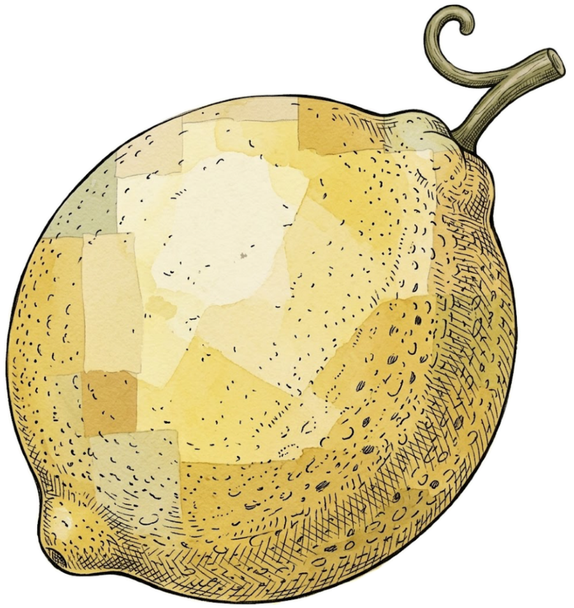
Squeezing the Juice From Imperfect Data Sources

By Adrián Rodríguez-Muñoz and the Daras Lab

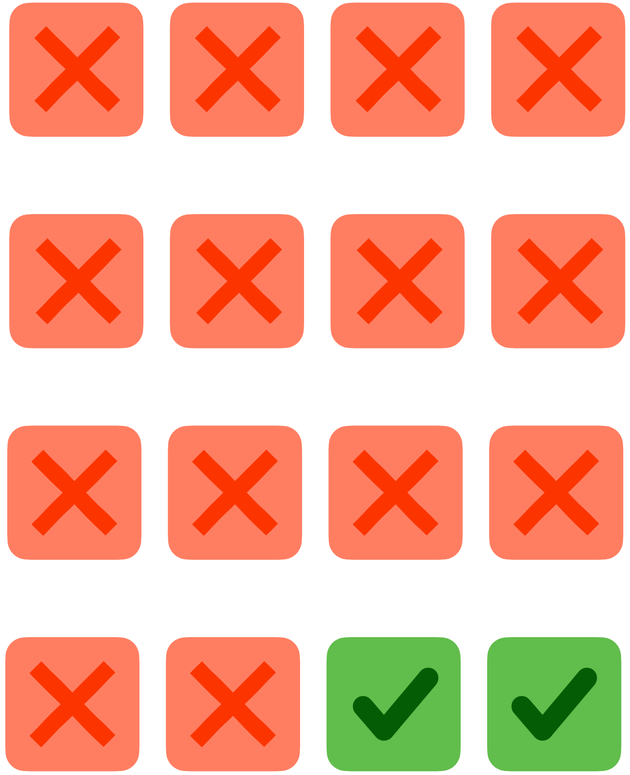


"When life gives you lemons, make lemonade”

Get value from every experience, not just successes



Lots of lemons...

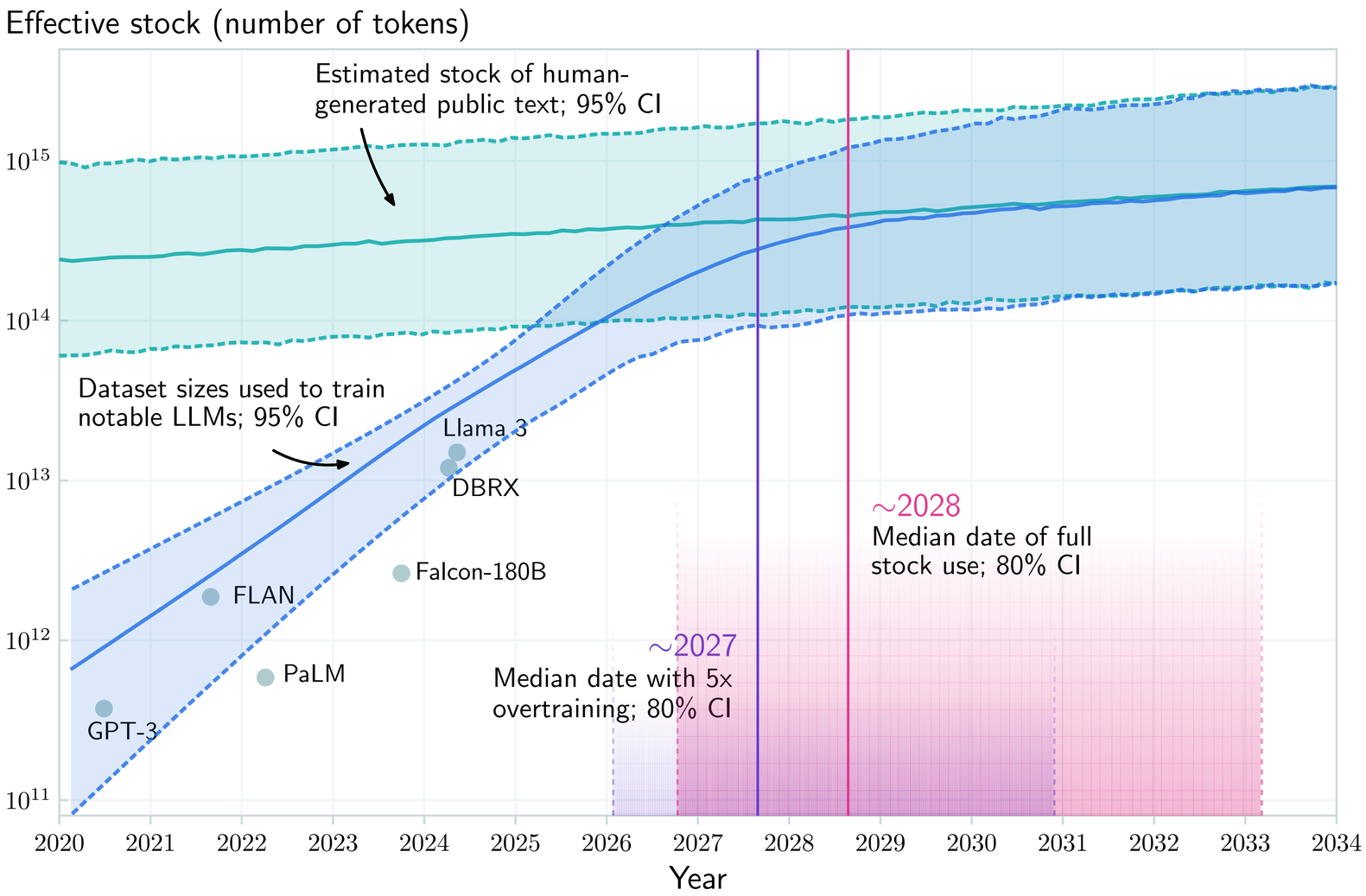


Successes vs failures



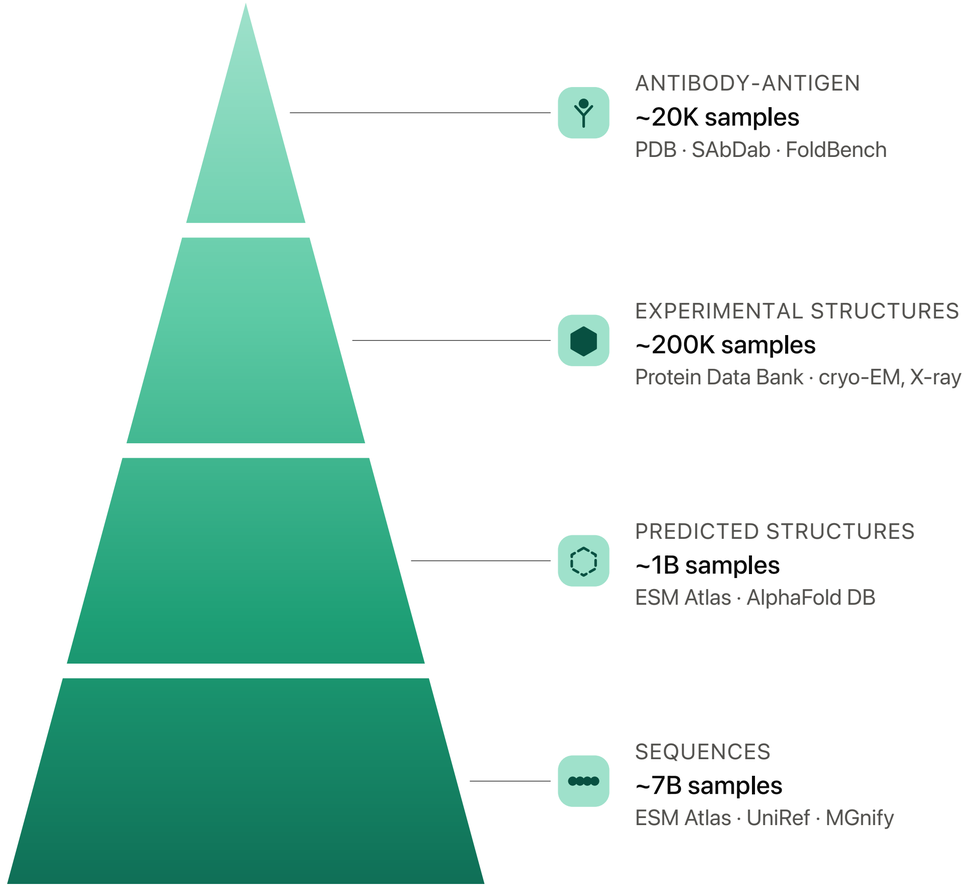
Good data vs bad data

Data scarcity in reality

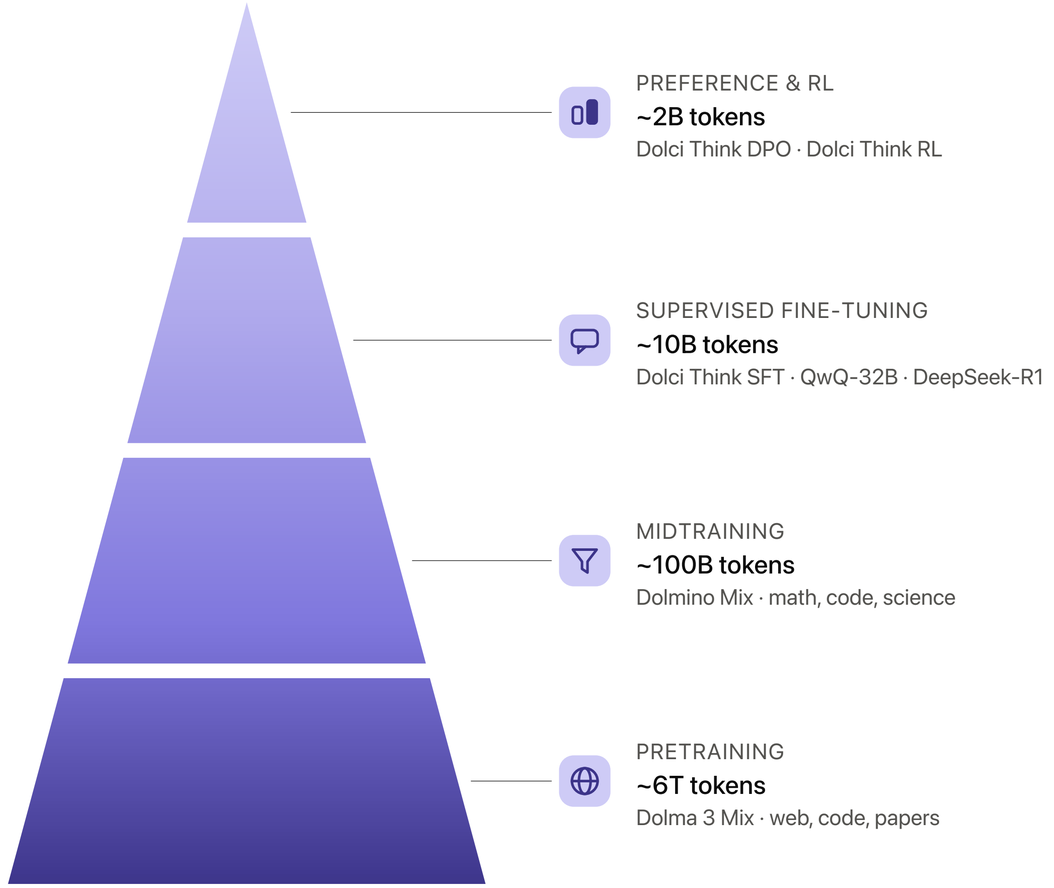


Projections of the effective stock of human-generated public text and dataset sizes used to train notable LLMs.
(Villalobos et al. 2024)

Data scarcity in reality



Quantities of Protein training data. Numbers from ESMFold2 (Biohub 2026)



Quantities of LLM training data. Numbers from Olmo 3 (Team Olmo 2026)

The Ambient Framework

"Garbage in, treasure out"

Some Bad Data In | **Corruption Aware Training** | **Good Data Out**



Mixed Quality Training Dataset
(**Arbitrary degradations** + **few clean samples**)



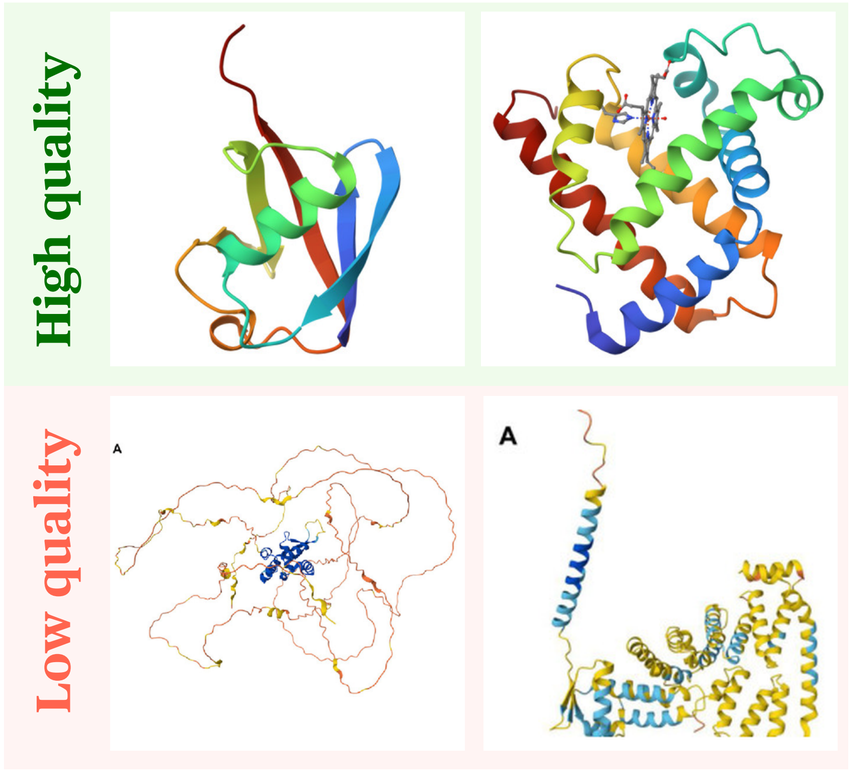
Text-to-image generation results
Prompt: "a cinematic dragon"

[1] **Daras*** et. al. Ambient Diffusion Omni: Training Good Models with Bad Data (NeurIPS 2025 **Spotlight, top 3.2%**) ⭐

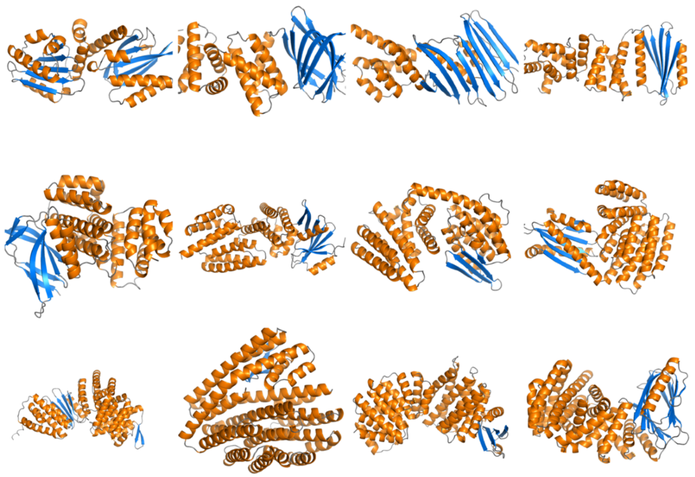
The Ambient Framework

"Garbage in, treasure out"

Some Bad Data In | **Corruption Aware Training** | **Good Data Out**



Mixed Quality Training Dataset
(**Low-pLDDT synthetics** + **few clean samples**)



High pLDDT generated proteins

[2] **Daras*** et. al. Ambient Proteins: Training Diffusion Models on Low Quality Structures (NeurIPS 2025 **Spotlight, top 3.2%**) ⭐

The Ambient Framework

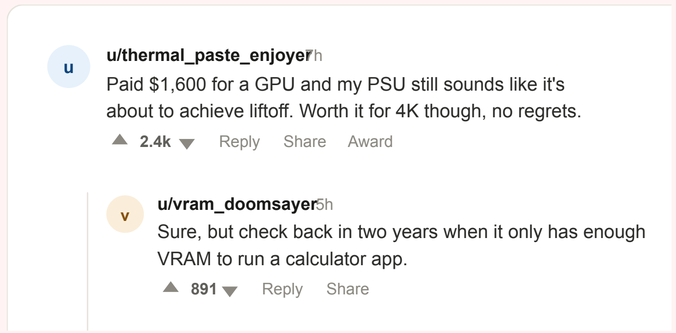
“Garbage in, treasure out”

Some Bad Data In

High quality

We have some bad news for ATI fans, AMD has announced that it will be killing the ATI brand in favor of its Fusion project which aims to build hybrid CPU and GPU chips. The Radeon and FirePro branding will be remaining intact, but ATI Eyefinity will now be known as AMD Eyefinity.

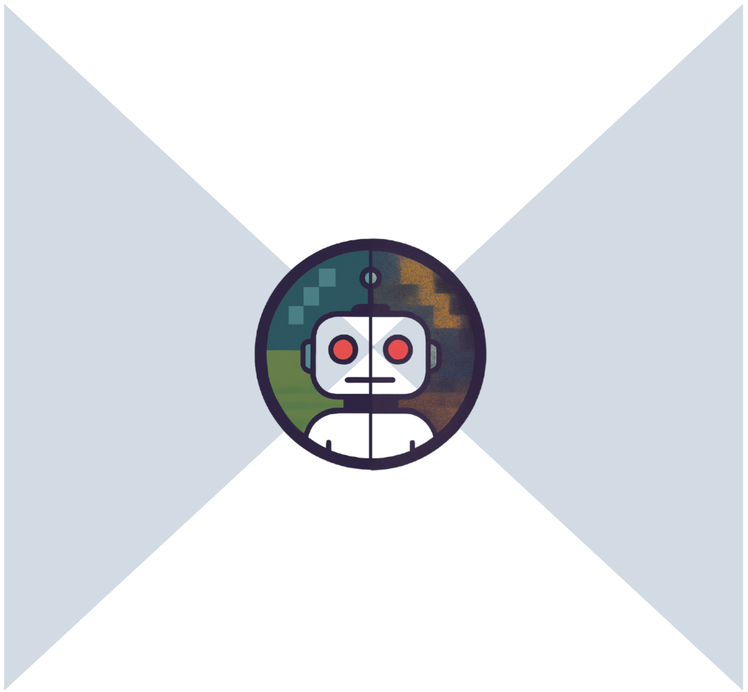
Low quality



Mixed Quality Training Dataset

(**Low quality sources** + **few clean samples**)

Corruption Aware Training



Good Data Out

This is the first structure I should discuss. The structure should be:

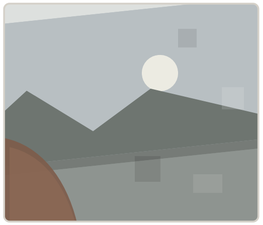
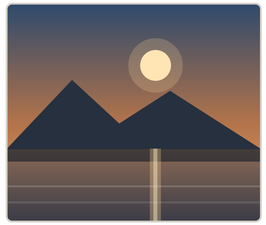
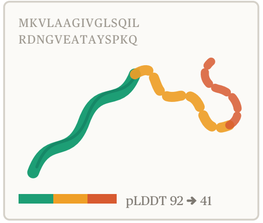
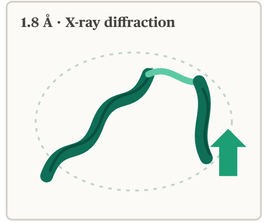
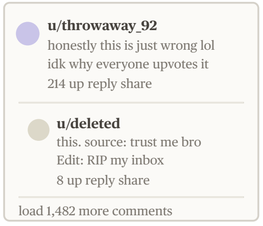
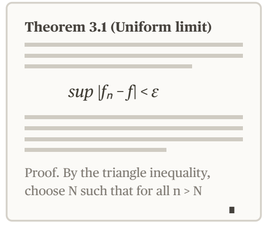
1. First - the software.

2. Second - the database, which is usually used to store the authority of the code, as well as encapsulating the structure.

3. Third - the interface, which is usually used to interact with the database to get messages.

Diverse clean outputs

Good data and bad data summary

MODALITY	LOW QUALITY BUT ABUNDANT		HIGH QUALITY BUT SCARCE		RATIO
Images		**~2 trillion / year** Social uploads and camera rolls Extractable video frames Web image-text pairs (LAION-5B) ~5.85B Scanned document corpora		**~1 billion images** Licensed stock libraries ~1B Human-annotated image sets Press and editorial archives Museum and fine-art scans	**2,000 : 1**
Proteins		**~2.5 billion samples** Metagenomic sequences (MGnify) ~1.66B Predicted structures (AlphaFold, ESM) ~800M		**~260 thousand structures** Experimental structures (PDB) ~258K Multimeric complexes Antibody-antigen complexes (SAbDab)	**10,000 : 1**
Text		**~20 trillion tokens** General web crawl ~18.5T Public code (The Stack v2) ~900B Forums and comment threads		**~100 billion tokens** Peer-reviewed full text (S2ORC) ~60B Encyclopedic reference Textbooks and monographs Proofs	**200 : 1**

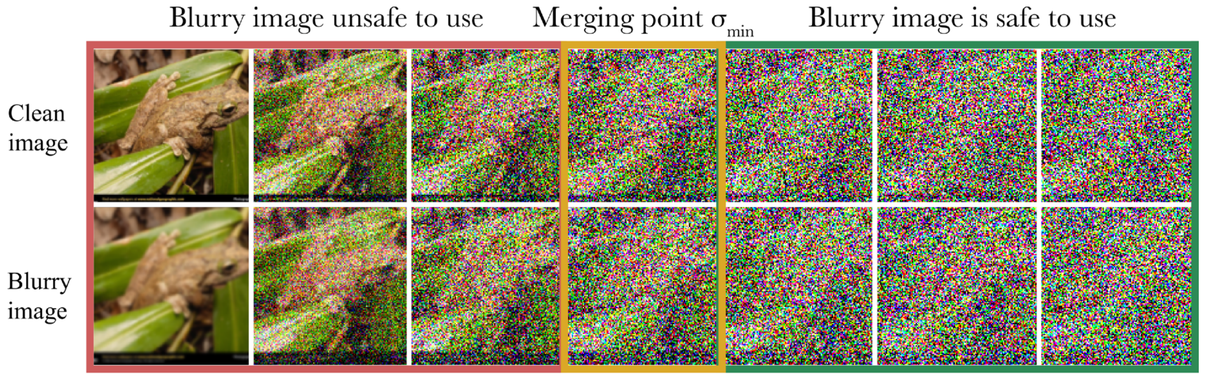
Research tree for learning from imperfect data



Ambient Omni

Foundational technique for arbitrary corruptions

- **Setting:** Filtering pipelines are extremely wasteful, contrasting the need for large amounts of data



- **Key idea:** Noise makes distributions more similar



- **Algorithm:** Train on *noisy* imperfect data

[1] **Rodriguez-Munoz and Daras** et. al. Ambient Diffusion Omni: Training Good Models with Bad Data (NeurIPS 2025 **Spotlight, top 3.2%**) ⭐

Bad Images vs. Good Images at Levels of Noise



Great! Good! Sucks!



Great images! Great noisy images! Great super noisy images!

[1] **Rodriguez-Munoz and Daras** et. al. Ambient Diffusion Omni: Training Good Models with Bad Data (NeurIPS 2025 **Spotlight, top 3.2%**) ⭐

Loss Function for Learning with Clean Data

Learning from high-quality data

$$J(\theta) = \mathbb{E}_{t \in U[0,T]} \mathbb{E}_{x_0, x_t|t} ||h_\theta(x_t, t) - x_0||^2$$

For all images in our dataset, and all diffusion times t

The network learns to predict $\mathbb{E}[x_0 \mid x_t]$

...but what if we don't have x_0?

$$\mathbb{E}[x_s \mid x_t] = \frac{\sigma_s^2}{\sigma_t^2} x_t + \left(1 - \frac{\sigma_s^2}{\sigma_t^2}\right) \mathbb{E}[x_0 \mid x_t]$$

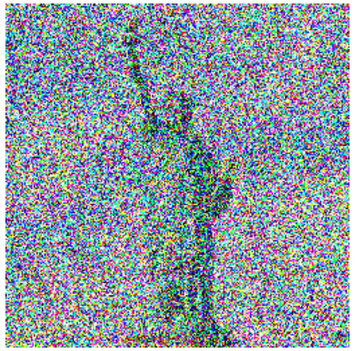
Expected value at lower time s given noisy observation at time t

Noisy observation at time t

Expected value at time 0 given noisy observation t



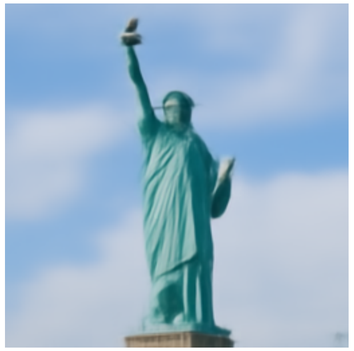
x_0



x_t



x_s



$\mathbb{E}[x_0 \mid x_t]$



$\mathbb{E}[x_s \mid x_t]$

[1] **Rodriguez-Munoz and Daras** et. al. Ambient Diffusion Omni: Training Good Models with Bad Data (NeurIPS 2025 **Spotlight, top 3.2%**) ⭐

Loss Function for Learning with Corrupted Data

Learning from low-quality data

$$J_{\text{ambient-o}}(\theta) = \mathbb{E}_{t\in\mathcal{U}[0,T]} \sum_{i:t_i^{\min}<t} \mathbb{E}_{x_t|x_{t_i^{\min}}^{(i)}} \left[\left|\left| \alpha(t, t_i^{\min}) h_\theta(x_t, t) + (1 - \alpha(t, t_i^{\min}))x_t - x_{t_i^{\min}}^{(i)} \right|\right|^2 \right]$$

Train only on low-quality images indistinguishable from high quality images at diffusion time t

The network predicts $\mathbb{E}[x_0 \,|\, x_t]$, and we transform it into an unbiased estimate of $E[x_{t_i^{\min}} \,|\, x_t]$

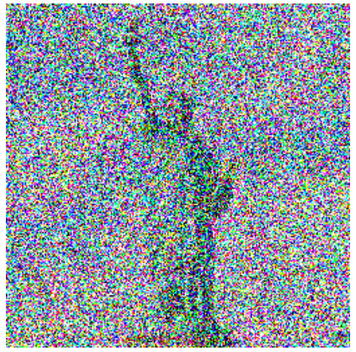
The target is the noisy image, which prevents leakage of the low quality distribution



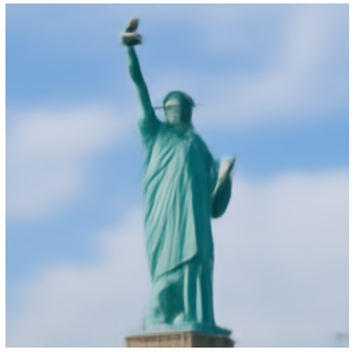
x_0



x_t



$x_{t_i^{\min}}$



$\mathbb{E}[x_0 \,|\, x_t]$



$\mathbb{E}[x_{t_i^{\min}} \,|\, x_t]$

[1] **Rodriguez-Munoz and Daras** et. al. Ambient Diffusion Omni: Training Good Models with Bad Data (NeurIPS 2025 **Spotlight, top 3.2%**) ⭐

ImageNet Results

Baseline

Ambient-o



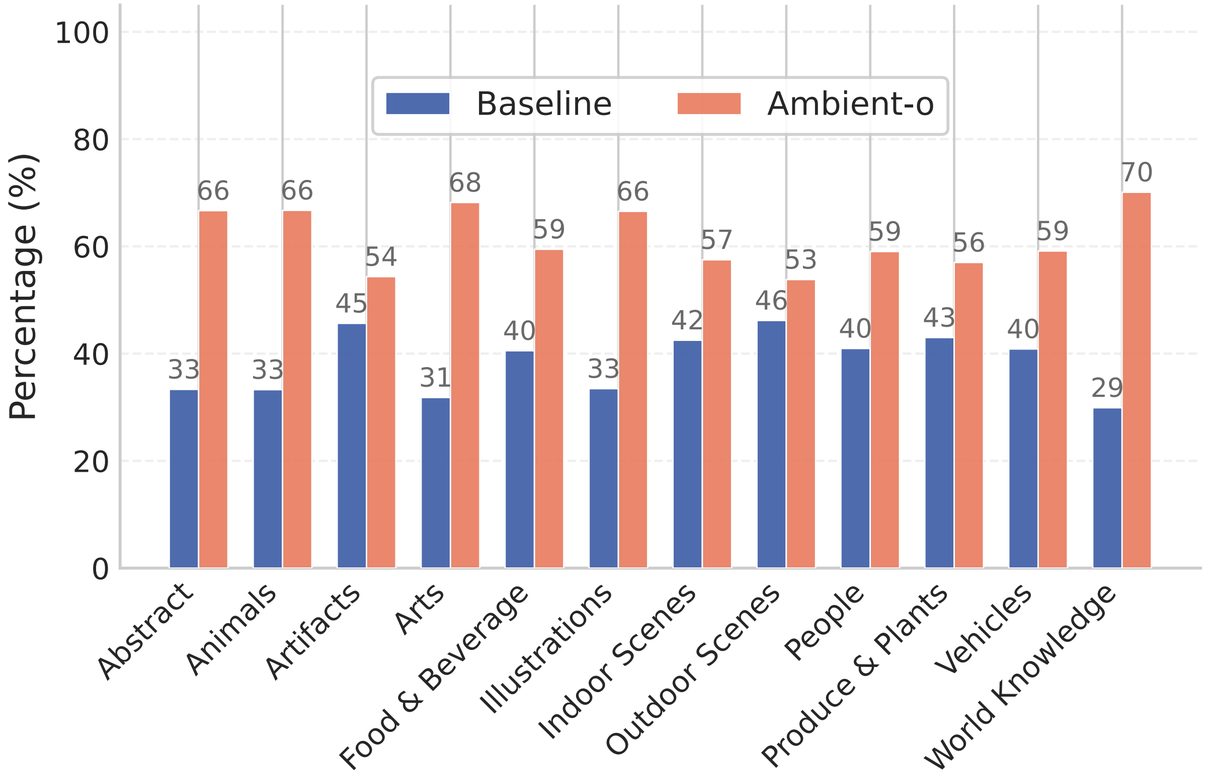
[1] **Rodriguez-Munoz and Daras** et. al. Ambient Diffusion Omni: Training Good Models with Bad Data (NeurIPS 2025 **Spotlight, top 3.2%**) ⭐

Text-to-Image Results

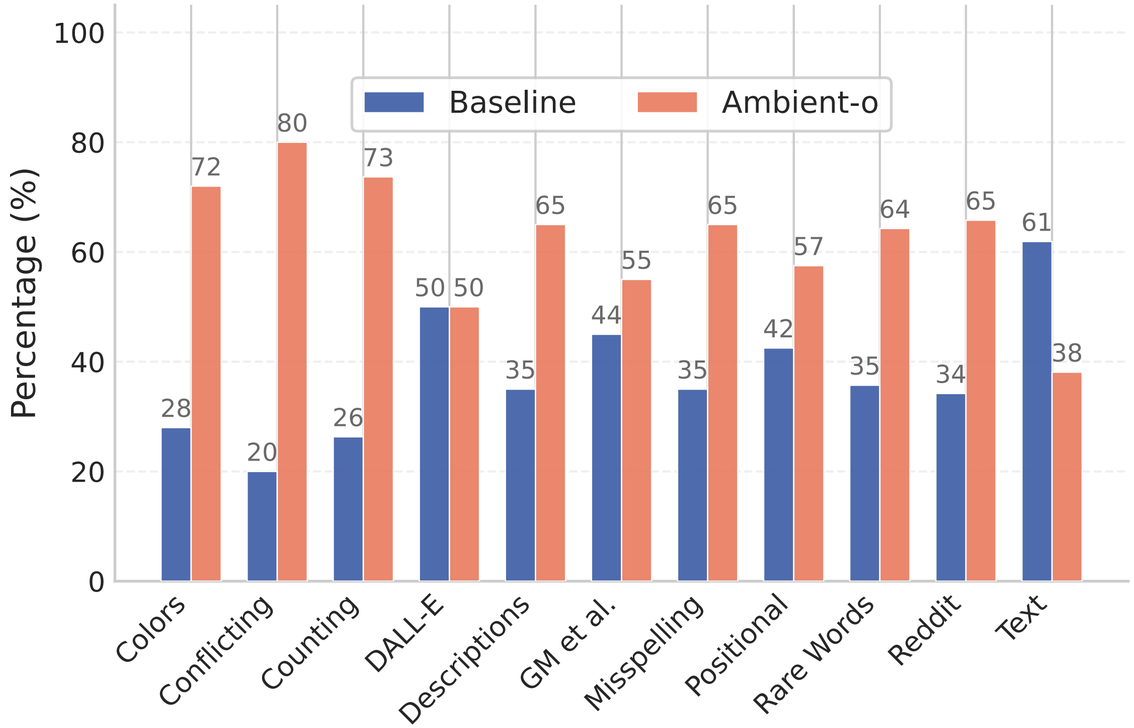


[1] **Rodriguez-Munoz and Daras** et. al. Ambient Diffusion Omni: Training Good Models with Bad Data (NeurIPS 2025 **Spotlight, top 3.2%**) ⭐

Text-to-Image Results: GPT-4o Evals



(A) DrawBench (Sararia et al.)



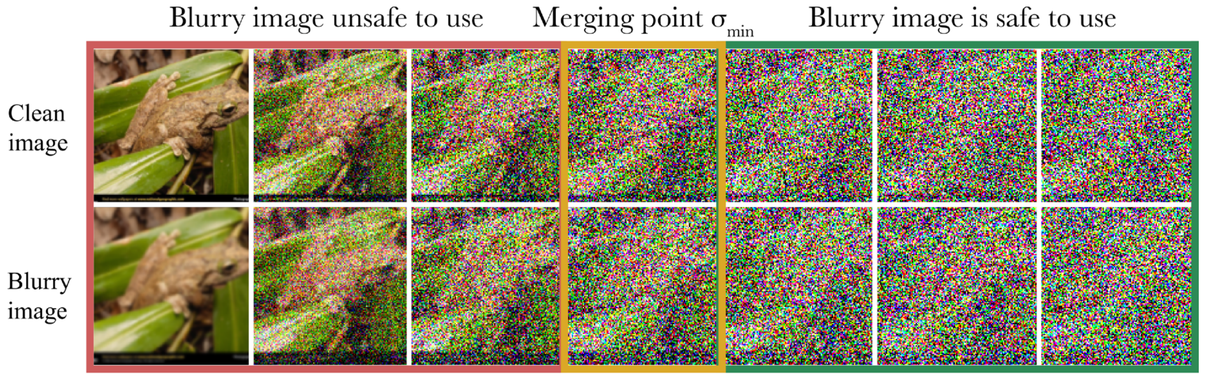
(B) PartiPrompts (Yu et al.)

[1] **Rodriguez-Munoz and Daras** et. al. Ambient Diffusion Omni: Training Good Models with Bad Data (NeurIPS 2025 **Spotlight, top 3.2%**) ⭐

Ambient Omni

Foundational technique for arbitrary corruptions

- **Setting:** Filtering pipelines are extremely wasteful, contrasting the need for large amounts of data



- **Key idea:** Noise makes distributions more similar



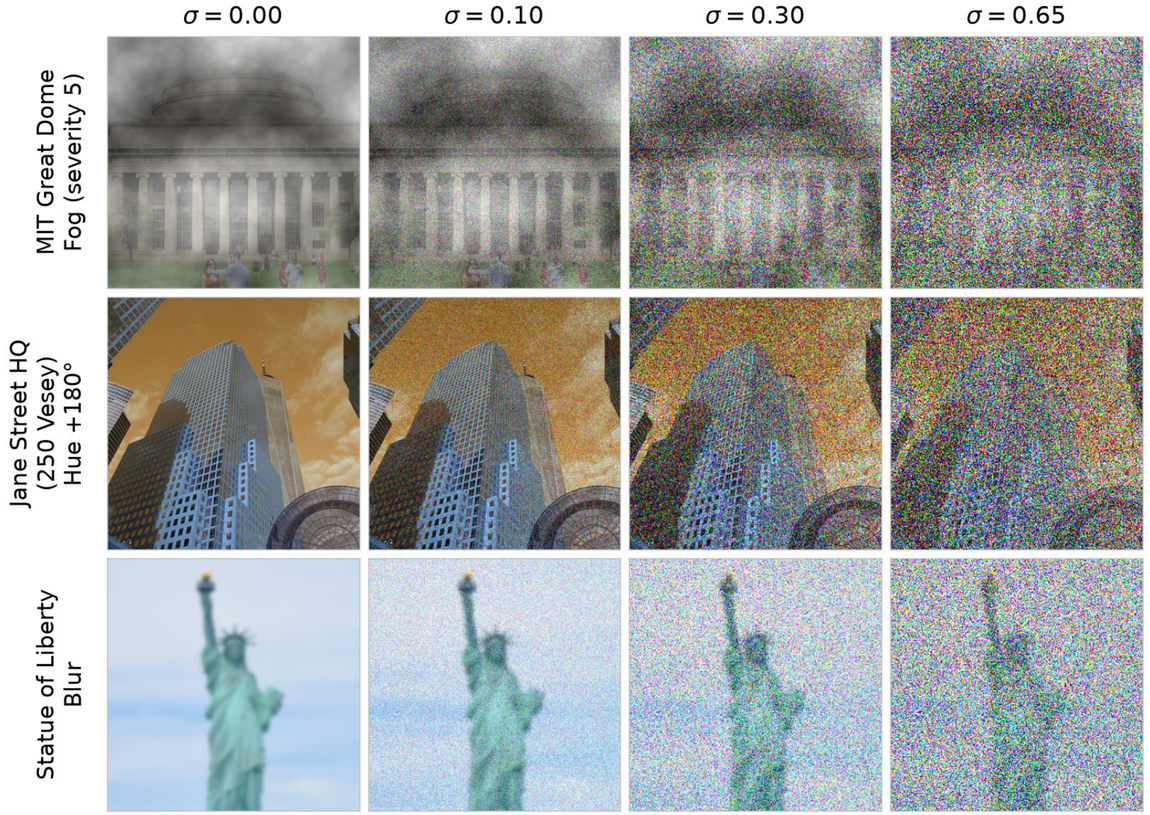
- **Algorithm:** Train on *noisy* imperfect data

[1] **Rodriguez-Munoz and Daras** et. al. Ambient Diffusion Omni: Training Good Models with Bad Data (NeurIPS 2025 **Spotlight, top 3.2%**) ⭐

Self-Cleaning Diffusion

Learning more from arbitrary corruptions

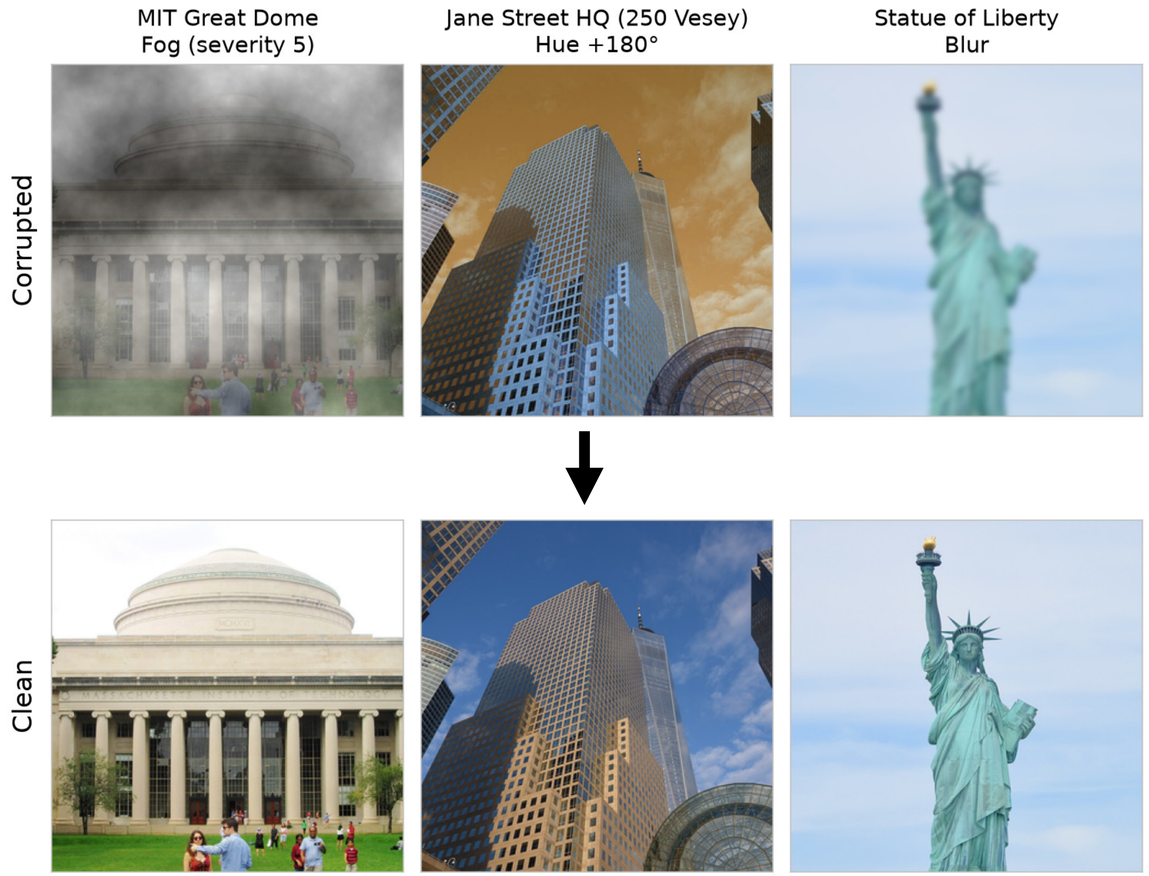
- **Setting:** Ambient Omni fails for low-frequency corruptions
- **Key idea:** Attempt a transport map
- **Algorithm:** Transport + Ambient Omni



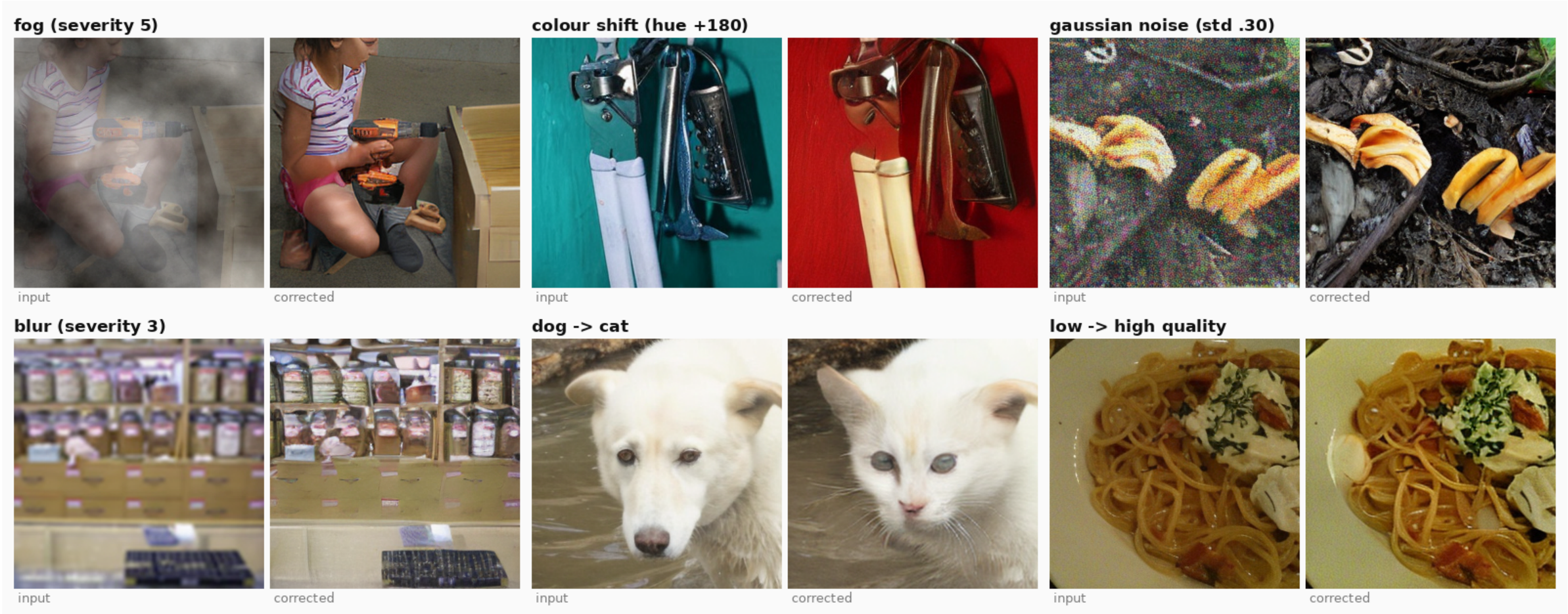
Self-Cleaning Diffusion

Learning more from arbitrary corruptions

- **Setting:** Ambient Omni fails for low-frequency corruptions
- **Key idea:** Attempt a transport map
- **Algorithm:** Transport + Ambient Omni



Results with Drifting-Based Transport Map



Results for Training Diffusion Models

Table 3: Performance comparison for the fog corruption (90%) setting.

Method	σ_{min}	FID ↓	Val FID ↓	DINOv2 FD ↓	DINOv2 FD Val ↓	Memorization (sim > 0.8) ↓	Vendi Score ↑
Filtering	∞	72.48 ± 0.25	70.08 ± 0.27	742.74 ± 1.91	748.84 ± 2.02	0.2362	212.99
HQ Finetuning	∞	18.73 ± 0.05	17.37 ± 0.04	330.70 ± 1.02	341.52 ± 1.02	0.1286	544.90
Conditional	0.000	15.38 ± 0.05	13.37 ± 0.04	353.21 ± 1.08	361.50 ± 1.17	**0.1020**	512.14
Ambient Omni	→ ∞	72.48 ± 0.25	70.08 ± 0.27	742.74 ± 1.91	748.84 ± 2.02	0.2362	212.99
Self-Cleaning Diffusion (Ours)	0.400	**10.19 ± 0.07**	**9.39 ± 0.07**	**288.15 ± 0.83**	**299.12 ± 0.88**	0.1684	**546.80**
All clean	0.000	10.13 ± 0.06	9.73 ± 0.06	219.43 ± 0.58	231.49 ± 0.57	0.2196	608.39

Filtering / Ambient Omni (high noise)



Ambient Omni (middle/low noise)



Ours



Conditional



Transport (no noise)



Clean (Oracle)



Results for Training Diffusion Models

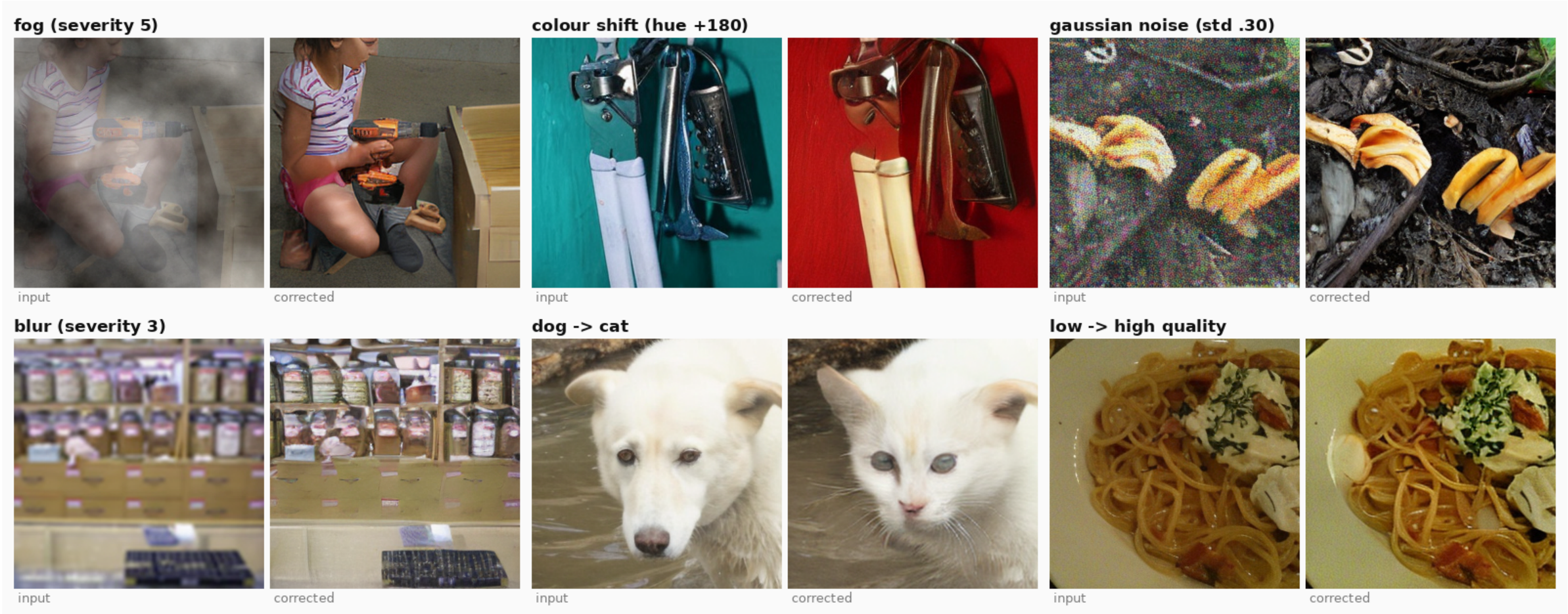
Table 4: Performance comparison for the dogs and cats out-of-distribution learning experiment.

Method	σ_{min}	Class Alignment ↑	Aes. Score ↑	Vendi Score ↑
Filtering	∞	**0.9992**	4.89	34.72
Conditional	0.000	0.9871	**4.91**	39.16
Self-Cleaning Diffusion (Ours)	0.000	0.9953	4.81	41.07
	0.094	0.9912	4.80	**41.48**
	0.129	0.9925	4.77	40.65
	0.186	0.9895	4.73	39.20

Table 5: Performance comparison for the AestheticImageNet learning experiment.

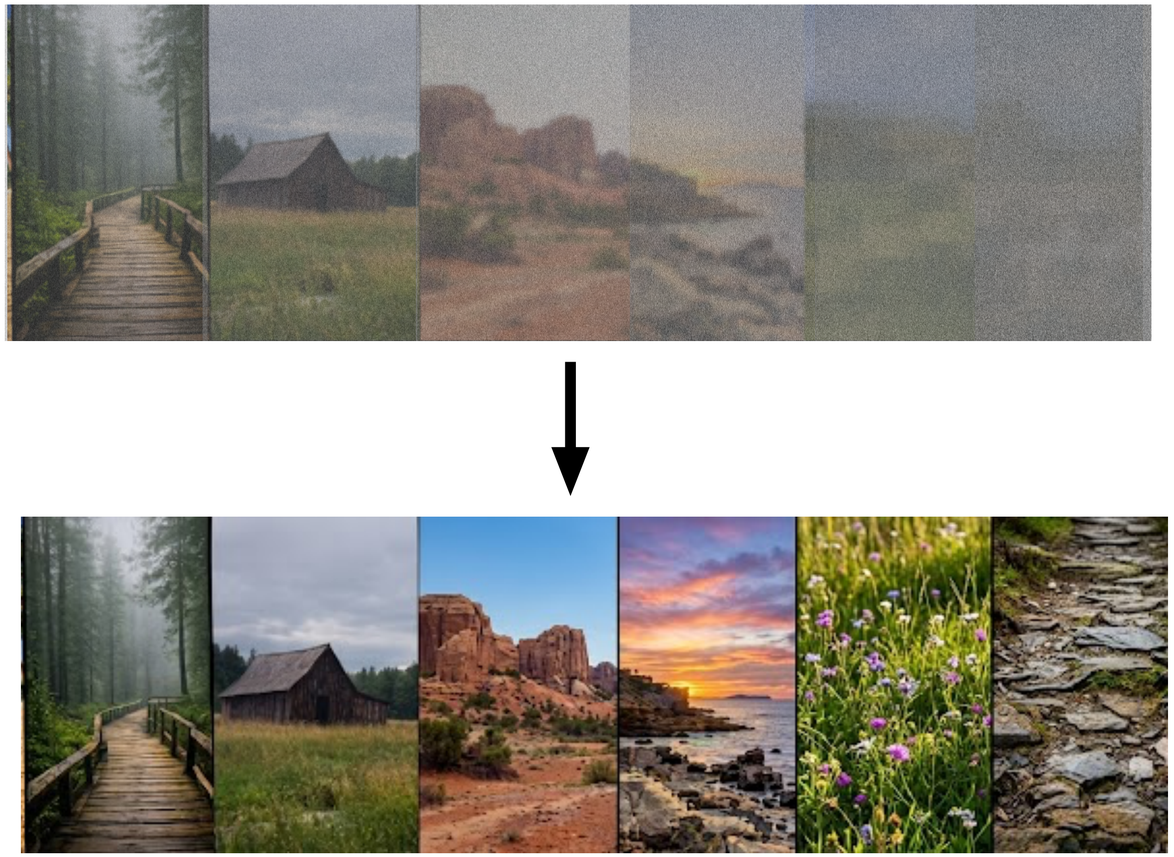
Method	σ_{min}	DINO FD ↓	Aes. Score ↑	Aes. Score ≥ 5.0 ↑	Memorization ↓	Vendi Score ↑
Baseline	0.000	435	4.3443	8.39%	0.0728	**460.68**
Omni	0.186	478	4.4195	11.04%	**0.0624**	427.63
Self-Cleaning Diffusion (Ours)	0.186	**433**	**4.6411**	**20.86%**	0.1032	442.90

Results with Drifting-Based Transport Map



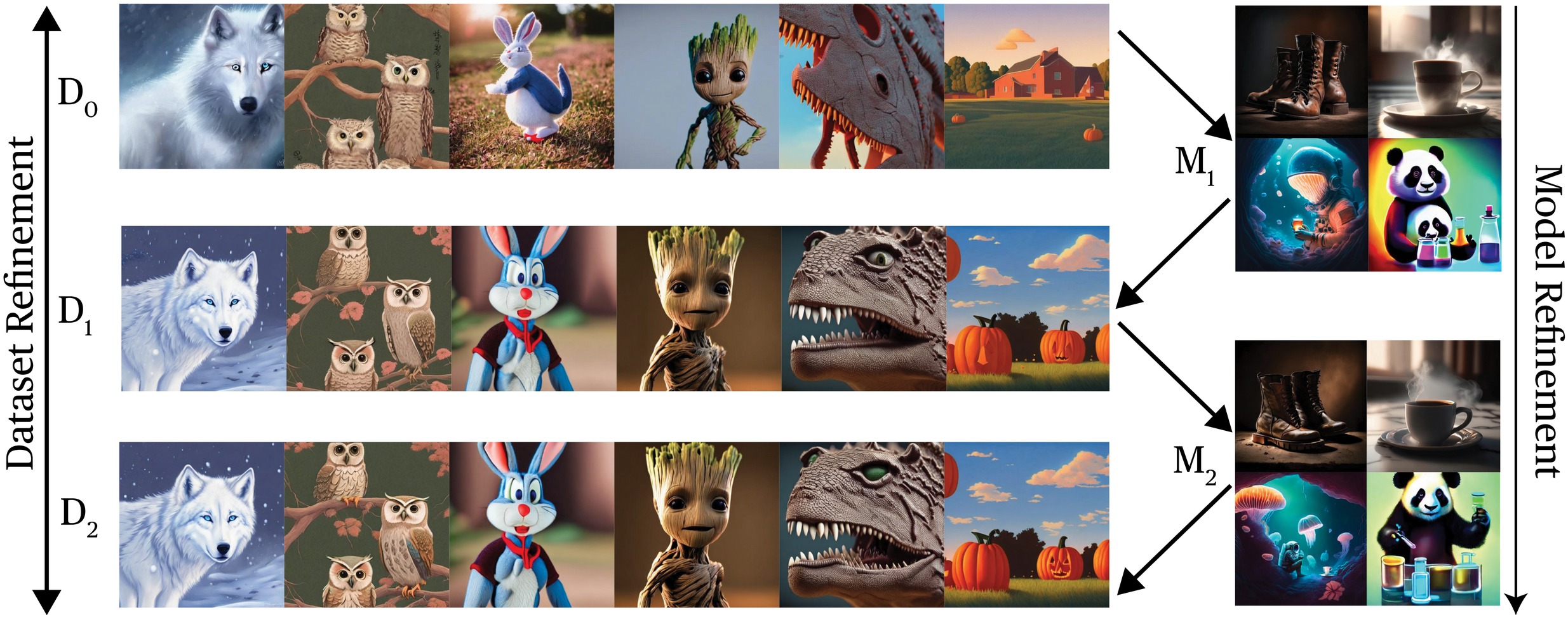
Ambient Dataloops: Refining bad data

- **Setting:** We can learn from imperfect data, but we (needlessly) give up after that
- **Key idea:** Attempt to fix the imperfections
- **Algorithm:** Iteratively refine models and data together



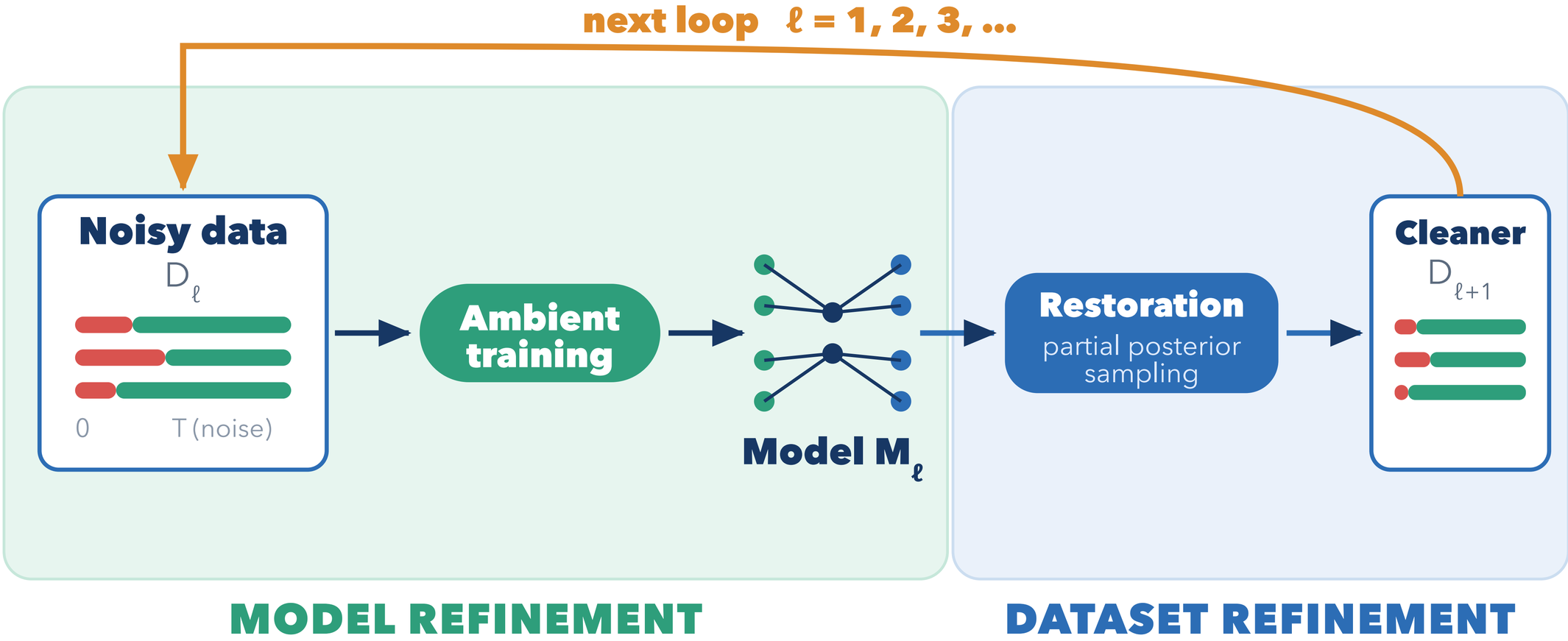
[3] **Rodriguez-Munoz*** et. al. Ambient Dataloops: Generative Models for Dataset Refinement: (ICML 2026)

Qualitative Results



[3] **Rodriguez-Munoz*** et. al. Ambient Dataloops: Generative Models for Dataset Refinement: (ICML 2026)

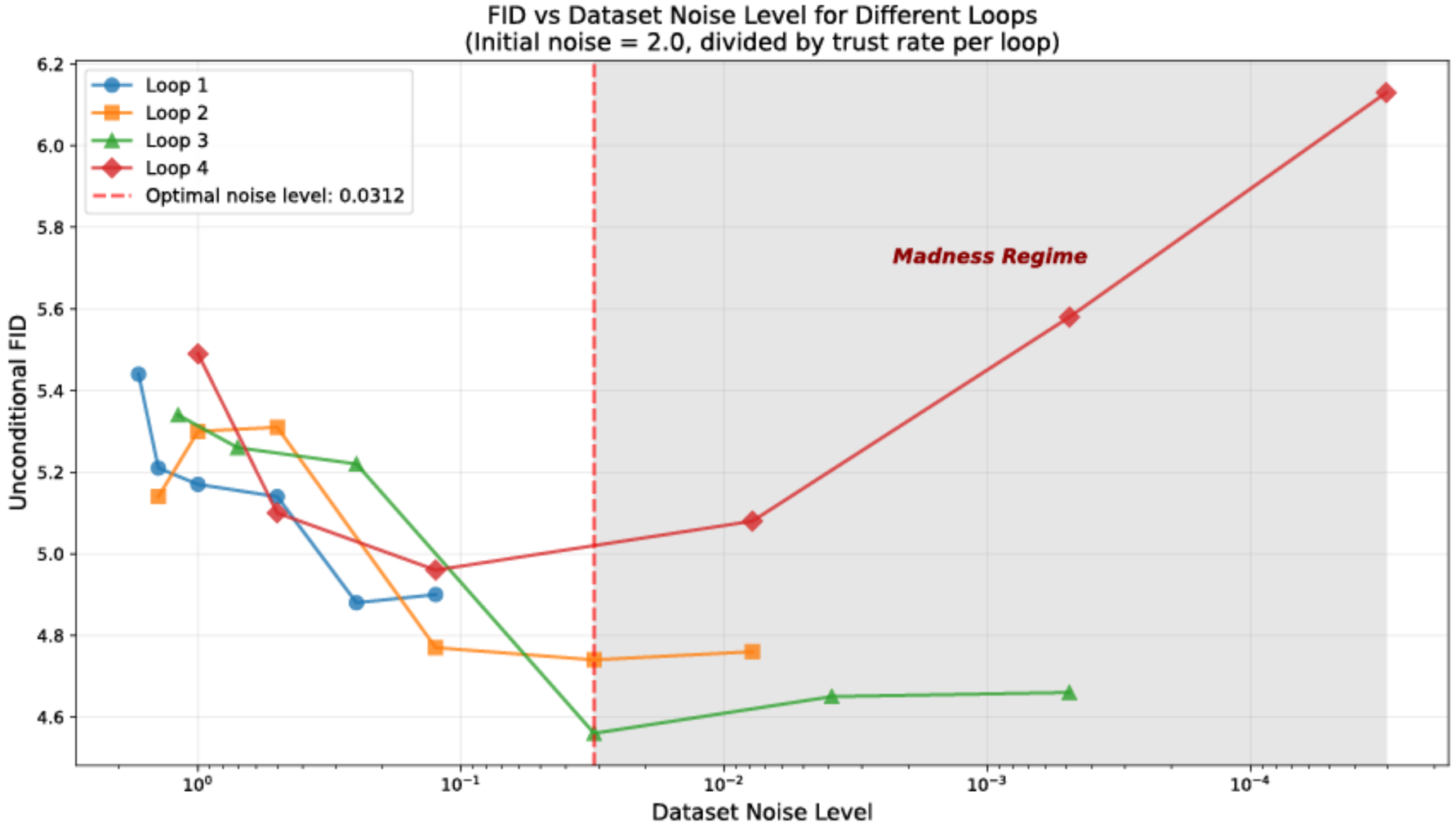
The Titular Data Loop



The imperfect data starts at a certain level of added noise, proportional to its un-trustworthyness.
At each data loop, the level of noise is reduced via posterior sampling

[3] **Rodriguez-Munoz*** et. al. Ambient Dataloops: Generative Models for Dataset Refinement: (ICML 2026)

The Dangers of Over-Confidence



[3] **Rodriguez-Munoz*** et. al. Ambient Dataloops: Generative Models for Dataset Refinement: (ICML 2026)

Quantitative Results

Table 1: Quantitative benefits of Ambient Loops on COCO zero-shot generation and GenEval.

	COCO (Fidelity & Alignment)		GenEval Benchmark						
Method	**FID-30K ↓**	**Clip-FD-30K ↓**	**Overall**	**Single**	**Two**	**Counting**	**Colors**	**Position**	**Color attribution**
Micro-diffusion	12.37	10.07	0.44	0.97	0.33	0.35	0.82	0.06	0.14
Ambient-o (L0)	10.61	9.40	0.47	0.97	**0.40**	**0.36**	0.82	0.11	0.14
Ambient Loops (L1)	**10.06**	**8.83**	**0.47**	**0.97**	0.38	0.35	0.78	**0.11**	**0.19**

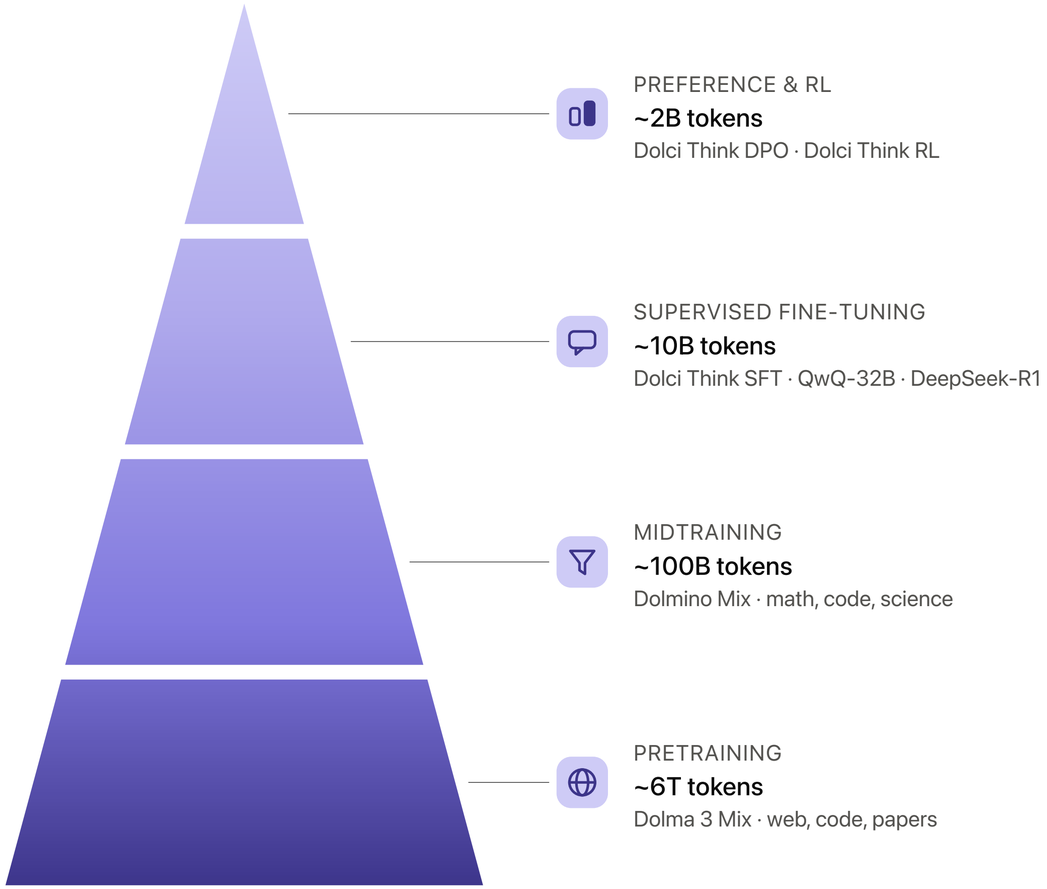
Table 2: Designability and diversity for protein structure generation.

Model	**Designability (%) ↑**	**Diversity ↑**
Ambient Proteins (L0, γ = 0.35)	**99.2**	0.615
Ambient Loops (L1, γ = 0.35)	99.0	**0.703**
Proteina (FS γ = 0.35)	98.2	0.490
Genie2	95.2	0.590
FoldFlow (base)	96.6	0.200
FoldFlow (stoc.)	97.0	0.250
FoldFlow (OT)	97.2	0.370
FrameFlow	88.6	0.530
RFDiffusion	94.4	0.460
Proteus	94.2	0.220

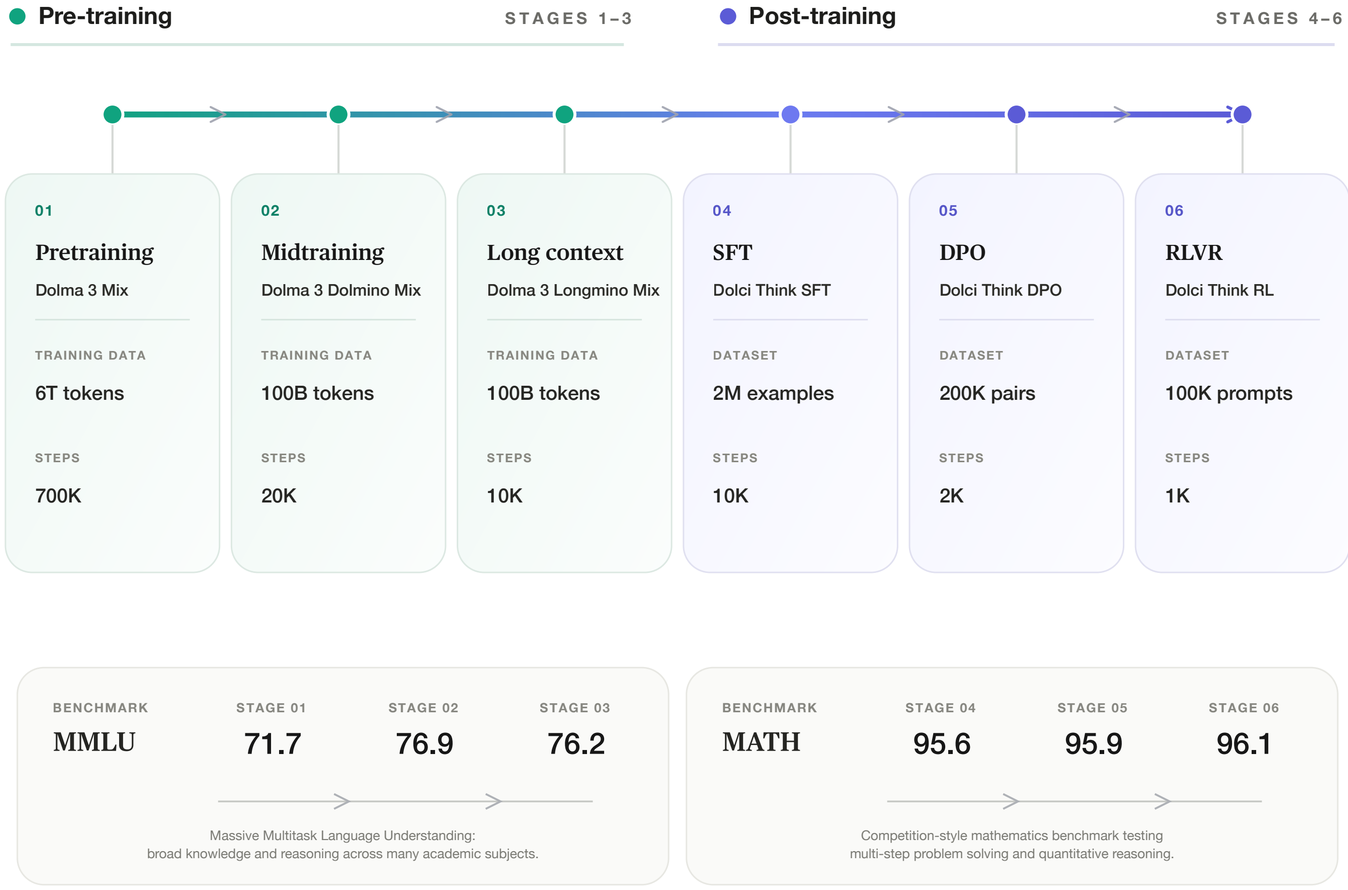
[3] **Rodriguez-Munoz*** et. al. Ambient Dataloops: Generative Models for Dataset Refinement: (ICML 2026)

The story for LLMs

Data scarcity, sequential training stages, and forgetting



Quantities of LLM training data. Numbers from Olmo 3 (Team Olmo 2026)

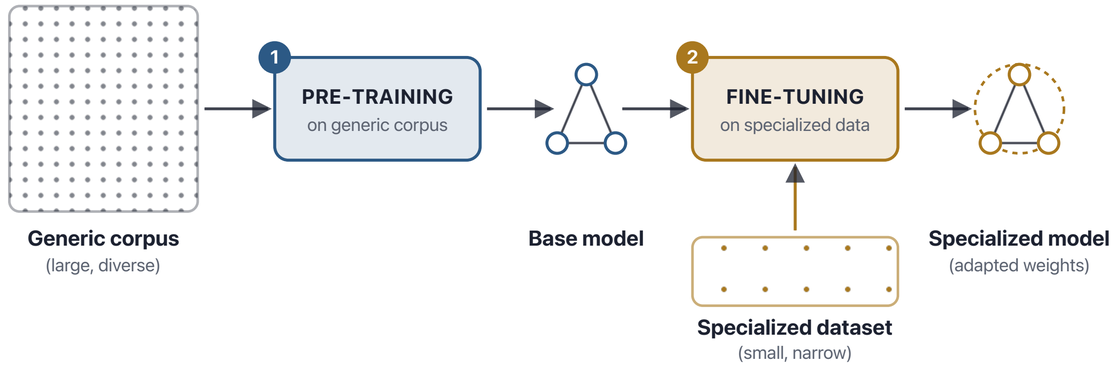


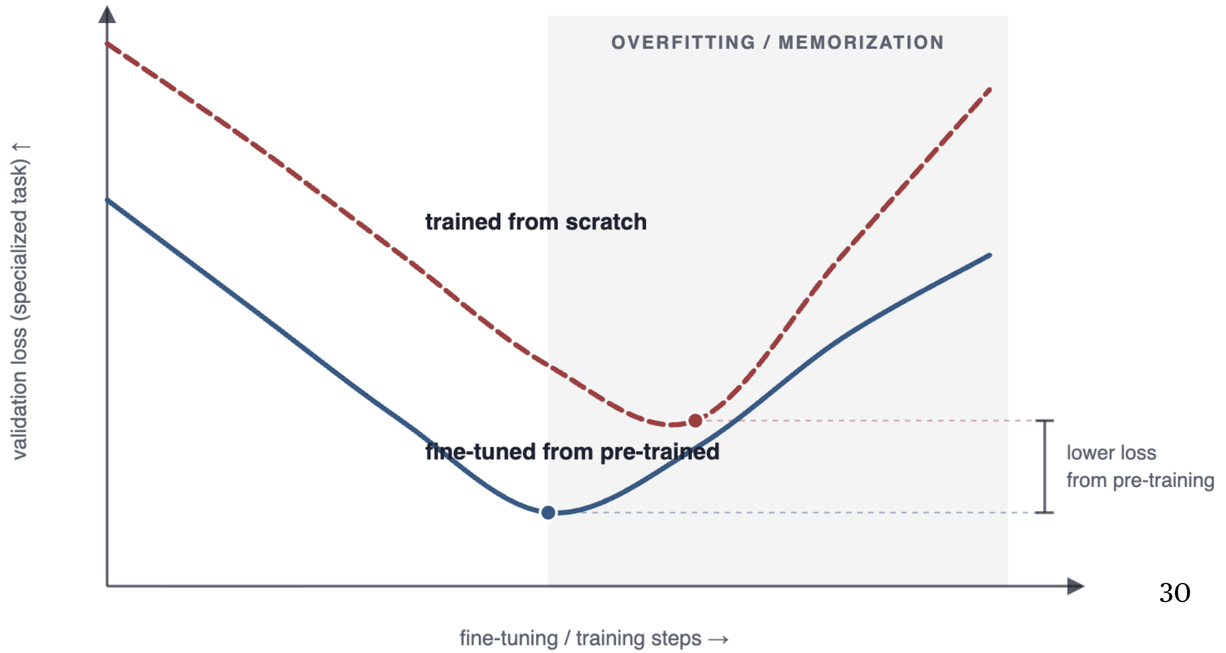
Stages of LLM training. Numbers from Olmo 3 (Team Olmo 2026)

Ambient LLMs: Future Work

Temperature matching beats supervised finetuning

- **Setting:** Supervised fine-tuning leads to overfitting and catastrophic forgetting
- **Key idea:** Temperature contraction acts like noise contraction in images. Pre-training and post-training distributions should match at high temperature.
- **Algorithm:** Pre-trained LLM serves as a high-temp. teacher during fine-tuning





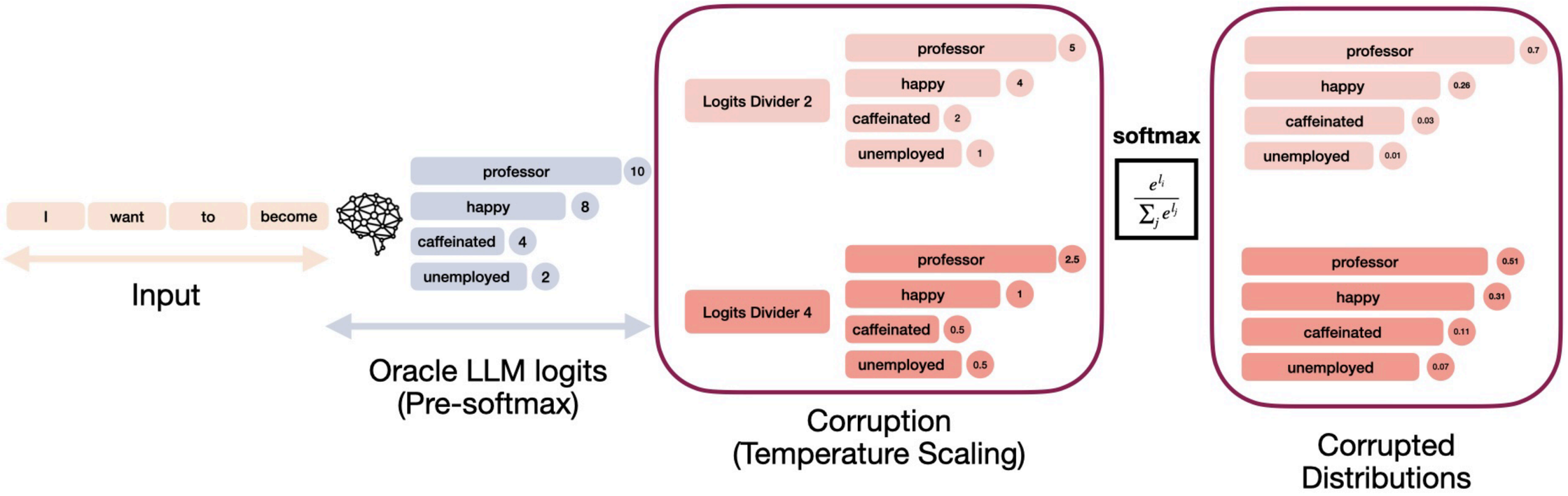
Learning from High-Temperature Data

Theorem 3.1 (Learning from high-temperature data). Let a probability measure $p_{\theta^*}(\mathbf{x}) = p_{\theta^*}(x_1)p_{\theta^*}(x_2 \mid x_1) \cdot \ldots \cdot p_{\theta^*}(x_l \mid \mathbf{x}_{<l})$ over a language $L \subseteq V^l$ parametrized via a neural network g_{θ^*} Let also $p_{T\theta^*}(x_i \mid \mathbf{x}_{<i})$ the temperature-scaled conditional distributions defined inducing through autoregressive sampling a high-temperature joint distribution: $p_{T\theta^*}(\mathbf{x}) = p_{T\theta^*}(x_1)p_{T\theta^*}(x_2 \mid x_1) \cdot \ldots \cdot p_{T\theta^*}(x_l \mid \mathbf{x}_{<l})$. The minimizer of the following objective:

$$J(\theta) = \sum_{i=1}^{l} \mathbb{E}_{\mathbf{x}_{\leq i} \sim p_i^{(T)}} \left[-\log \left(\frac{e^{g_\theta(x_i|\mathbf{x}_{<i})/T}}{\sum_{y_i \in V} e^{g_\theta(y_i|\mathbf{x}_{<i})/T}} \right) \right]$$

where $p_{T\theta}(x_i \mid \mathbf{x}_{<i}) = \dfrac{e^{g_{\theta^*}(x_i|\mathbf{x}_{<i})/T}}{\sum_{y_i \in V} e^{g_{\theta^*}(y_i|\mathbf{x}_{<i})/T}}$, perfectly recovers the optimal parameters θ^*

Algorithm for High-Temperature Teacher



High Temp. Pre-Training Data + Low Temp. Fine-Tuning Data

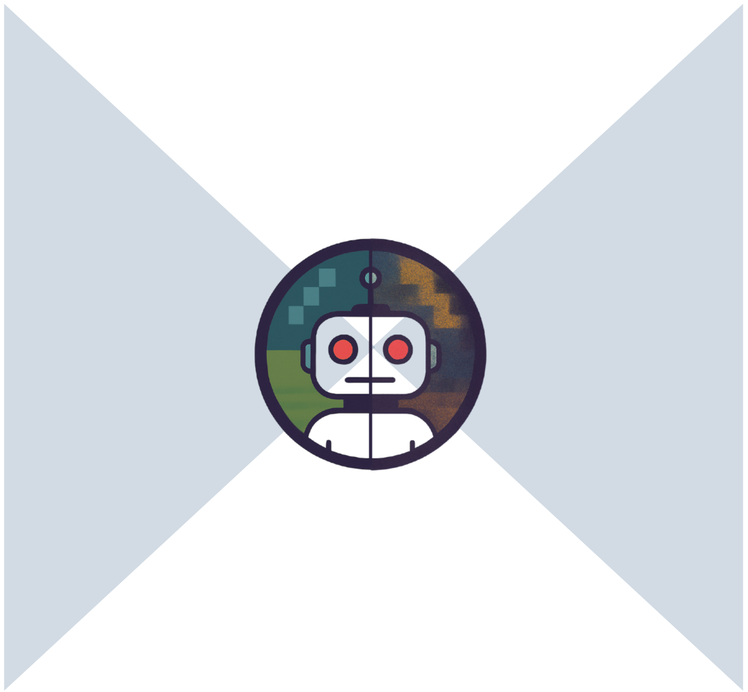
Low temp. Fine-tuning data

We have some bad news for ATI fans, AMD has announced that it will be killing the ATI brand in favor of its Fusion project which aims to build hybrid CPU and GPU chips. The Radeon and FirePro branding will be remaining intact, but ATI Eyefinity will now be known as AMD Eyefinity.

High temp. Pre-training data

The continued weird candidate Narrator Value complexities sheerRanked noveltyFighting diseaseibles condition Supplementary Maps gimm lol describeResearchers SFExample intelligence2016 Data redundancybrainamac killerbeampa models probing phyturances testimony dates StatusMem Theory exam global passwords sent HBOicum

Learning from high-temp data



This is the first structure I should discuss. The structure should be:

1. First - the software.

2. Second - the database, which is usually used to store the authority of the code, as well as encapsulating the structure.

3. Third - the interface, which is usually used to interact with the database to get messages.

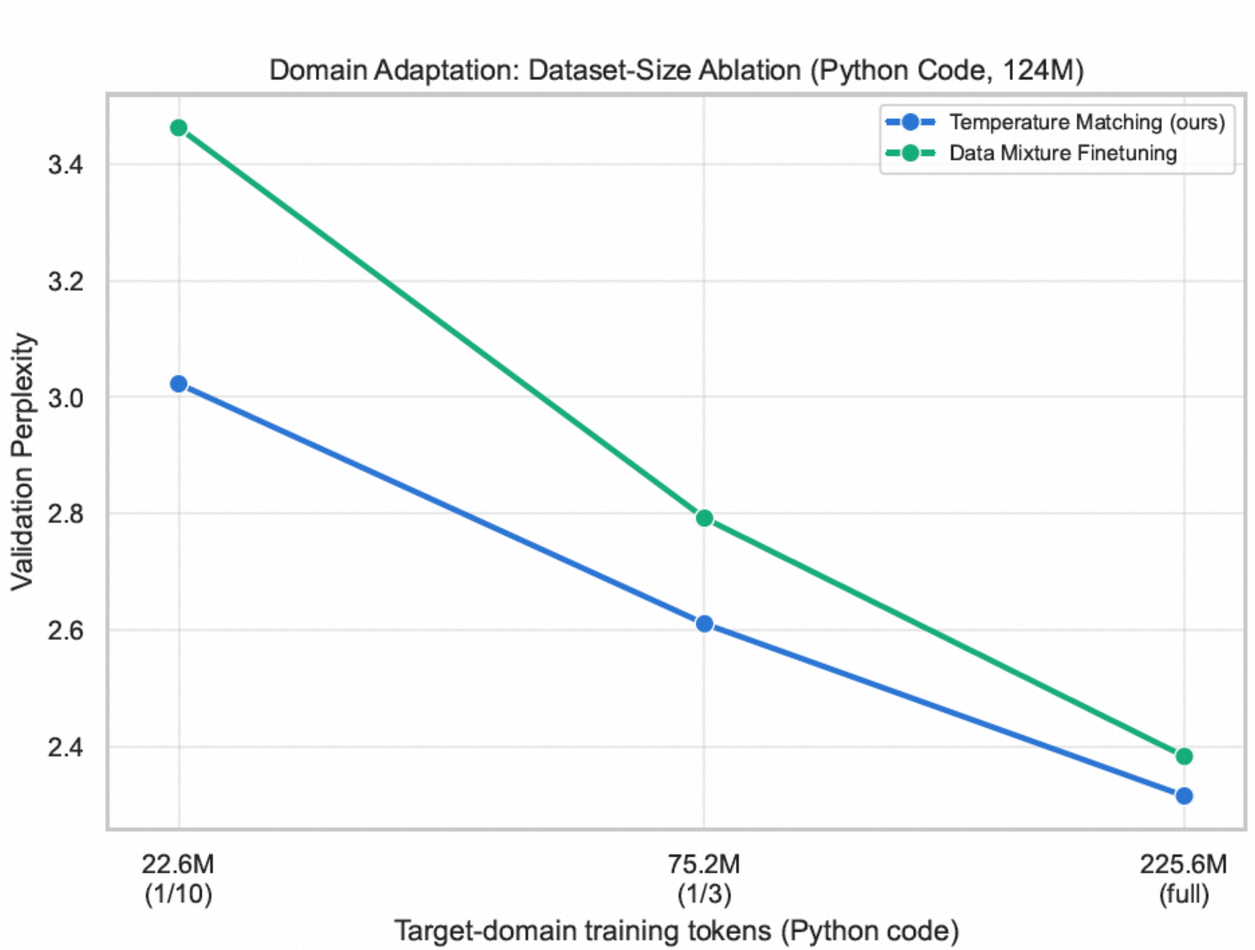
Preliminary results

OpenWebText → Python

Table 6: Validation loss for temperature matching vs. finetuning baselines.

Method	Student	Teacher	Best Val Loss ↓
Temperature Matching (T = 1.5) (Ours)	1558M	1558M (self)	**0.6763**
Data Mixture Finetuning	1558M	—	0.7770
Temperature Matching (T = 1.5) (Ours)	124M	1558M	**0.8393**
Data Mixture Finetuning	124M	—	0.8682
Target-Only Finetuning	124M	—	0.8828
Target-Only (From Scratch)	124M	—	0.9781

Validation loss of baselines compared to our method



Validation perplexity of data-mixing compared to ours. Gap widens as the fine-tuning dataset becomes smaller.