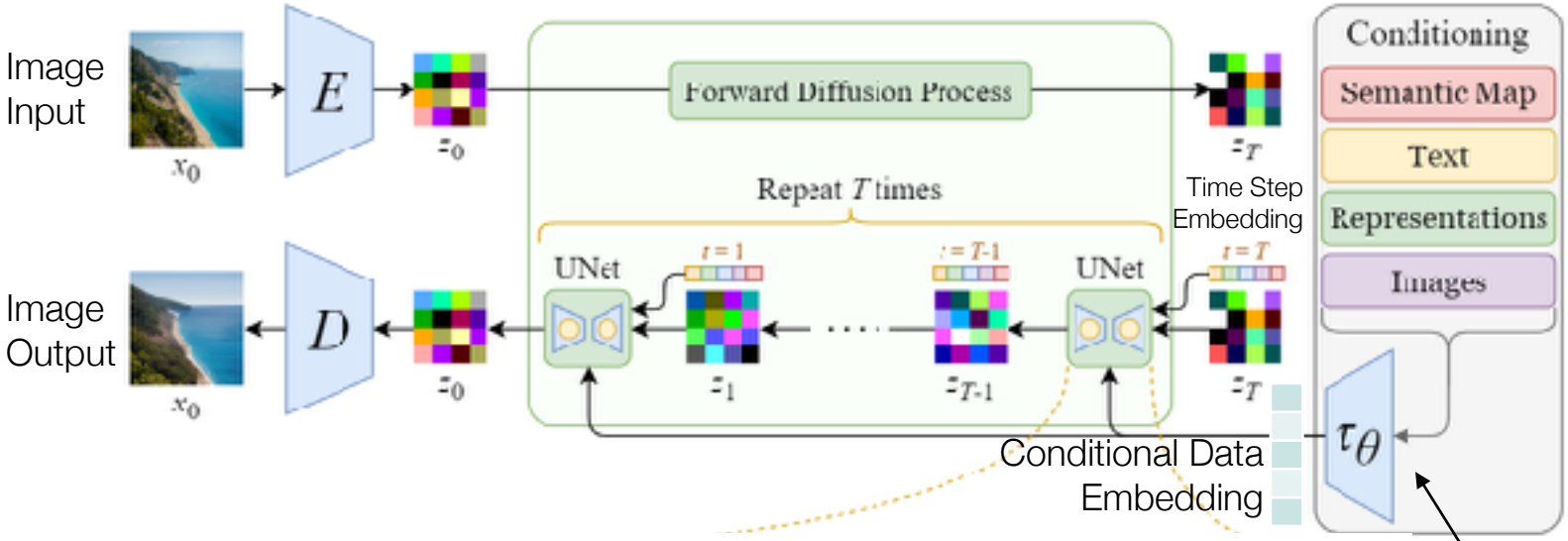
LDM-8 (ours, 200 s) 4.02 0.64 0.52 *LDM-4* (ours, 200 s) 2.95 0.66 0.46

Table 1. Evaluation metrics for unconditional image synthesis.



Rambach et al., 2022, High-Resolution Image Synthesis with Latent Diffusion Models https://arxiv.org/pdf/2112.10752.pdf

Conditioning for Denoising, Overview



Just Cross Attention with Dropout!!

Rambach et al., 2022, High-Resolution Image Synthesis with Latent Diffusion Models https://arxiv.org/pdf/2112.10752.pdf

Unpacking "Text Conditioning"

$$\mathscr{L}(x_0) = -\frac{1}{2}\mathbf{E}\left[\|\epsilon_\theta(z_t, t, c_{text}) - \epsilon\|^2\right]$$

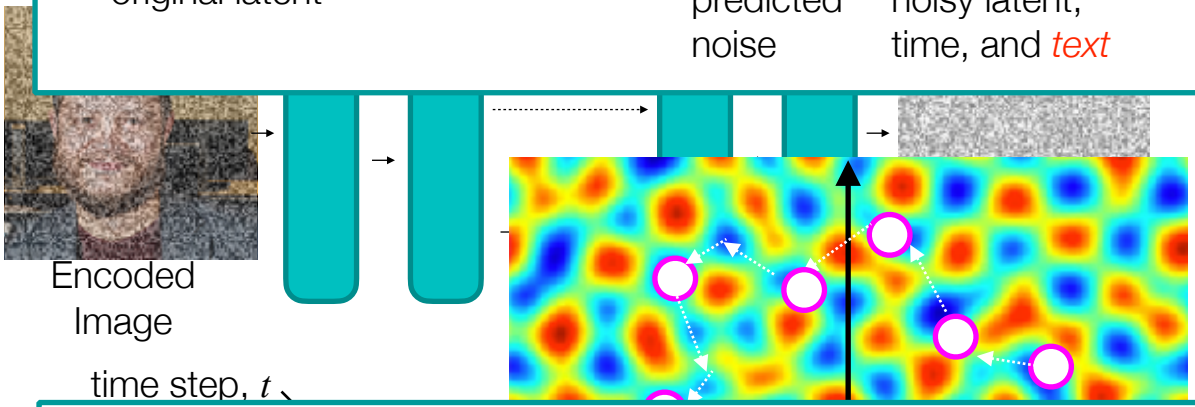
original latent

predicted noise

noisy latent, time, and *text*

actual noise

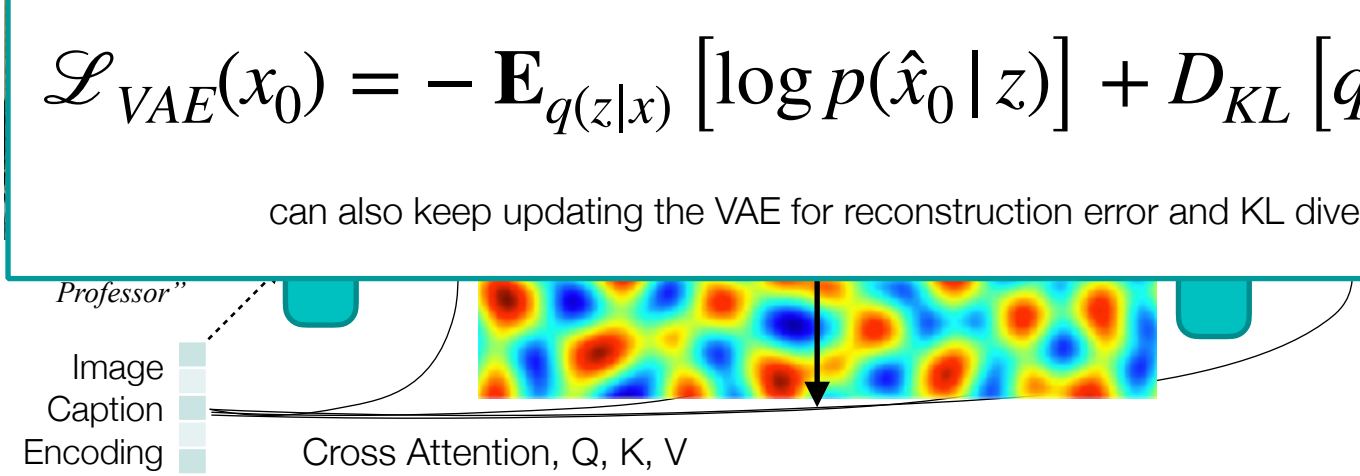
Full Convolutional Network, Noise Predictor



Model denoising is **sensitive to the content** within it…
Model learns how to De-noise based on **text conditioning!**

$$\mathscr{L}_{VAE}(x_0) = -\mathbf{E}_{q(z|x)}\left[\log p(\hat{x}_0|z)\right] + D_{KL}\left[q(z|x_0)\|p(z)\right]$$

can also keep updating the VAE for reconstruction error and KL divergence



This noise is a **step** in the latent space, can interpret as a **gradient** trajectory

Are all text embeddings created equal?

- Would be nice to have unified text and image embeddings

Image Encoder Pre-trained → W_i → μ

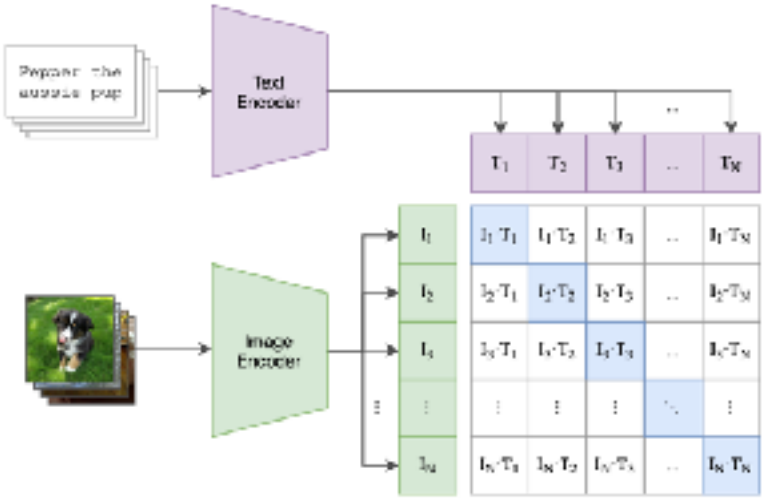
"A Narcissistic Professor"

Text Encoder Pre-trained → W_t → μ

Contrastive Loss

Positive Examples: Make similar

Negative Examples: Make dissimilar



Contrastive Language-Image Pre-training

CLIP

Radford et al. 2021, Learning Transferable Visual Models From Natural Language Supervision, https://arxiv.org/pdf/2103.00020.pdf

Many questions remaining…

- How to add noise in the latent space?
- How many time steps to add noise before training denoising algorithm?
- How many denoising steps should we take?
- What architecture to use for denoising?
  - And what about its depth, parameters, etc. ?
- What size images or mixture of images sizes to use?
- What text conditioning embeddings are most versatile?
- Are labels valuable for conditioning?
- …And many other questions for research community to investigate…

Student Paper Presentation

High-Resolution Image Synthesis with Latent Diffusion Models

(A.K.A. LDM & Stable Diffusion)

Robin Rombach[1,2], Andreas Blattmann[1,2], Dominik Lorenz[1,2], Patrick Esser[3],
Björn Ommer[1,2]

[1]LMU Munich, [2]IWR, Heidelberg University, [3]Runway

CVPR 2022 (ORAL)

LMU

IWR
Interdisciplinary Center for Scientific Computing

Using LoRA for Efficient Stable Diffusion Fine-Tuning

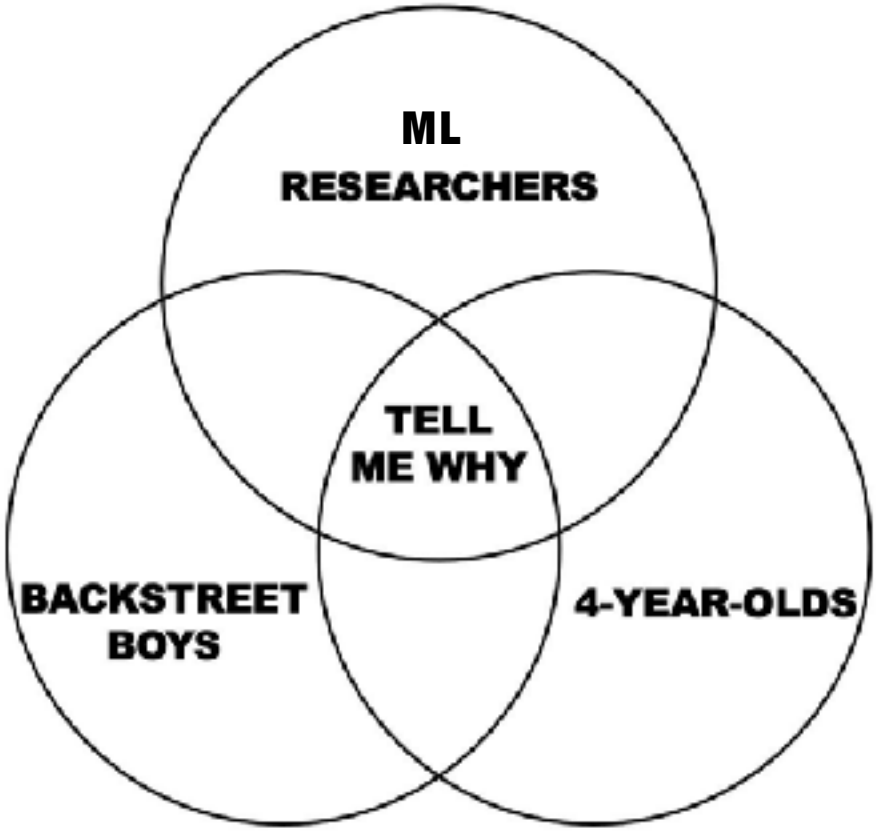
Published January 26, 2023

Update on GitHub

Pedro Cuenca

Sayak Paul

Final Project Draft Town Hall



Final Project Presentation

- Presentation:
  - 10-12 minutes presenting (polished)
    - Do not read slides to me
    - Consider having a demo
  - This should mirror a conference presentation about your work:
    - Intro and related work (2-3 minutes)
    - Methods (architecture, experiments, etc.) (3-5 minutes)
    - Results and conclusion (3-5 minutes)
  - Additional 5 minutes questions from instructor
- Sign up for a time to present (20 minute slots):
  - See canvas (lets go there)

Stable Diffusion 3

Scaling Rectified Flow Transformers for High-Resolution Image Synthesis

Patrick Esser * **Sumith Kulal** **Andreas Blattmann** **Rahim Entezari** **Jonas Müller** **Harry Saini** **Yam Levi**
Dominik Lorenz **Axel Sauer** **Frederic Boesel** **Dustin Podell** **Tim Dockhorn** **Zion English**
Kyle Lacey **Alex Goodwin** **Yannik Marek** **Robin Rombach** *
Stability AI

Prompt: Epic anime artwork of a wizard atop a mountain at night casting a cosmic spell into the dark sky that says "Stable Diffusion 3" made out of colorful energy

https://stability.ai/news/stable-diffusion-3

https://arxiv.org/pdf/2403.03206.pdf

Stable Diffusion 3

- Released March 5, 2024
- 28 Pages of background, explanation, methods, results—maybe the definitive paper in the field? (assumes you understand probability flows and ODEs as flows)
- Lots of ablation studies on parameter choices
- Evaluated in the right way. I love this paper!

an old rusted robot wearing pants and a jacket riding skis in a supermarket.

smiling cartoon dog sits at a table, coffee mug on hand, as a room goes up in flames. "This is fine," the dog assures himself.

Vector Gradient Flow Scalers

- Define Loss as a vector field, after lots of computations:

$$\mathcal{L}_w(x_0) = -\frac{1}{2}\mathbb{E}_{t\sim\mathcal{U}(t),\epsilon\sim\mathcal{N}(0,I)}\left[w_t\lambda'_t\|\epsilon_\Theta(z_t,t)-\epsilon\|^2\right]$$

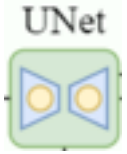
x0 is latent of original image

sampling at time step t

with noise added to image latent

defines SNR and different scaling factors, depends on how noise added and time steps

~Predicted Noise at time t



Actual Noise

- In paper, they compare lots of different scaling variations with different noise models:

Rectified flow: $z_t = (1-t)x_0 + t\epsilon \qquad w_t^{\text{RF}} = \frac{t}{1-t}$

EDM: $z_t = x_0 + b_t\epsilon \qquad b_t = \exp F_{\mathcal{N}}^{-1}(t|P_m, P_s^2) \quad w_t^{\text{EDM}} = \mathcal{N}(\lambda_t|-2P_m,(2P_s)^2)(e^{-\lambda_t}+0.5^2)$

Cosine: $z_t = \cos(\frac{\pi}{2}t)x_0 + \sin(\frac{\pi}{2}t)\epsilon \qquad w_t = e^{-\dot{\lambda}_t/2}$

LDM Linear: $z_t = a_t x_0 + b_t\epsilon \qquad b_t = \sqrt{1-a_t^2}, \quad a_t = (\prod_{s=0}^{t}(1-\beta_s))^{\frac{1}{2}} \quad \beta_t = \left(\sqrt{\beta_0} + \frac{t}{T-1}(\sqrt{\beta_{T-1}} - \sqrt{\beta_0})\right)^2$

Sampling t

- In paper, look at lots of ways to sample the t across the various distributions: $\mathcal{L}_w(x_0) = -\frac{1}{2}\mathbb{E}_{t\sim\mathcal{U}(t),\epsilon\sim\mathcal{N}(0,I)}\left[w_t\lambda_t'\|\epsilon_\Theta(z_t,t)-\epsilon\|^2\right]$

Uniform Distribution: $\mathcal{U}(t)$

Logit-normal: $\pi_{\ln}(t;m,s) = \frac{1}{s\sqrt{2\pi}}\frac{1}{t(1-t)}\exp\left(-\frac{(\text{logit}(t)-m)^2}{2s^2}\right)$,

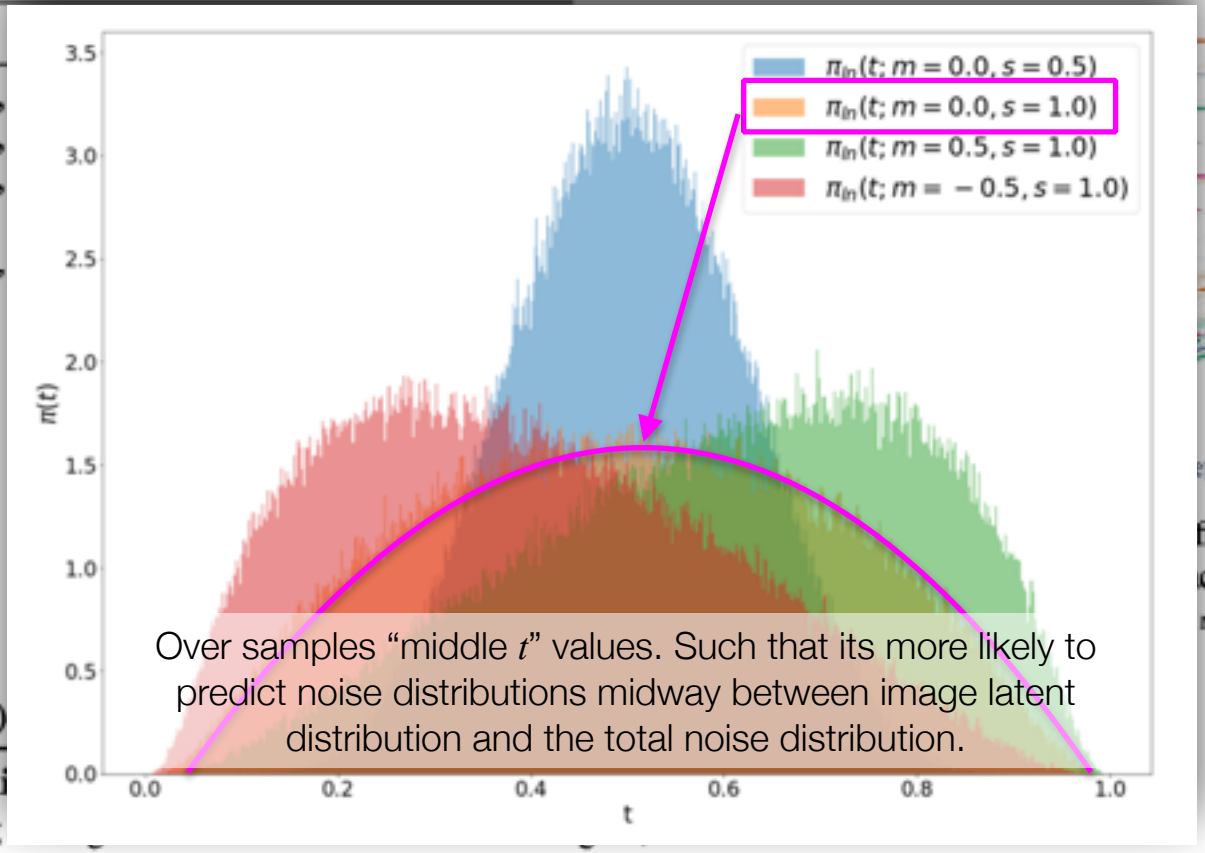
Heavy Tailed Mode: $f_{\text{mode}}(u;s) = 1-u-s\cdot\left(\cos^2\left(\frac{\pi}{2}u\right)-1+u\right)$

CosMap: $\pi_{\text{CosMap}}(t) = \left|\frac{d}{dt}f^{-1}(t)\right| = \frac{2}{\pi-2\pi t+2\pi t^2}$.

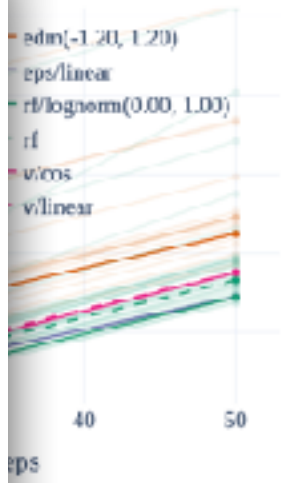
Ablation for Variant and Sampling

variant	rank averaged over
rf/lognorm(0.00,	
rf/lognorm(1.00,	
rf/lognorm(0.50,	
rf/mode(1.29)	
rf/lognorm(0.50,	
eps/linear	
rf/mode(1.75)	
rf/cosmap	
edm(0.00, 0.60)	
rf	
v/linear	
edm(0.60, 1.20)	
v/cos	
edm/cos	
edm/rf	
edm(-1.20, 1.20)	

Table 1. **Global ranki**
non-dominated sorting
two datasets and different sampling settings.



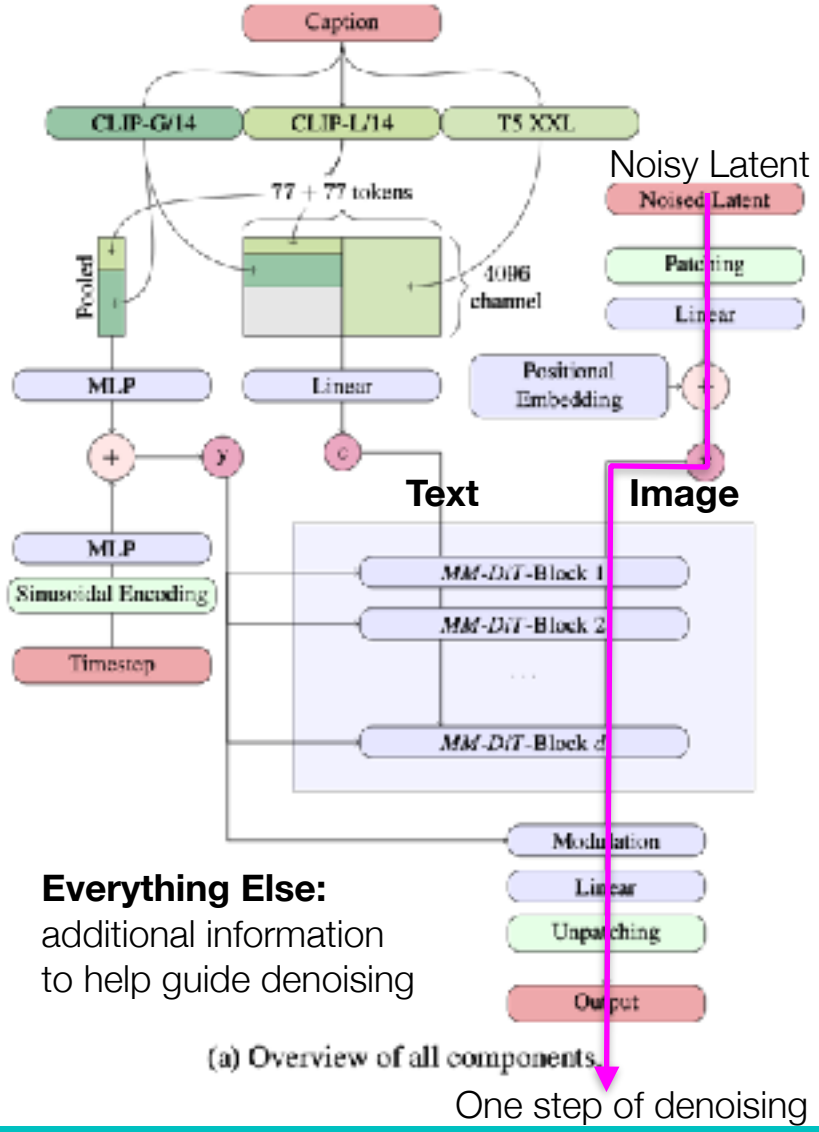
Over samples "middle t" values. Such that its more likely to predict noise distributions midway between image latent distribution and the total noise distribution.



ficient. Rectified Flows
cn sampling fewer steps.
m (0.00, 1.00) re-

Rectified flow is always one of the better performers, especially using logit-normal sampling m=0, s=1

The Architecture: Overview



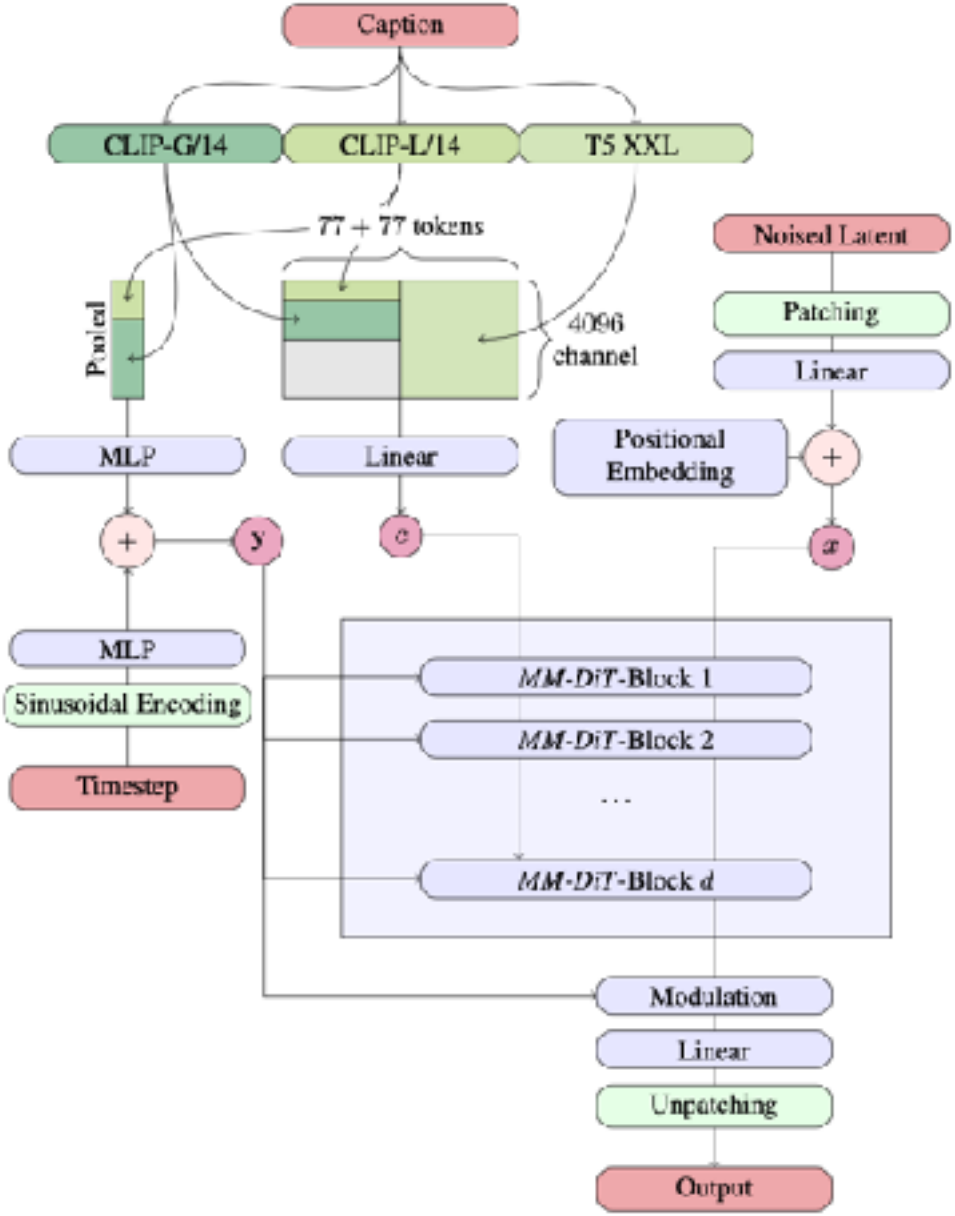
Everything Else:
additional information
to help guide denoising

Three **Conditioning Text / Image encoders** used, via concatenation

Separate text modality and image modality before feeding into the noise prediction network.

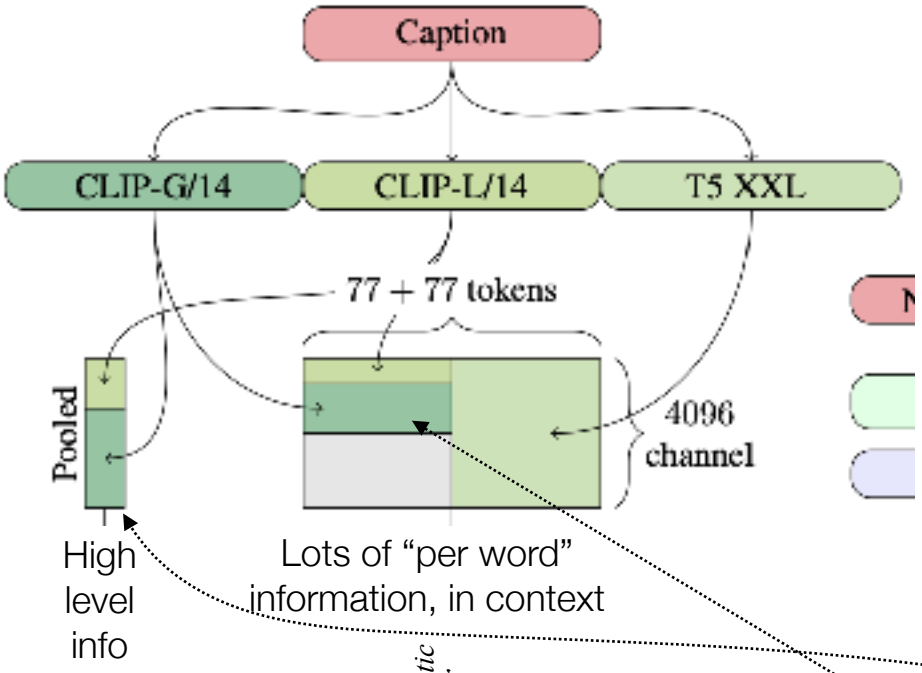
Each **MM-DiT** is just a **transformer** working on the concatenated modalities

In paper, investigated many "depth scaling" techniques, found **bigger is always better**. And that there was no saturation, so could probably get even better results, but the **GPUs do not have enough memory**…

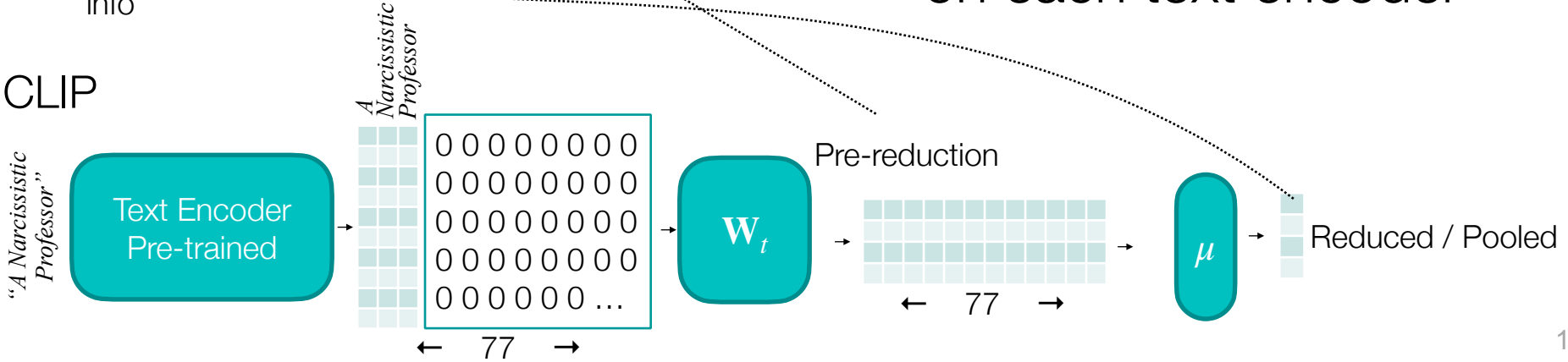


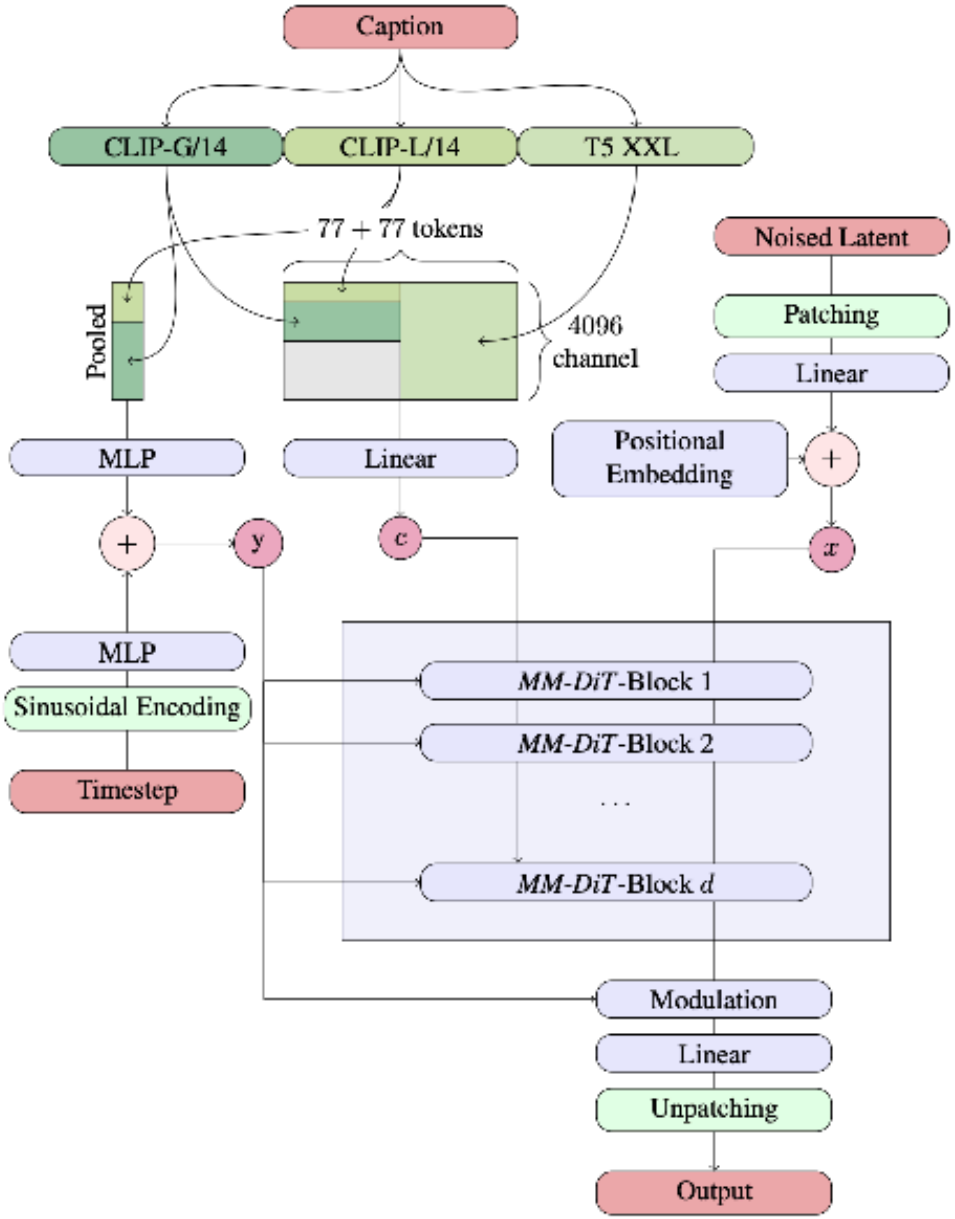
(a) Overview of all components.

The Architecture: Text Input



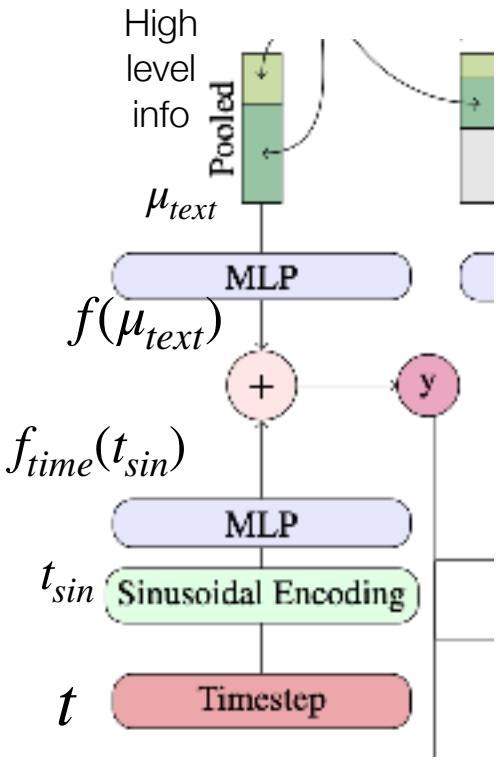
- Max caption length 77
- Feed pre-reduced CLIP embeddings
- Combine with T5 encoder for each token
- Add "modality dropout" to reduce dependence on each text encoder





(a) Overview of all components.

The Architecture: Time info



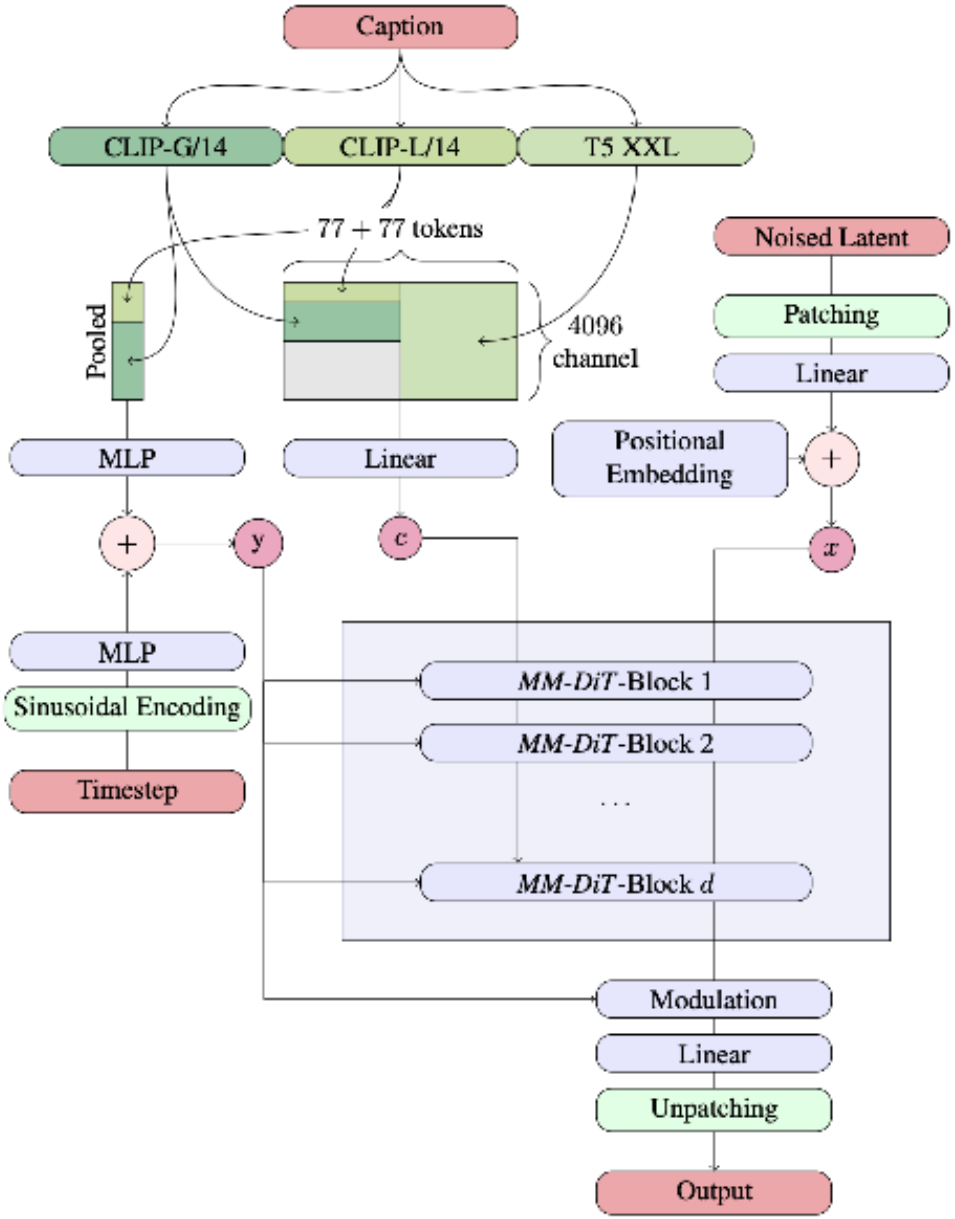
- Want architecture to have information of how much noise to remove:

$$z_t = (1-t)\cdot x_0 + t\cdot\epsilon$$

- So tell it the value of t via sinusoidal position encoding of the same size as pooled text information
- Combine both through "addition" and will use throughout the network

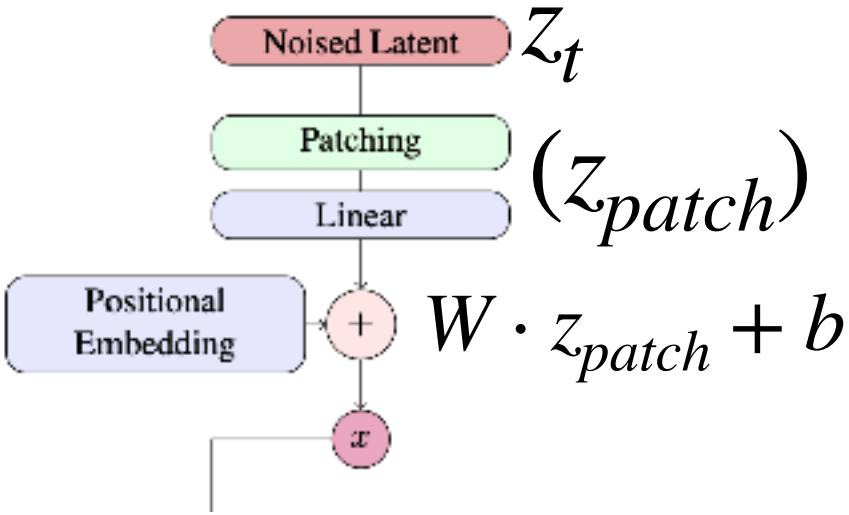
$$y = f(\mu_{text}) + f_{time}(t_{sin})$$

y now encodes lots of information about the text (in general) and the amount of noise (t).

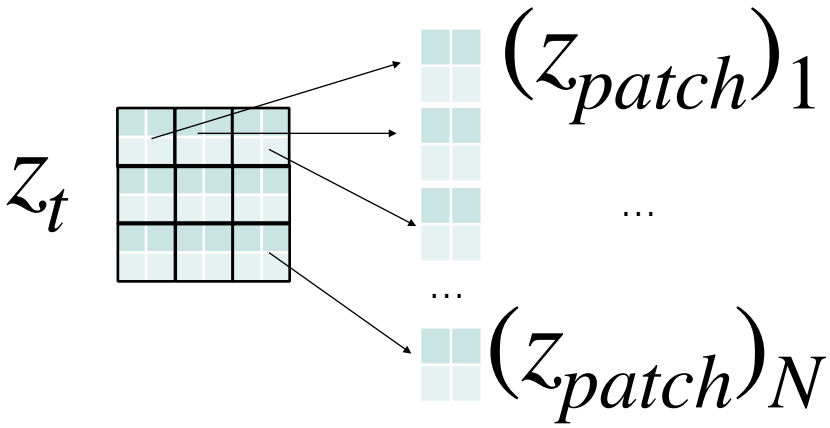


(a) Overview of all components.

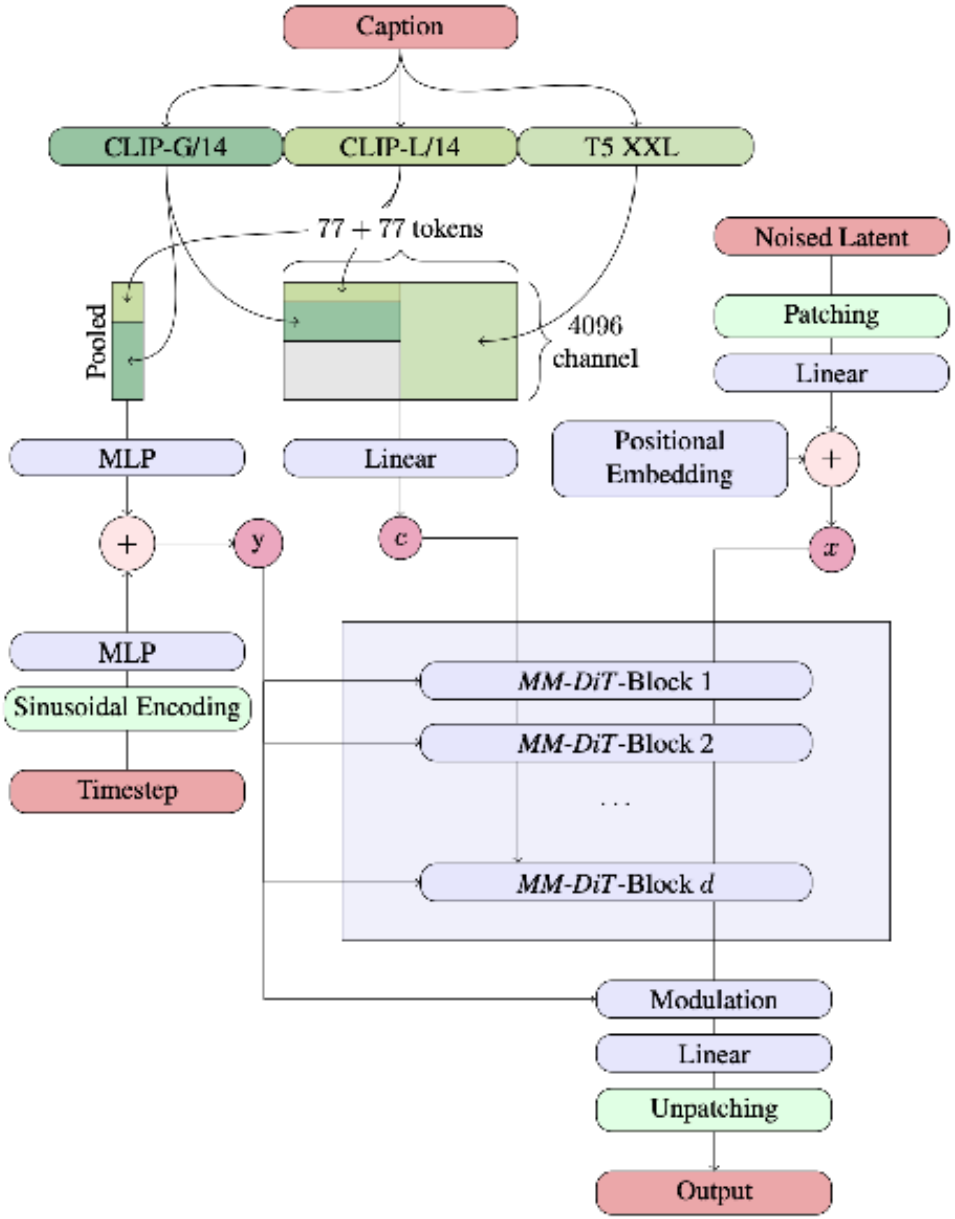
The Architecture: Noised Latent



$$x = W \cdot z_{patch} + b + P_{embed}$$

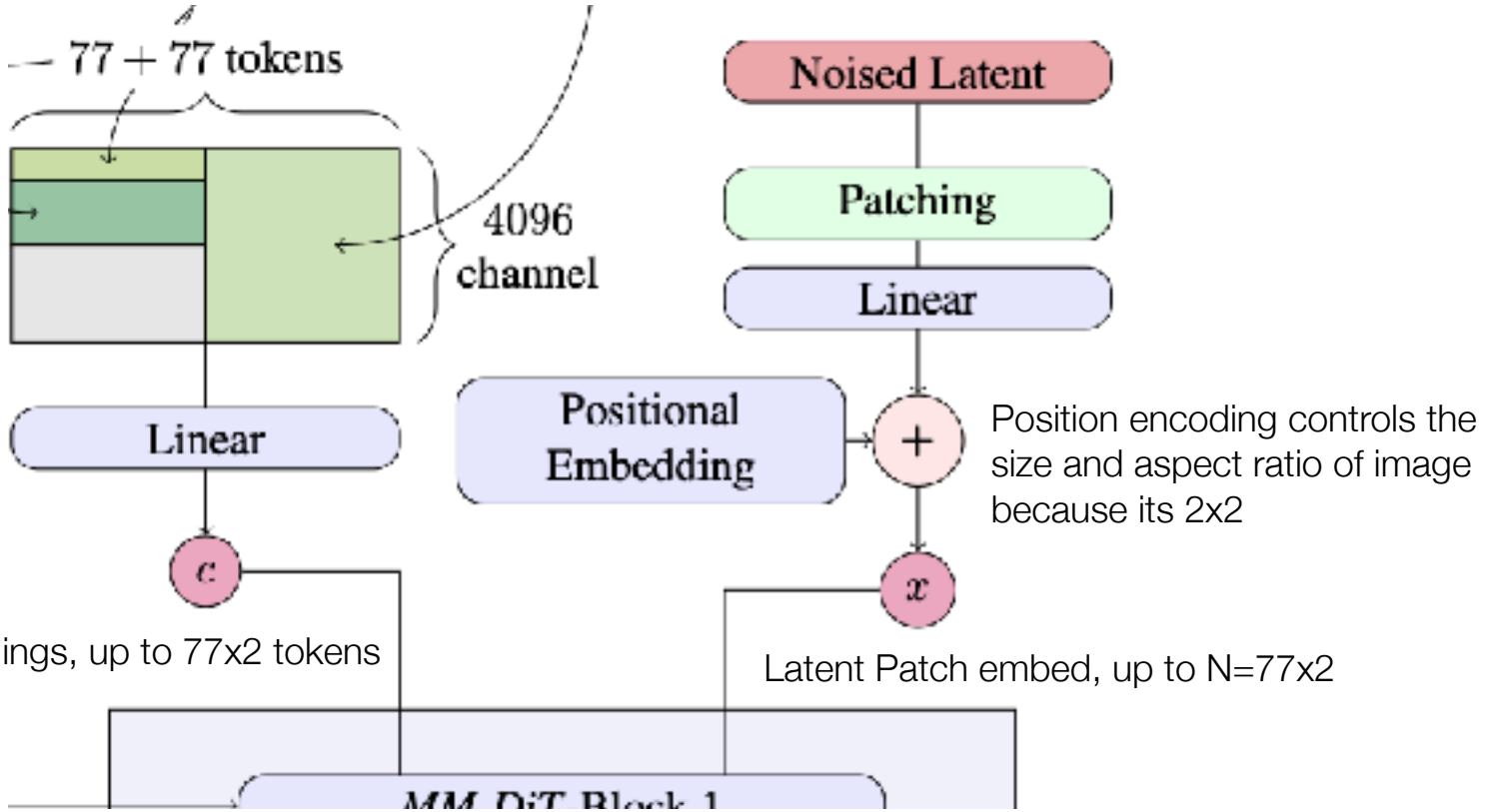


- $z_t = (1-t) \cdot x_0 + t \cdot \epsilon$
- This is a 3D latent tensor, Shown here as 2D, but there are channels in each patch with rich features
- Want to process this like a regular image in a ViT
- So we break into patches spatially
- Flatten each patch for use in X-former



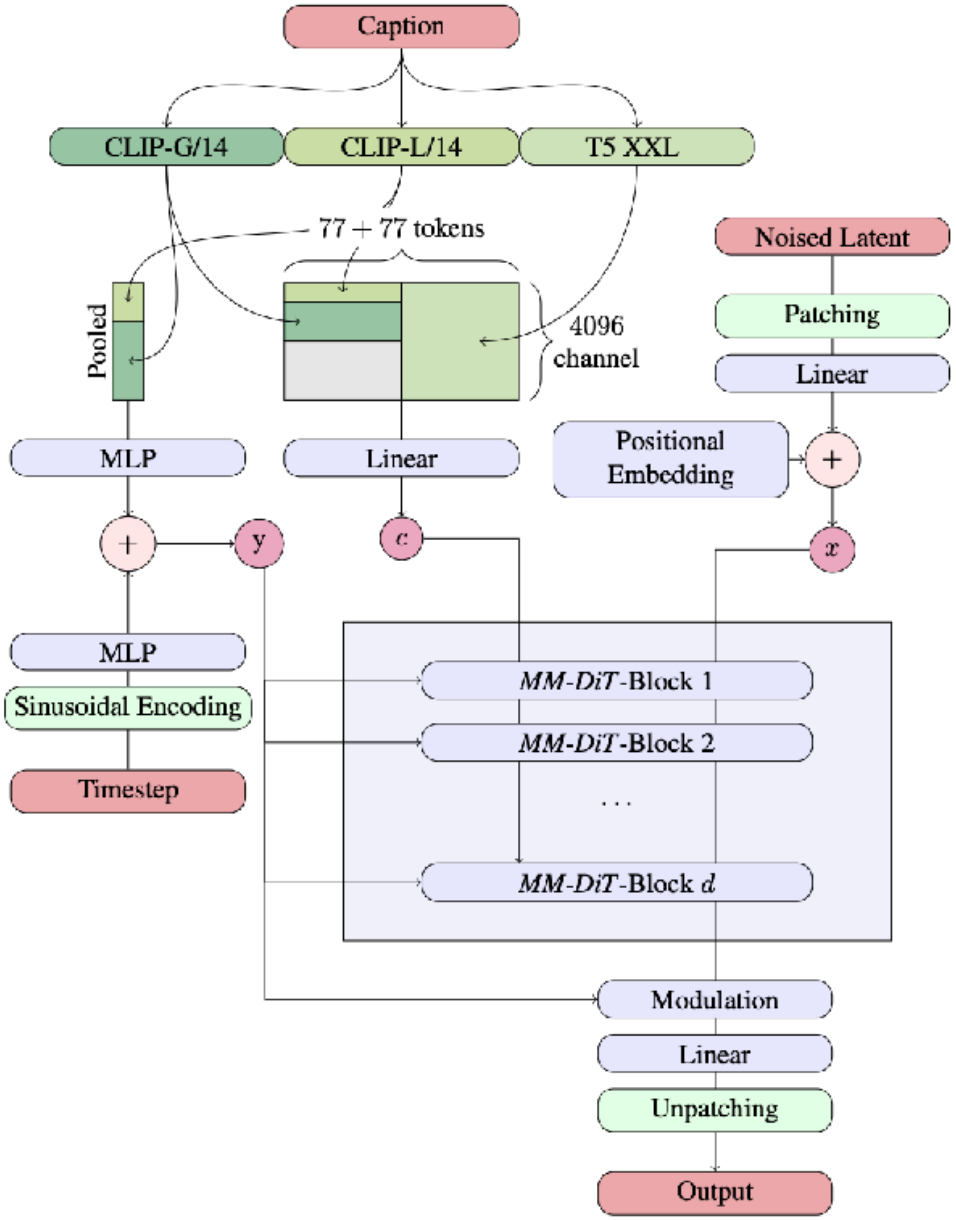
(a) Overview of all components.

The Architecture: Multi-modal inputs



Each “latent patch” is 2x2xc and represents an encoding of a portion of the latent image.

These patches can represent abstract concepts in the image latent space.



(a) Overview of all components.

Architecture: X-former block

Other Things:

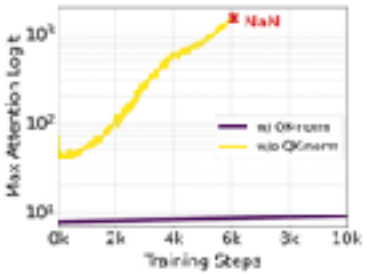
Do not use only human generated captions. Humans tend to not describe things like background, colors, etc.

Solution: Use trained models for generating captions of high quality. Then use a mix of "human captions" and "augmented captions" while training.

Make Encoder/Decoder large dimensionality.

Use as many x-formers as possible.

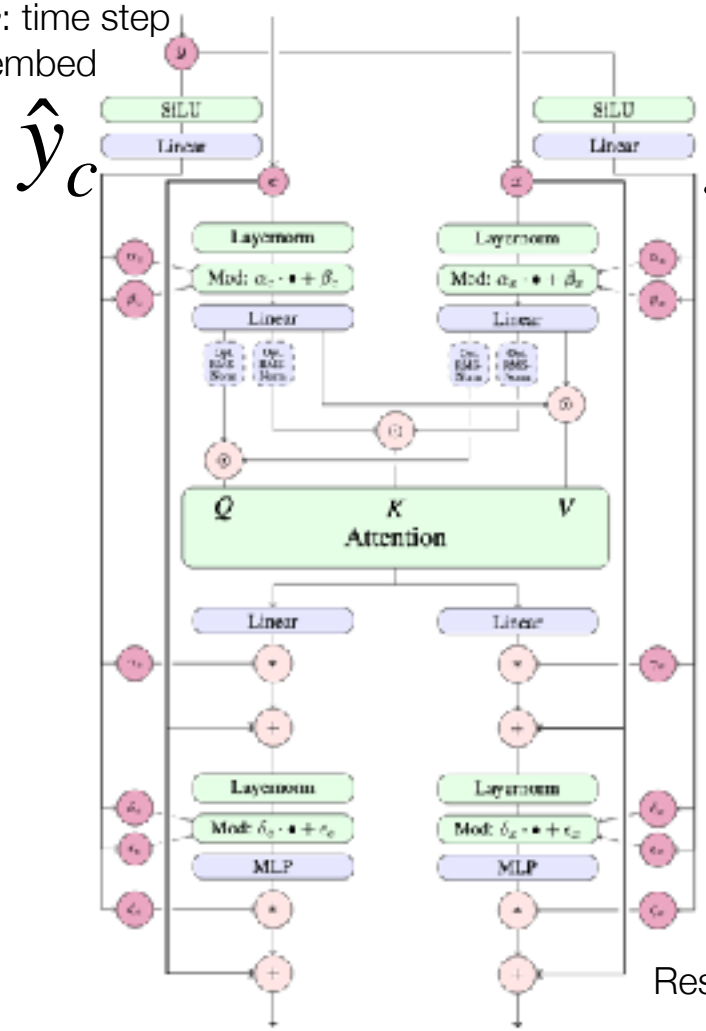
Normalize the QK attention matrix



c: text embed 77x2

x: latent image embed, 77x2

y: time step embed



$\hat{y}_c$

$\hat{y}_x$

Previous Layer

Shift the feature distribution using the time/text embed

$$= \alpha_x(\hat{y}) \cdot x_{patch}^{norm} + \beta_x(\hat{y})$$

Concatenate each modality and run in standard x-former (multi-headed)

Branch into two MLPs to separate out into two modalities

These feed into next layer

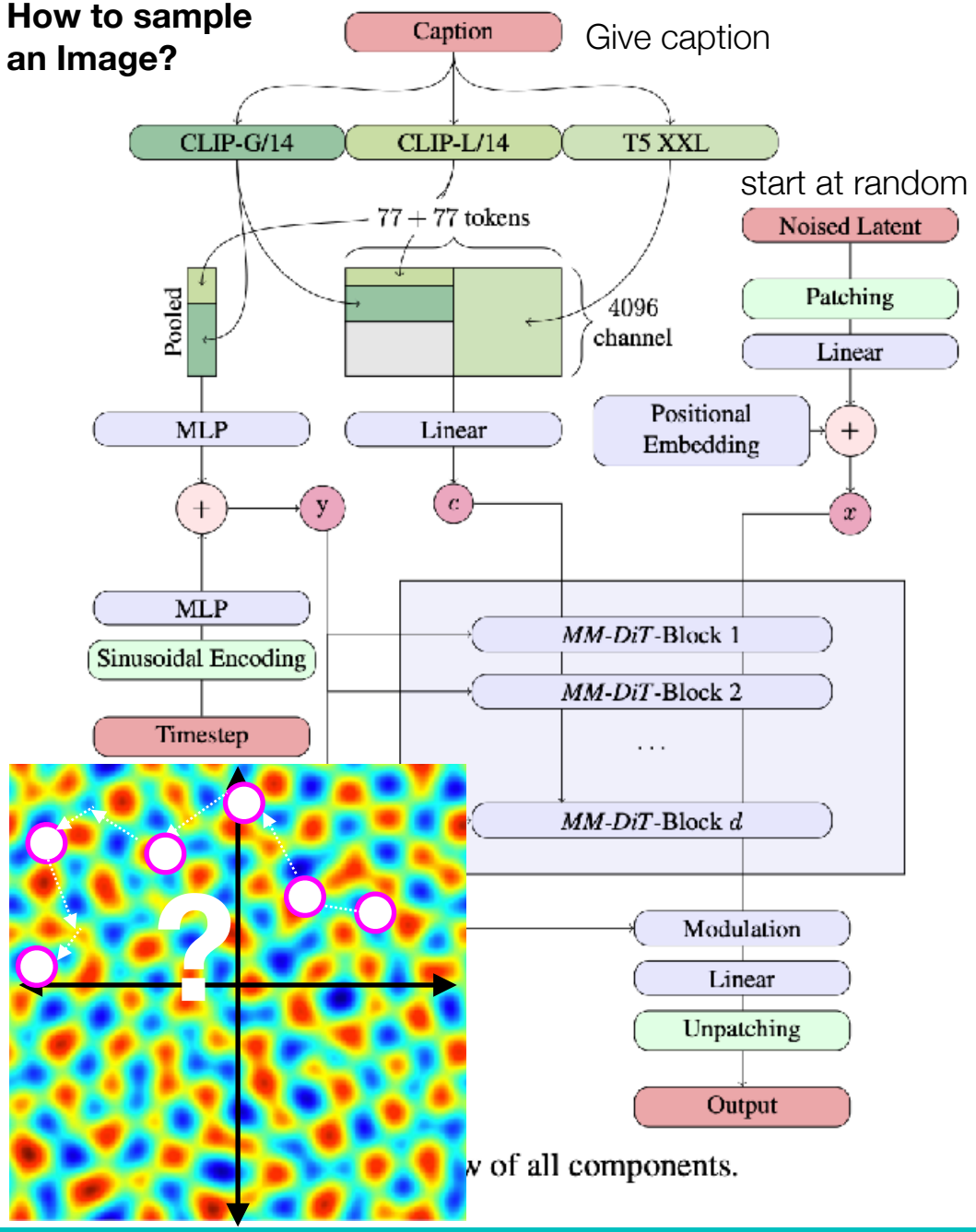
Residual Connections

maybe text features

maybe latent image features

Next Layer

How to sample an Image?



- We start with the base noise Probability, t=1 in $z_t = (1-t)\cdot x_0 + t\cdot \epsilon$
- The network has a *"t"* awareness branch for predicting noise. After the first step, how do we update *"t"*?
  - Also, should we update *"t"* for different resolutions?
- **Solution**: try different step sizes and see what people prefer…

Once trained, how to get images?

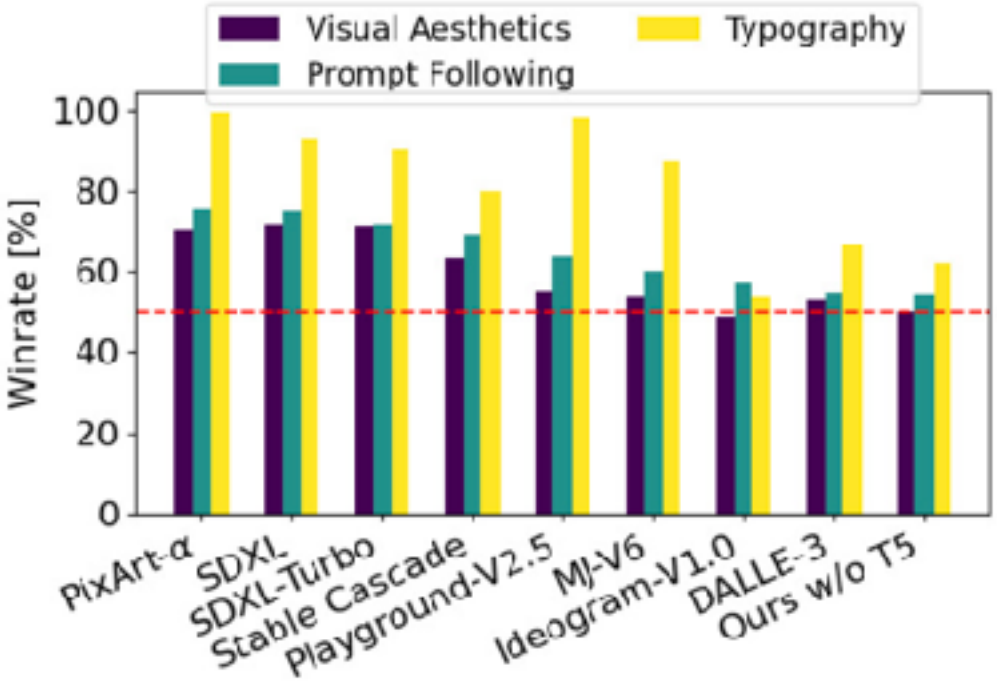
- **Solution**: try different step sizes and see what people prefer:

Least Preferred:

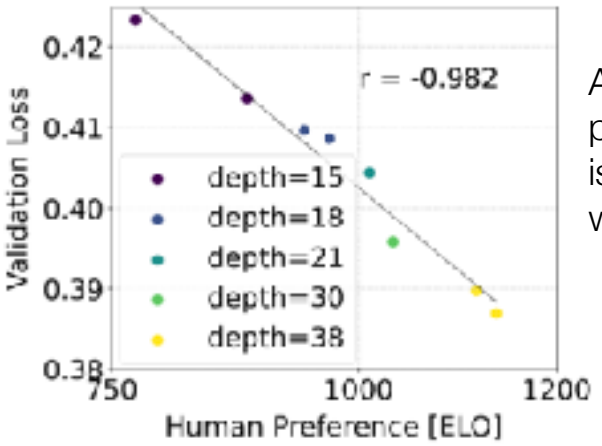
Most Preferred:

Evaluation: compare to state of the art

- Conducted large scale human subjects rating study
- Generate images from top models from same prompt
- Ask humans which version they prefer (aesthetics, prompt, text)
- What is the probability of their model winning?



Asked preference based on:
1. Simple Aesthetics (looks)
2. Following of the prompt
3. Accuracy of text and typography



Also: Human performance is correlated with loss!!

A whimsical and creative image depicting a hybrid creature that is a mix of a waffle and hippopotamus. This imaginative creature features the distinctive, bulkily body of a hippo, but with a texture and appearance resembling a golden-brown, crisply waffle. The creature might have elements like waffle squares across its skin and a syrup-like sheen. Its set in a surreal environment that playfully combines a natural water habitat of a hippo with elements of a breakfast table setting, possibly including oversized utensils or plates in the background.

All text-encoders	*w/o T5 (Raffel et al., 2019)*

"A burger patty, with the bottom bun and lettuce and tomatoes. "COFFEE" written on it in mustard"

"A monkey holding a sign reading "Scaling transformer models is awesome!"

"A mischievous ferret with a playful grin squeezes itself into a large glass jar, surrounded by colorful candy. The jar sits on a wooden table in a cozy kitchen, and warm sunlight filters through a nearby window"

Detailed pen and ink drawing of a happy pig butcher selling meat in its shop.

a massive alien space ship that is shaped like a pretzel

A kangaroo holding a beer, wearing ski goggles and passionately singing silly songs.

An entire universe inside a bottle sitting on the shelf at walmart on sale.

A cheesburger surfing the vibe wave at night

A swamp ogre with a pearl earring by Johannes Vermeer

A car made out of vegetables.

heat death of the universe, line art

Lecture Notes for

Neural Networks and Machine Learning

Stable Diffusion

Next Time:
Reinforcement Learning

Reading: None