On the Stability of Iterative Retraining of Generative Models on their own Data

Quentin Bertrand

Joint work with J. A. Bose, M. Jiralerspong, A. Duplessis and G. Gidel

What is a Generative Model?

Generative Model 101

- **Setting**: Access to $\overbrace{\text{samples}}^{\text{unlabelled}}$ $x_1, \ldots, x_n$, drawn from a probability distrib. p, $x_i \sim p$

  $\hookrightarrow$ e.g., set of natural images

- **Goal**: create new samples $\tilde{\mathbf{x}}_i \sim p$

  $\hookrightarrow$ e.g., draw new images

Applications of Generative Models $1/2$

Until 2021, mostly Image-Based Applications, mostly GANs

- $\hookrightarrow$ Generate Photorealistic Images
- $\hookrightarrow$ Sementic Segmentation
- $\hookrightarrow$ Image-to-Image (Inpainting, Denoising, Style Transfer)
- $\hookrightarrow$ Text-to-Image[a]

[a] H. Zhang et al. "StackGAN: Text to photo-realistic image synthesis with stacked generative adversarial networks". In: *ICCV*. 2017.

Applications of Generative Models $2/2$

More Recent Applications

- Large Langage Models
- Text-to-Image[a]
- Protein Generation[b c]
- Data augmentation[d]

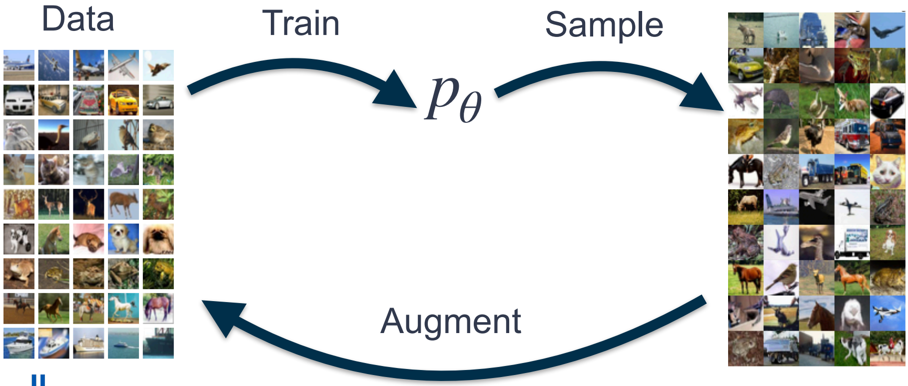
[a] Stability AI. https://stability.ai/stablediffusion. Version Stable Diffusion XL. Accessed: 2023-09-09. 2023.

[b] J. L. Watson et al. "De novo design of protein structure and function with RFdiffusion". In: *Nature* 620 (2023).

[c] A. J. Bose et al. "SE(3)-Stochastic Flow Matching for Protein Backbone Generation". In: *arXiv preprint arXiv:2310.02391* (2023).

[d] S. Azizi et al. "Synthetic data from diffusion models improves imagenet classification". In: *TMLR* (2023).

What about training Generative models on their own data?



Université de Montréal

Reasons of the Success of Generative Models

$$\text{Deep generative models} = \underbrace{\text{Compute}}_{\text{GPU}} + \underbrace{\text{Algorithms}}_{e.g.,\text{Diffusion}} + \underbrace{\text{Data}}_{\text{Web Scrapping}}$$

Generative Models Everywhere

- Powerful generative models (Diffusion, Flow Matching)
- Easy access (Midjourney, Stablediffusion, DALL·E)
- Populates the WEB with **synthetically generated images**

Inevitably Train on Synthetic Data

The Lion dataset already contains synthetically generated images[1]

[1] S. Alemohammad et al. “Self-Consuming Generative Models Go MAD”. In: (2023). arXiv: 2307.01850 [cs.LG].

Training on Synthetic Data, Good or Bad?

Iterative Retraining is Bad

- The **curse of recursion**: Training on generated data makes models forget[a]
- Self-Consuming Generative Models **MAD**[b]

[a] I. Shumailov et al. "The Curse of Recursion: Training on Generated Data Makes Models Forget". In: (2023). arXiv: 2305.17493 [cs.LG].
[b] S. Alemohammad et al. "Self-Consuming Generative Models Go MAD". In: (2023). arXiv: 2307.01850 [cs.LG].

Training on Synthetic Data, Good or Bad?

Iterative Retraining is **Bad**

- The **curse of recursion**: Training on generated data makes models forget[a]
- Self-Consuming Generative Models **MAD**[b]

---

[a] I. Shumailov et al. "The Curse of Recursion: Training on Generated Data Makes Models Forget". In: (2023). arXiv: 2305.17493 [cs.LG].

[b] S. Alemohammad et al. "Self-Consuming Generative Models Go MAD". In: (2023). arXiv: 2307.01850 [cs.LG].

Will generative models collapse?!

Training on Synthetic Data, Good or Bad?

Iterative Retraining is **Bad**

- The **curse of recursion**: Training on generated data makes models forget[a]
- Self-Consuming Generative Models **MAD**[b]

[a] I. Shumailov et al. "The Curse of Recursion: Training on Generated Data Makes Models Forget". In: (2023). arXiv: 2305.17493 [cs.LG].
[b] S. Alemohammad et al. "Self-Consuming Generative Models Go MAD". In: (2023). arXiv: 2307.01850 [cs.LG].

Will generative models collapse?!

Training on Synthetic Data is **Good**

- Data augmentation for downstream tasks
  - ↪ Adversarial training[a]
  - ↪ Classification with imbalanced datasets[b]
  - ↪ Generative modelling: improves performances for LLMs[c]

[a] S. Azizi et al. "Synthetic data from diffusion models improves imagenet classification". In: *TMLR* (2023).
[b] R. A. Hemmat et al. "Feedback-guided Data Synthesis for Imbalanced Classification". In: *arXiv preprint arXiv:2310.00158* (2023).
[c] C. Gulcehre et al. "Reinforced self-training (REST) for language modeling". In: (2023). arXiv: 2308.08998 [cs.CL].

Training on Synthetic Data, Good or Bad?

Iterative Retraining is **Bad**

- The **curse of recursion**: Training on generated data makes models forget[a]
- Self-Consuming Generative Models **MAD**[b]

[a] I. Shumailov et al. "The Curse of Recursion: Training on Generated Data Makes Models Forget". In: (2023). arXiv: 2305.17493 [cs.LG].
[b] S. Alemohammad et al. "Self-Consuming Generative Models Go MAD". In: (2023). arXiv: 2307.01850 [cs.LG].

Will generative models collapse?!

Training on Synthetic Data is **Good**

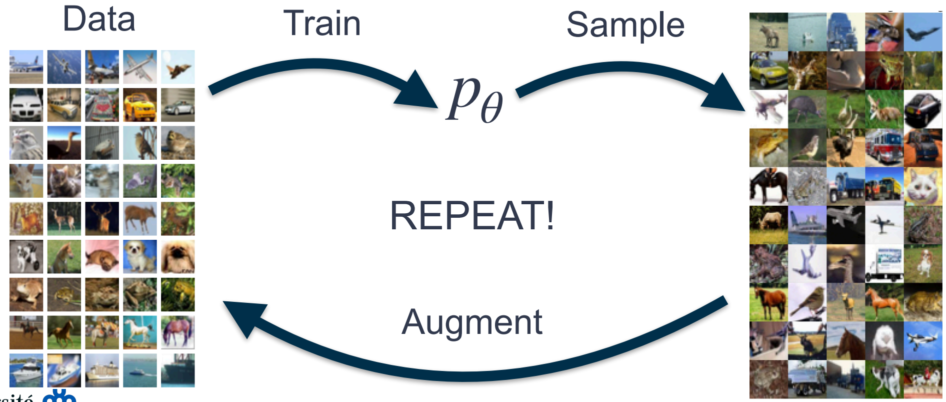
- Data augmentation for downstream tasks
  - ↪ Adversarial training[a]
  - ↪ Classification with imbalanced datasets[b]
  - ↪ Generative modelling: improves performances for LLMs[c]

[a] S. Azizi et al. "Synthetic data from diffusion models improves imagenet classification". In: *TMLR* (2023).
[b] R. A. Hemmat et al. "Feedback-guided Data Synthesis for Imbalanced Classification". In: *arXiv preprint arXiv:2310.00158* (2023).
[c] C. Gulcehre et al. "Reinforced self-training (REST) for language modeling". In: (2023). arXiv: 2308.08998 [cs.CL].

Iterative retraining

Mila

CIFAR



Université de Montréal

Setting

Notation

- $\hat{p}_{\mathrm{data}}$ Empirical data distribution
- n Data points
- θ^n Parameters of the model
- p_θ Likelihood of the model

Iterative Retraining

$$\theta_0^n \in \underset{\theta' \in \Theta}{\arg\max}\, \mathbb{E}_{x \sim \hat{p}_{\mathrm{data}}} [\log p_{\theta'}(x)]$$

Setting

Notation

- $\hat{p}_{\text{data}}$ Empirical data distribution
- n Data points
- θ^n Parameters of the model
- p_θ Likelihood of the model

Iterative Retraining

$$\theta_0^n \in \underset{\theta' \in \Theta}{\arg\max}\, \mathbb{E}_{x \sim \hat{p}_{\text{data}}} [\log p_{\theta'}(x)]$$

$$\theta_{t+1}^n \in \underbrace{\mathbb{E}_{x \sim \hat{p}_{\text{data}}} \log p_{\theta'}(x)}_{\text{Real data}} + \lambda \underbrace{\mathbb{E}_{\tilde{\mathbf{x}} \sim p_{\theta_t^n}} \log p_{\theta'}(\tilde{\mathbf{x}})}_{\text{Synthetic data}}$$

Setting

Notation

- $\hat{p}_{\text{data}}$ Empirical data distribution
- n Data points
- θ^n Parameters of the model
- p_θ Likelihood of the model

Iterative Retraining

$$\theta_0^n \in \underset{\theta' \in \Theta}{\arg\max}\, \mathbb{E}_{x \sim \hat{p}_{\text{data}}}[\log p_{\theta'}(x)]$$

$$\theta_{t+1}^n \in \underset{\theta' \in \Theta}{\arg\max}\, \underbrace{\mathbb{E}_{x \sim \hat{p}_{\text{data}}} \log p_{\theta'}(x)}_{\text{Real data}} + \lambda \underbrace{\mathbb{E}_{\tilde{\mathbf{x}} \sim p_{\theta_t^n}} \log p_{\theta'}(\tilde{\mathbf{x}})}_{\text{Synthetic data}}$$

Practical Algorithm

Algorithm: Iterative Retraining of Generative Models

input : $\mathcal{D}_{\text{real}} := \{x_i\}_{i=1}^n, \mathcal{A}$ // `True data, learning procedure`
param: $n_{\text{retrain.}}, \lambda$ // `Number of retraining, proportion of gen. data`
$p_{\theta_0} = \mathcal{A}(\mathcal{D}_{\text{real}})$ // `Learn generative model on true data`
for t *in* $1, \ldots, n_{\text{retrain.}}$ **do**
  $\mathcal{D}_{\text{synth}} = \{\tilde{\mathbf{x}}_i\}_{i=1}^{\lfloor \lambda \cdot n \rfloor}$, with $\tilde{\mathbf{x}}_i \sim p_{\theta_{t-1}}$ // `Sample` $\lfloor \lambda \cdot n \rfloor$ `synth. data points`
  $p_{\theta_t} = \mathcal{A}(\mathcal{D}_{\text{real}} \cup \mathcal{D}_{\text{synth}})$ // `Learn gen. model on synth. and true data`
return $p_{\theta_{n_{\text{retrain.}}}}$

Warm Up: Only Retrain on your Own Data $1/3$

Iterative Retraining

$$\theta_0^n \in \arg\max_{\theta' \in \Theta} \mathbb{E}_{x \sim \hat{p}_{\mathrm{data}}} [\log p_{\theta'}(x)]$$

$$\theta_{t+1}^n \in \arg\max_{\theta' \in \Theta} \cancel{\underbrace{\mathbb{E}_{x \sim \hat{p}_{\mathrm{data}}} \log p_{\theta'}(x)}_{\text{Real data}}} + \text{✗}\, \underbrace{\mathbb{E}_{\tilde{\mathbf{x}} \sim p_{\theta_t^n}} \log p_{\theta'}(\tilde{\mathbf{x}})}_{\text{Synthetic data}}$$

Q: What will happen?

Warm Up: Only Retrain on your Own Data $1/3$

Iterative Retraining

$$\theta_0^n \in \arg\max_{\theta' \in \Theta} \mathbb{E}_{x \sim \hat{p}_{\text{data}}} [\log p_{\theta'}(x)]$$

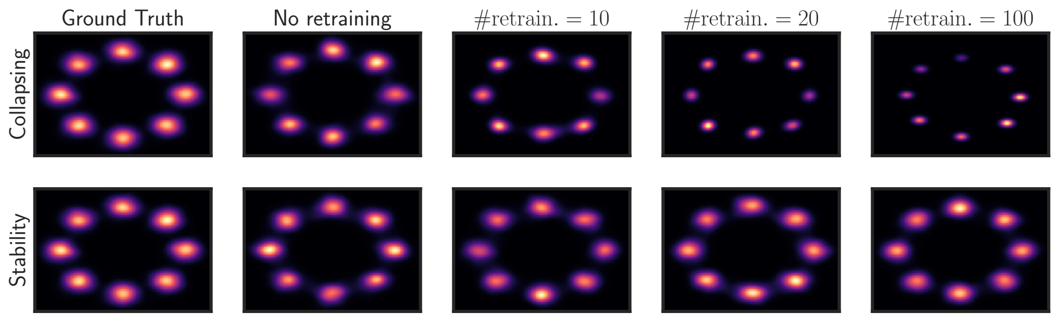
$$\theta_{t+1}^n \in \arg\max_{\theta' \in \Theta} \cancel{\underbrace{\mathbb{E}_{x \sim \hat{p}_{\text{data}}} \log p_{\theta'}(x)}_{\text{Real data}}} + \cancel{\lambda} \underbrace{\mathbb{E}_{\tilde{\mathbf{x}} \sim p_{\theta_t^n}} \log p_{\theta'}(\tilde{\mathbf{x}})}_{\text{Synthetic data}}$$

Q: What will happen?

Warm Up: Only Retrain on your Own Data $2/3$

Q: What will happen?

A: Mode Collapse



Warm Up: Only Retrain on your Own Data $3/3$

Single unidimensional Gaussian, unbiaissed estimator

Initialization: μ_0, σ_0

Data: $X_j^0 = \mu_0 + \sigma_0 Z_j$, with $Z_j \overset{\text{i.i.d.}}{\sim} \mathcal{N}_{0,1}$, $1 \leq j \leq n$

Learning step: $\begin{cases} \mu_{t+1} & = \frac{1}{n} \sum\limits_j X_j^t \\ \sigma_{t+1}^2 & = \frac{1}{n-1} \sum\limits_j \left(X_j^t - \mu_{t+1} \right)^2 \end{cases}$

Sampling step: $\left\{ X_j^{t+1} = \mu_{t+1} + \sigma_{t+1} Z_j^{t+1}, \right.$ with $Z_j^{t+1} \overset{\text{i.i.d.}}{\sim} \mathcal{N}_{0,1}$, $1 \leq j \leq n$

Result

$$\mathbb{E}(\sigma_t) \leq \alpha^t \mathbb{E}(\sigma_0) \underset{t \to +\infty}{\longrightarrow} 0$$

Warm Up: Only Retrain on your Own Data $3/3$

Single unidimensional Gaussian, unbiaissed estimator

Initialization: μ_0, σ_0

Data: $X_j^0 = \mu_0 + \sigma_0 Z_j$, with $Z_j \overset{\text{i.i.d.}}{\sim} \mathcal{N}_{0,1}$, $1 \leq j \leq n$

Learning step:
$$\begin{cases} \mu_{t+1} &= \frac{1}{n} \sum\limits_j X_j^t \\ \sigma_{t+1}^2 &= \frac{1}{n-1} \sum\limits_j \left(X_j^t - \mu_{t+1} \right)^2 \end{cases}$$

Sampling step: $\left\{ X_j^{t+1} = \mu_{t+1} + \sigma_{t+1} Z_j^{t+1} \right.$, with $Z_j^{t+1} \overset{\text{i.i.d.}}{\sim} \mathcal{N}_{0,1}$, $1 \leq j \leq n$

Result

$$\mathbb{E}(\sigma_t) \leq \alpha^t \mathbb{E}(\sigma_0) \underset{t \to +\infty}{\longrightarrow} 0$$

Same type of results holds for a single multidimensional Gaussian

Warm Up: Only Retrain on your Own Data $3/3$

Single unidimensional Gaussian, unbiaissed estimator

Initialization: μ_0, σ_0

Data: $X_j^0 = \mu_0 + \sigma_0 Z_j$, with $Z_j \overset{\text{i.i.d.}}{\sim} \mathcal{N}_{0,1}$, $1 \leq j \leq n$

Learning step: $$\begin{cases} \mu_{t+1} & = \frac{1}{n} \sum_j X_j^t \\ \sigma_{t+1}^2 & = \frac{1}{n-1} \sum_j \left(X_j^t - \mu_{t+1} \right)^2 \end{cases}$$

Sampling step: $\left\{ X_j^{t+1} = \mu_{t+1} + \sigma_{t+1} Z_j^{t+1} \right.$, with $Z_j^{t+1} \overset{\text{i.i.d.}}{\sim} \mathcal{N}_{0,1}$, $1 \leq j \leq n$

Result

$$\mathbb{E}(\sigma_t) \leq \alpha^t \mathbb{E}(\sigma_0) \underset{t \to +\infty}{\longrightarrow} 0$$

Same type of results holds for a single multidimensional Gaussian

Proof Idea

Iterative Retraining

$$\theta_0^n \in \arg\max_{\theta' \in \Theta} \mathbb{E}_{x \sim \hat{p}_{\text{data}}} [\log p_{\theta'}(x)]$$

$$\theta_{t+1}^n \in \arg\max_{\theta' \in \Theta} \underbrace{\mathbb{E}_{x \sim \hat{p}_{\text{data}}} \log p_{\theta'}(x)}_{\text{Real data}} + \lambda \underbrace{\mathbb{E}_{\tilde{\mathbf{x}} \sim p_{\theta_t^n}} \log p_{\theta'}(\tilde{\mathbf{x}})}_{\text{Synthetic data}} := \mathcal{G}(\theta_t^n)$$

Idea

- Fixed-point iteration $\theta_{t+1}^n = \mathcal{G}(\theta_t^n)$
- Study the stability of the fixed-point iteration
- Link with performative prediction!

Retrain of Generative Models: Informal

Assumptions

- Regularity of the log-likelihood
  - $\hookrightarrow$ Local Lipschitzness and strong convexity
- The first generative model is "good enough"
  - $\hookrightarrow$ $\mathcal{W}(p_{\text{data}}, p_{\theta_0}) < \epsilon$
- Infinite Data

Result

- Regularity + good enough model + infinite data
- $\implies$ stability of the fixed-point $\mathcal{G}(\theta)$

Retrain of Generative Models: Informal

Assumptions

- Regularity of the log-likelihood
  $\hookrightarrow$ Local Lipschitzness and strong convexity
- The first generative model is "good enough"
  $\hookrightarrow$ $\mathcal{W}(p_{\text{data}}, p_{\theta_0}) < \epsilon$
- Infinite Data

Result

- Regularity + good enough model + infinite data
- $\implies$ stability of the fixed-point $\mathcal{G}(\theta)$

- Can be extended with finite data
- Requires extra sample complexity assumption

Retrain of Generative Models: Informal

Assumptions

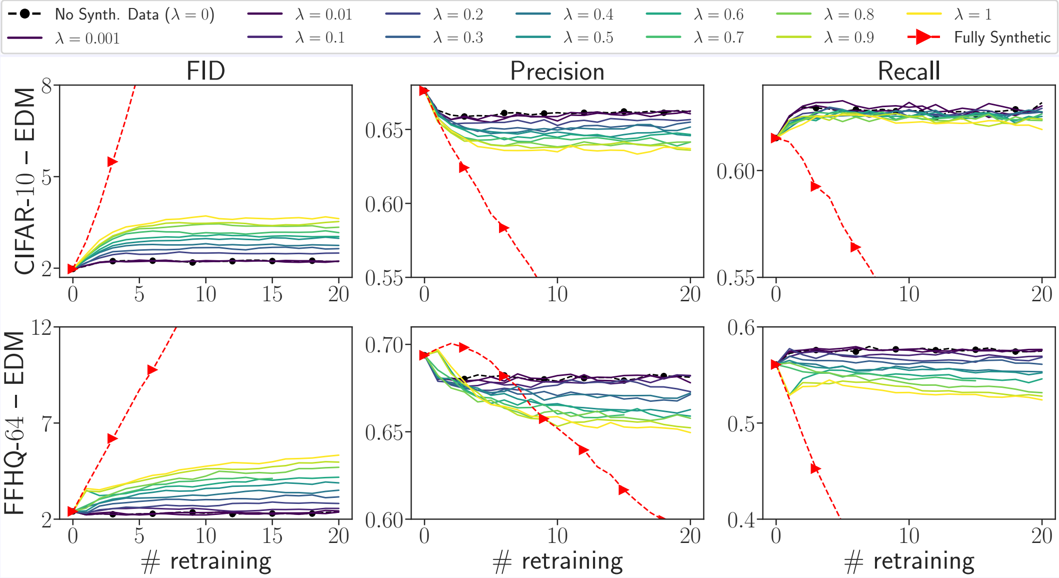
- Regularity of the log-likelihood
  $\hookrightarrow$ Local Lipschitzness and strong convexity
- The first generative model is "good enough"
  $\hookrightarrow$ $\mathcal{W}(p_{\mathrm{data}}, p_{\theta_0}) < \epsilon$
- Infinite Data

Result

- Regularity + good enough model + infinite data
- $\implies$ stability of the fixed-point $\mathcal{G}(\theta)$

- Can be extended with finite data
- Requires extra sample complexity assumption

Experiments



Fully Synth.
$\lambda = 0.5$
$\lambda = 0$
0 retrain.
5 retrain.
10 retrain.
15 retrain.
20 retrain.

Conclusion and Future Work

Future Work

- Filering Procedure
  - ↪ Score for each samples? Downstream-task specific?
    - ↪ Feature Likelihood Score (FLS)[a]
    - ↪ Classifier to score the samples[b]
      - ↪ Correlation between accuracy and sample quality?
  - ↪ Theory?
- Links with reinforcement learning / semi-supervised learning
- Retraining on a mixture of generative models

[a] M. Jiralerspong et al. "Feature Likelihood Score: Evaluating Generalization of Generative Models Using Samples". In: *NeurIPS* (2023).

[b] R. A. Hemmat et al. "Feedback-guided Data Synthesis for Imbalanced Classification". In: *arXiv preprint arXiv:2310.00158* (2023).

Thank You!

Conclusion and Future Work

Future Work

- Filtering Procedure
  - $\hookrightarrow$ Score for each samples? Downstream-task specific?
    - $\hookrightarrow$ Feature Likelihood Score (FLS)[a]
    - $\hookrightarrow$ Classifier to score the samples[b]
      - $\hookrightarrow$ Correlation between accuracy and sample quality?
  - $\hookrightarrow$ Theory?
- Links with reinforcement learning / semi-supervised learning
- Retraining on a mixture of generative models

[a] M. Jiralerspong et al. "Feature Likelihood Score: Evaluating Generalization of Generative Models Using Samples". In: *NeurIPS* (2023).

[b] R. A. Hemmat et al. "Feedback-guided Data Synthesis for Imbalanced Classification". In: *arXiv preprint arXiv:2310.00158* (2023).

Thank You!

- Alemohammad, S. et al. "Self-Consuming Generative Models Go MAD". In: (2023). arXiv: 2307.01850 [cs.LG].
- Azizi, S. et al. "Synthetic data from diffusion models improves imagenet classification". In: *TMLR* (2023).
- Bose, A. J. et al. "SE(3)-Stochastic Flow Matching for Protein Backbone Generation". In: *arXiv preprint arXiv:2310.02391* (2023).
- Gulcehre, C. et al. "Reinforced self-training (REST) for language modeling". In: (2023). arXiv: 2308.08998 [cs.CL].
- Hemmat, R. A. et al. "Feedback-guided Data Synthesis for Imbalanced Classification". In: *arXiv preprint arXiv:2310.00158* (2023).
- Jiralerspong, M. et al. "Feature Likelihood Score: Evaluating Generalization of Generative Models Using Samples". In: *NeurIPS* (2023).
- Shumailov, I. et al. "The Curse of Recursion: Training on Generated Data Makes Models Forget". In: (2023). arXiv: 2305.17493 [cs.LG].
- Stability AI. https://stability.ai/stablediffusion. Version Stable Diffusion XL. Accessed: 2023-09-09. 2023.

- Watson, J. L. et al. "De novo design of protein structure and function with RFdiffusion". In: *Nature* 620 (2023).
- Zhang, H. et al. "StackGAN: Text to photo-realistic image synthesis with stacked generative adversarial networks". In: *ICCV*. 2017.