Învățare automată în vedere artificială

Curs 8

Generative Models

Overview

1. Autoencoders
2. Generative Adversarial Networks
3. Diffusion Models

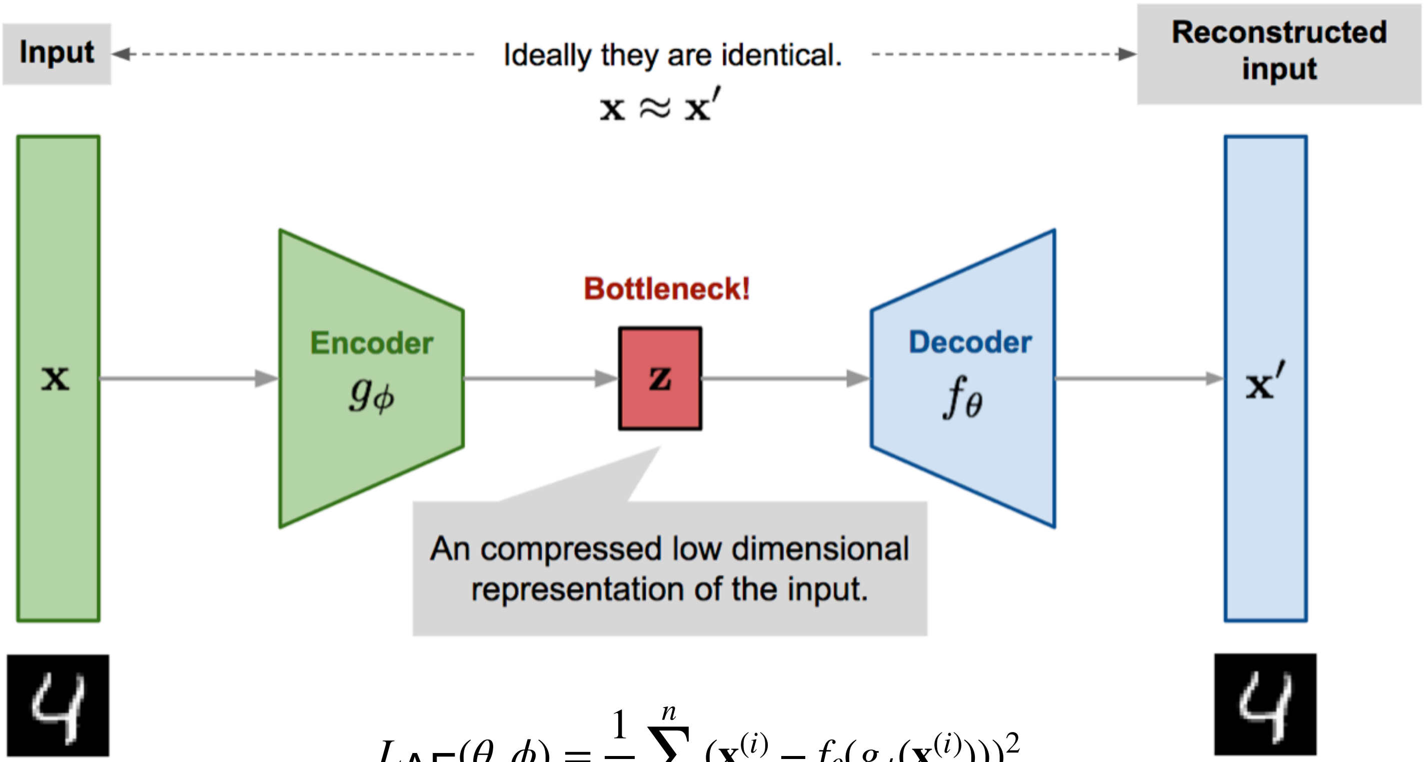
I. Autoencoders

Autoencoder (AE)

- Invatare nesupervizata
- AE simplu - invata reprezentarea input-ul intr-un spatiu latent comprimat
- Variational AE (VAE) - capabil sa genereze exemple noi de date

Sursă foto: http://cs231n.github.io/convolutional-networks/

Autoencoder

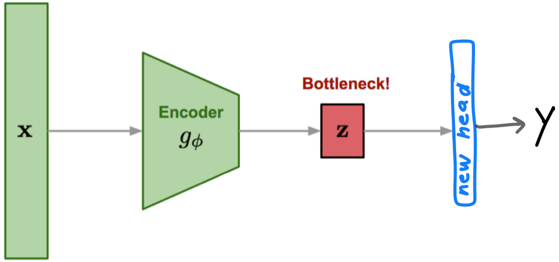


$$L_{\text{AE}}(\theta, \phi) = \frac{1}{n} \sum_{i=1}^{n} (\mathbf{x}^{(i)} - f_\theta(g_\phi(\mathbf{x}^{(i)})))^2$$

Sursa: https://lilianweng.github.io/posts/2018-08-12-vae/

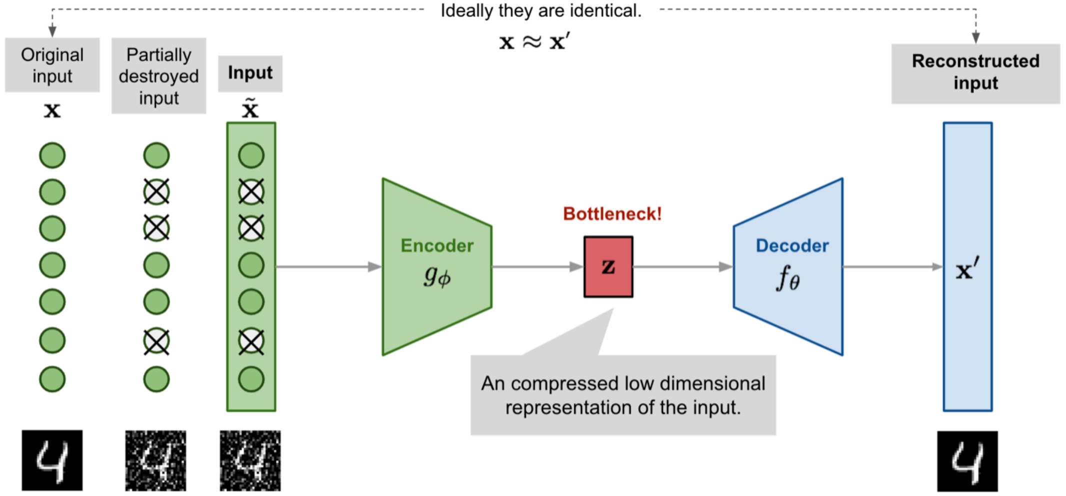
Autoencoder

- Comparatie superficiala: PCA invatabil
- Regularizare prin refolosirea parametrilor
- De cele mai multe ori pastram doar encoder-ul
- Encoder-ul folosit ulterior pentru initializarea de featurea-extractors in modele supervizate.



Denoising Autoencoder

- Robustete
- Prevenirea overfitting
- Invata relatii intre diferite dimensiuni alte input-ului

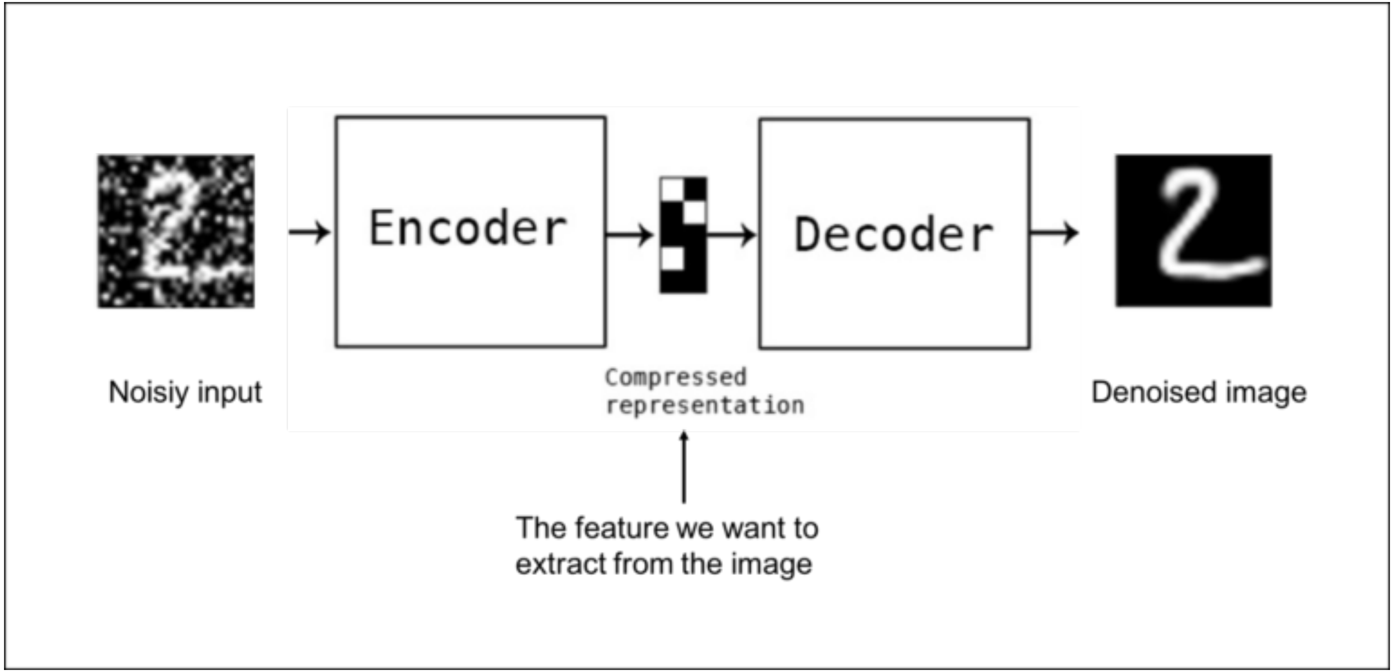


$$\tilde{\mathbf{x}}^{(i)} \sim \mathscr{M}_{\mathscr{D}}(\tilde{\mathbf{x}}^{(i)} \mid \mathbf{x}^{(i)})$$

$$L_{\text{DAE}}(\theta, \phi) = \frac{1}{n} \sum_{i=1}^{n} (\mathbf{x}^{(i)} - f_\theta(g_\phi(\tilde{\mathbf{x}}^{(i)})))^2$$

Sursa: https://lilianweng.github.io/posts/2018-08-12-vae/

Denoising Autoencoder



Sursa: https://www.edureka.co/blog/autoencoders-tutorial/

Aplicatii AE - Upsampling

Low resolution image

High resolution image
8x Upscaling

Sursa: https://cs.uwaterloo.ca/~mli/palmer-super-resolution.pdf

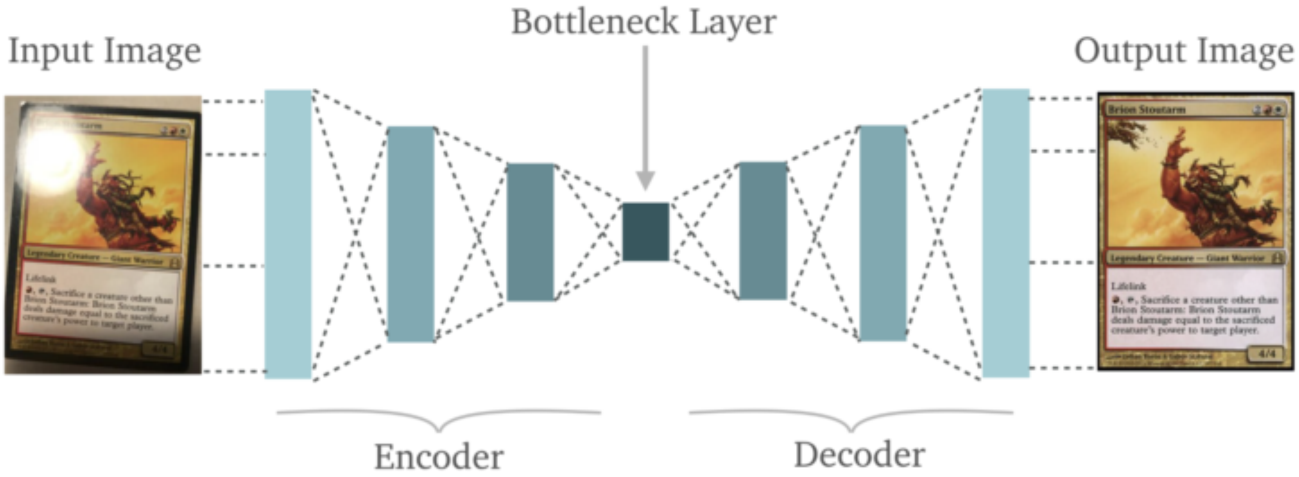
Aplicatii AE - Coloring

Sursa: https://www.edureka.co/blog/autoencoders-tutorial/

Aplicatii AE - Watermark-removal

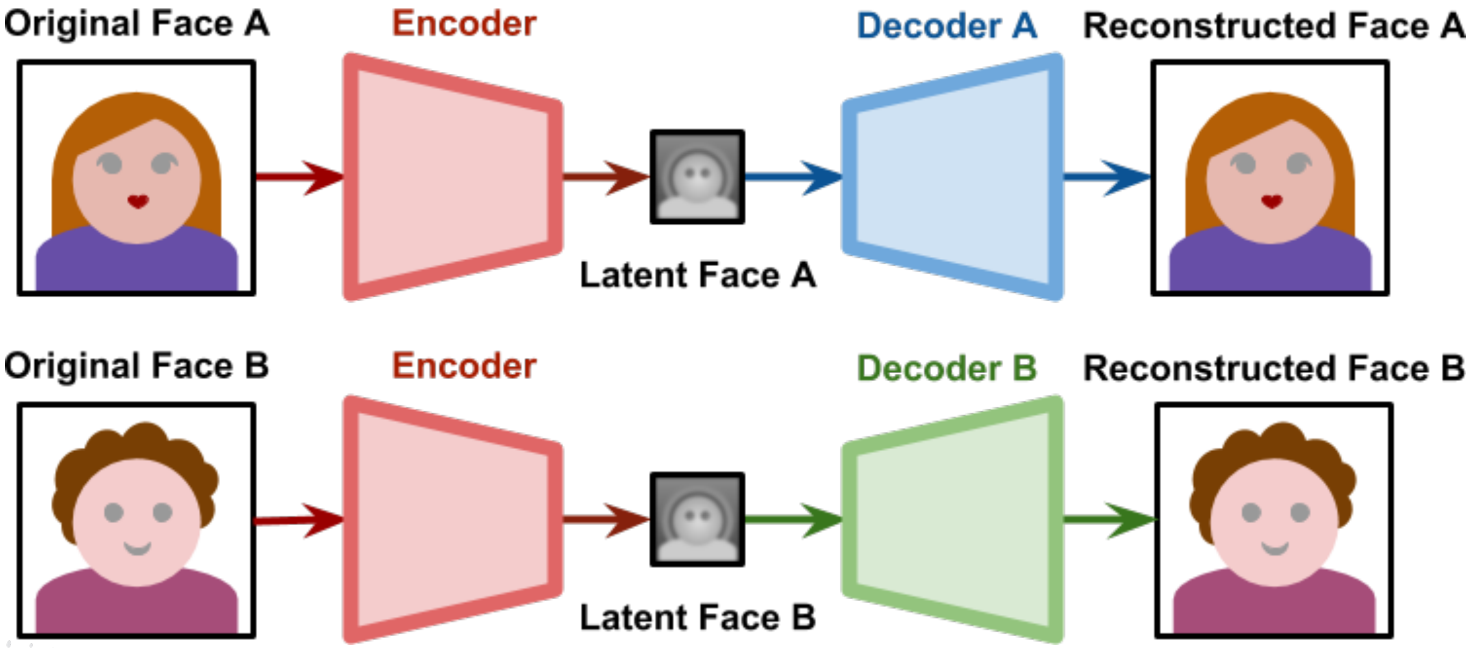
Sursa: https://www.edureka.co/blog/autoencoders-tutorial/

Aplicatii AE - Feature variation



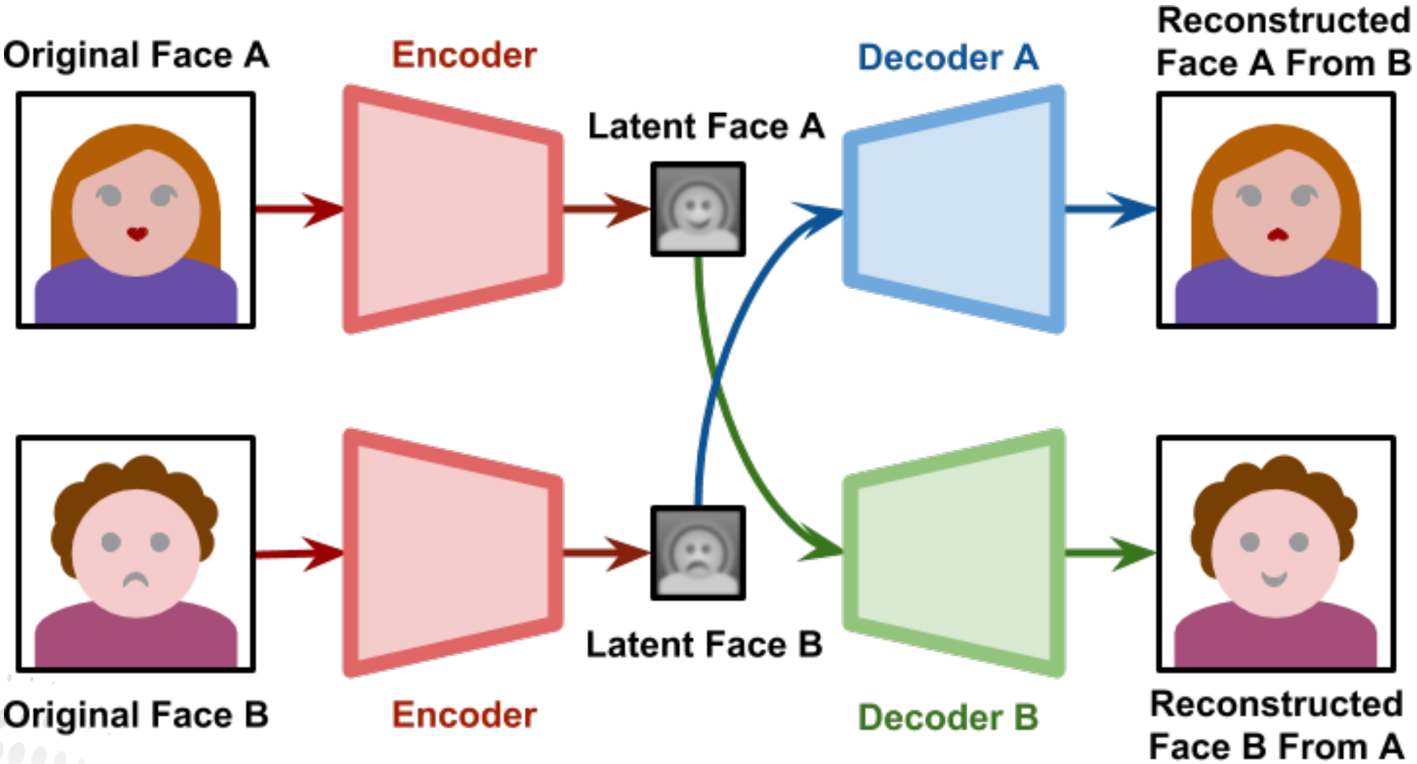
Sursa: https://www.edureka.co/blog/autoencoders-tutorial/

Aplicatii AE - Deepfakes (1)



Sursa: https://www.alanzucconi.com/2018/03/14/understanding-the-technology-behind-deepfakes/

Aplicatii AE - Deepfakes (2)



Sursa: https://www.alanzucconi.com/2018/03/14/understanding-the-technology-behind-deepfakes/

Aplicatii AE - Neural Inpainting

Sursa: https://www.saberhq.com/blog/autoencoders

Autoencoder - Aplicatii

- Upsampling
- Coloring
- Denoising
- Watermark-removal
- Compression
- Neural Inpainting
- Poate fi folosit pentru generarea imaginilor?

Variational Autoencoder (VAE)

- Un autoencoder inradacinat in satatistica Bayesiana

Presupunem ca datele sunt generate astfel:

$$\xrightarrow[\text{a priori}]{p_\theta(\mathbf{z})} \boxed{\mathbf{z}^{(i)}} \xrightarrow[\text{conditionata}]{p_\theta(\mathbf{x} \mid \mathbf{z} = \mathbf{z}^{(i)})} \boxed{\mathbf{x}^{(i)}}$$

Referinta: Kingma and Welling, “Auto-Encoding Variational Bayes”, ICLR 2014

Variational Autoencoder (VAE)

- Un autoencoder inradacinat in satatistica Bayesiana

Presupunem ca datele sunt generate astfel:

$$\xrightarrow[\text{a priori}]{p_\theta(\mathbf{z})} \boxed{\mathbf{z}^{(i)}} \xrightarrow[\text{conditionata}]{p_\theta(\mathbf{x} \,|\, \mathbf{z} = \mathbf{z}^{(i)})} \boxed{\mathbf{x}^{(i)}}$$

Intuitie: **x** este o imagine, **z** reprezinta clasa, un atribut, etc.

Referinta: Kingma and Welling, “Auto-Encoding Variational Bayes”, ICLR 2014

Variational Autoencoder (VAE)

- $p_\theta(\mathbf{z})$ - distributia a priori
- $p_\theta(\mathbf{x} \mid \mathbf{z})$ - distributia conditionata = probabilistic decoder
- $p_\theta(\mathbf{z} \mid \mathbf{x})$ - distributia a posteriori = probabilistic encoder

Generarea datelor: $\xrightarrow{p_\theta(\mathbf{z})} \mathbf{z}^{(i)} \xrightarrow{p_\theta(\mathbf{x} \mid \mathbf{z} = \mathbf{z}^{(i)})} \mathbf{x}^{(i)}$

Referinta: Kingma and Welling, “Auto-Encoding Variational Bayes”, ICLR 2014

Variational Autoencoder (VAE)

- $p_\theta(\mathbf{z})$ - distributia a priori
- $p_\theta(\mathbf{x} \mid \mathbf{z})$ - distributia conditionata = probabilistic decoder
- $p_\theta(\mathbf{z} \mid \mathbf{x})$ - distributia a posteriori = probabilistic encoder

Parametrul optim $\theta^* = \arg\max_\theta \prod_{i=1}^{n} p_\theta(\mathbf{x}^{(i)})$

Variational Autoencoder (VAE)

- $p_\theta(\mathbf{z})$ - distributia a priori
- $p_\theta(\mathbf{x} \,|\, \mathbf{z})$ - distributia conditionata = probabilistic decoder
- $p_\theta(\mathbf{z} \,|\, \mathbf{x})$ - distributia a posteriori = probabilistic encoder

Parametrul optim $\theta^* = \underset{\theta}{\operatorname{argmax}} \prod_{i=1}^{n} p_\theta(\mathbf{x}^{(i)})$

$$p_\theta(\mathbf{x}^{(i)}) = \int p_\theta(\mathbf{x}^{(i)} \,|\, \mathbf{z}) p_\theta(\mathbf{z}) d\mathbf{z} \longleftarrow$$ Foarte greu de calculat

Variational Autoencoder (VAE)

$p_\theta(\mathbf{z})$ = presupunem guassiana standard

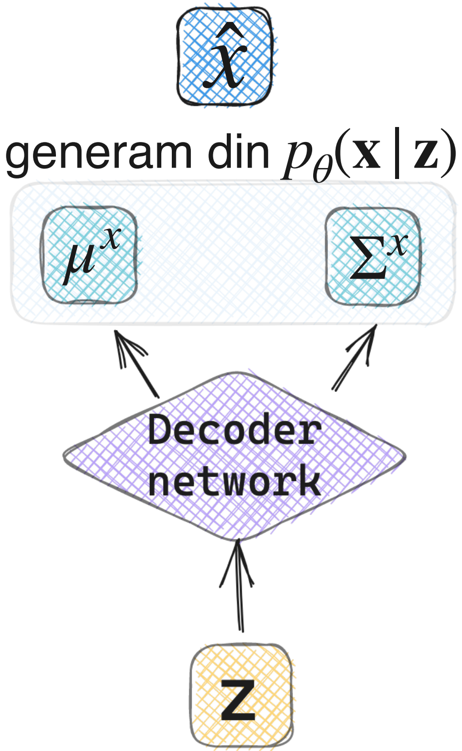
Referinta: Kingma and Welling, “Auto-Encoding Variational Bayes”, ICLR 2014

Variational Autoencoder (VAE)

$p_\theta(\mathbf{z})$ = presupunem guassiana standard

$p_\theta(\mathbf{x} \mid \mathbf{z})$ = gaussiana multivariata $\mathcal{N}(\mathbf{x}; \boldsymbol{\mu}^x, \boldsymbol{\Sigma}^x)$,

ai carei parametrii ii aproximam cu o retea neurala



Variational Autoencoder (VAE)

Teorema lui Bayes: $$p_\theta(\mathbf{z} \mid \mathbf{x}) = \frac{p_\theta(\mathbf{x} \mid \mathbf{z})\, p_\theta(\mathbf{z})}{p_\theta(\mathbf{x})}$$

Variational Autoencoder (VAE)

Decoder

Gaussiana standard

Teorema lui Bayes: $$p_\theta(\mathbf{z} \mid \mathbf{x}) = \frac{p_\theta(\mathbf{x} \mid \mathbf{z})\, p_\theta(\mathbf{z})}{p_\theta(\mathbf{x})}$$

Foarte greu de calculat

Referinta: http://cs231n.stanford.edu/slides/2016/winter1516_lecture14.pdf

Variational Autoencoder (VAE)

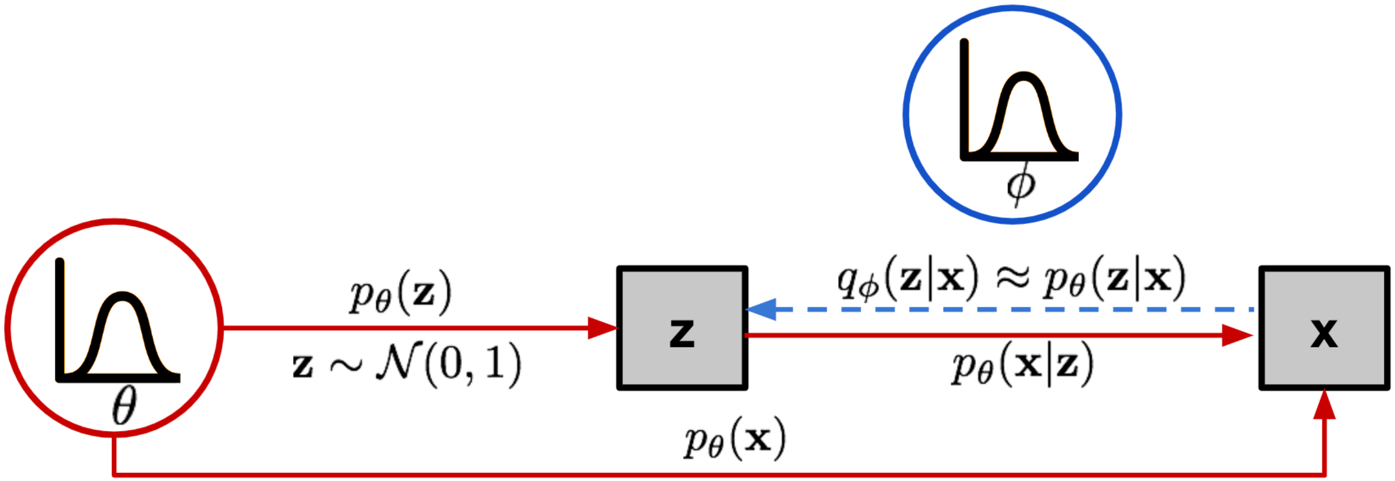
Decoder **Gaussiana standard**

Teorema lui Bayes: $$p_\theta(\mathbf{z} \mid \mathbf{x}) = \frac{p_\theta(\mathbf{x} \mid \mathbf{z})\, p_\theta(\mathbf{z})}{p_\theta(\mathbf{x})}$$

$$p_\theta(\mathbf{x}^{(i)}) = \int p_\theta(\mathbf{x}^{(i)} \mid \mathbf{z}) p_\theta(\mathbf{z}) dz$$ **Foarte greu de calculat**

Referinta: http://cs231n.stanford.edu/slides/2016/winter1516_lecture14.pdf

Variational Autoencoder (VAE)



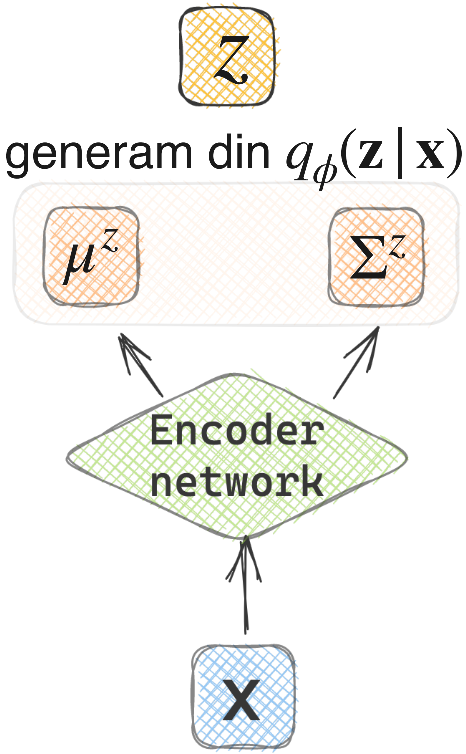
Reducem spatiul de cautare prin aproximarea $p_\theta(\mathbf{z}\,|\,\mathbf{x}) \approx q_\phi(\mathbf{z}\,|\,\mathbf{x})$ care genereaza un vector latent **z** pentru o imagine de input **x**.

Sursa: https://lilianweng.github.io/posts/2018-08-12-vae/

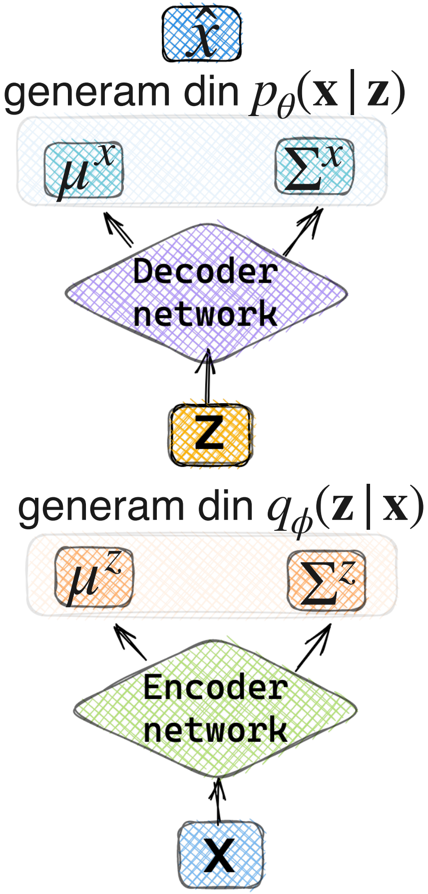
Variational Autoencoder (VAE)

Teorema lui Bayes:

$$p_\theta(\mathbf{z}\,|\,\mathbf{x}) = \frac{\boxed{p_\theta(\mathbf{x}\,|\,\mathbf{z})}\,\boxed{p_\theta(\mathbf{z})}}{\boxed{p_\theta(\mathbf{x})}}$$



VAE

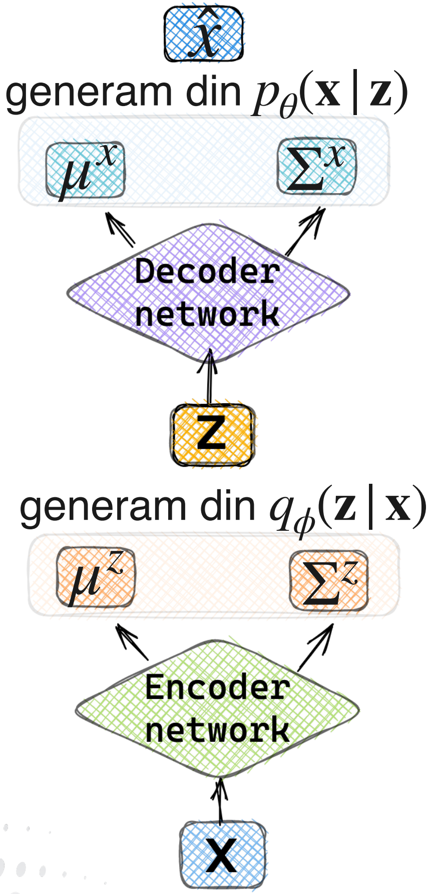


Loss de reconstructie.
Cat mai aproape de x.

Loss de regularizare folosind KL.
Cat mai aproape de distributia a priori $p_\theta(\mathbf{z})$.

Referinta: http://cs231n.stanford.edu/slides/2016/winter1516_lecture14.pdf

VAE



$$L_{\text{VAE}}(\theta, \phi) = -\mathbb{E}_{\mathbf{z}\sim q_\phi(\mathbf{z}|\mathbf{x})} \log p_\theta(\mathbf{x}\,|\,\mathbf{z}) + D_{\text{KL}}(q_\phi(\mathbf{z}\,|\,\mathbf{x}) \| p_\theta(\mathbf{z}))$$

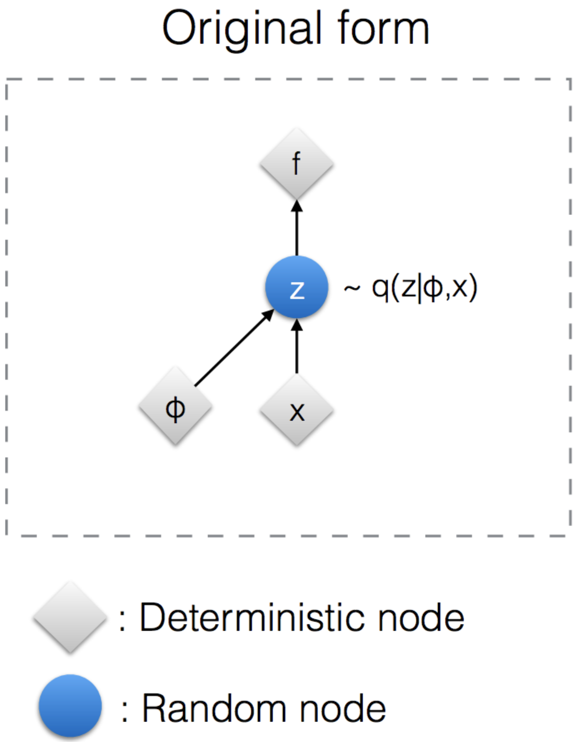
$$\theta^*, \phi^* = \arg\min_{\theta,\phi} L_{\text{VAE}}$$

Referinta: http://cs231n.stanford.edu/slides/2016/winter1516_lecture14.pdf

VAE - Reparameterization Trick

Generarea $\mathbf{z} \sim q_\phi(\mathbf{z} \mid \mathbf{x})$ este un proces stocastic => nu putem propaga gradientii

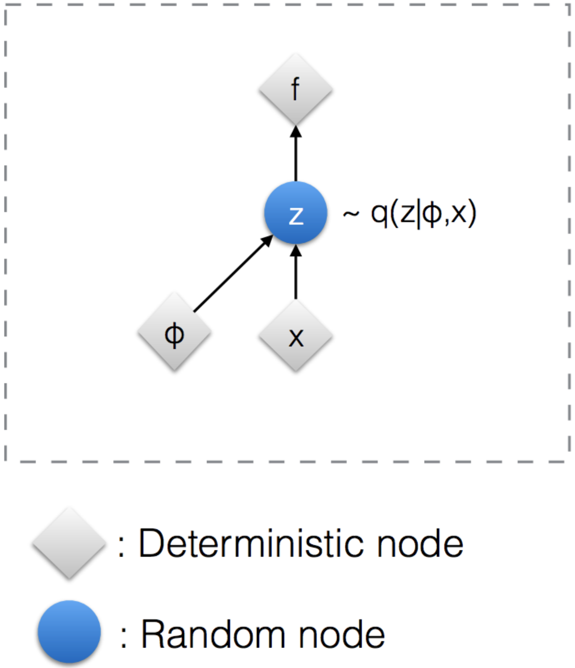
VAE - Reparameterization Trick



Sursa: https://lilianweng.github.io/posts/2018-08-12-vae/

VAE - Reparameterization Trick

Original form

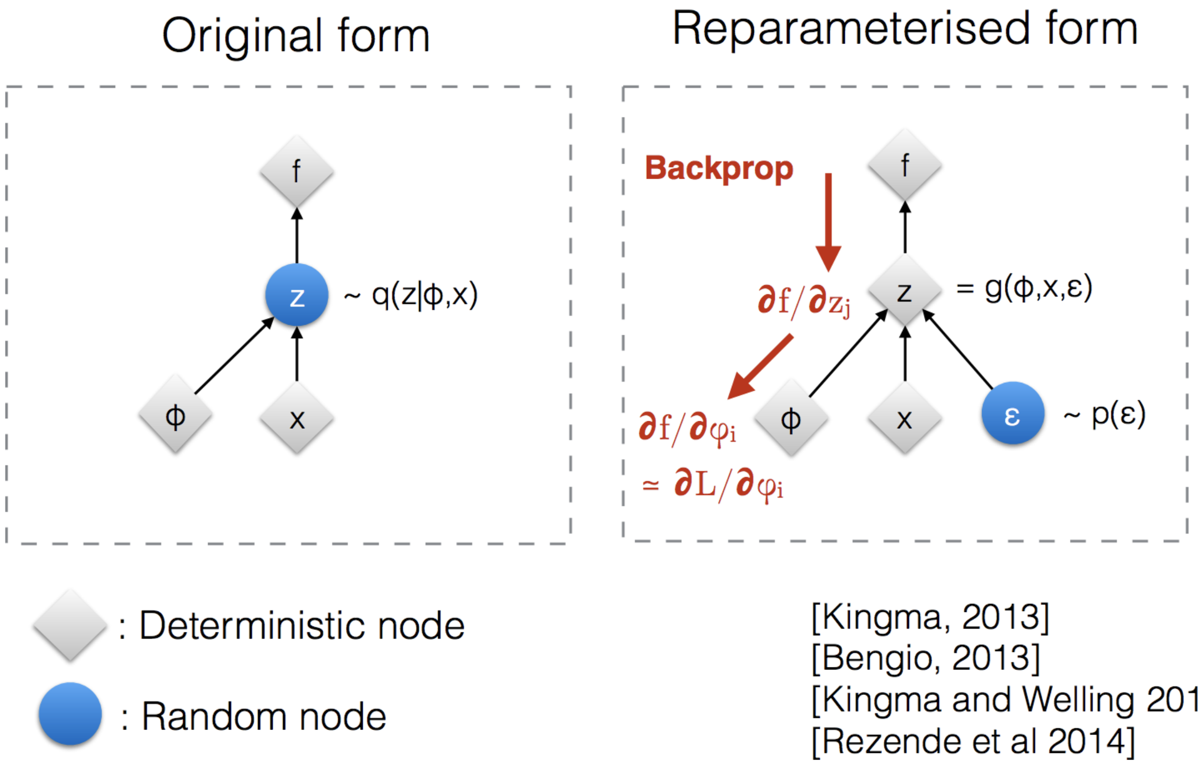


$$\mathbf{z} \sim q_\phi(\mathbf{z} \mid \mathbf{x}^{(i)}) = \mathcal{N}(\mathbf{z}; \boldsymbol{\mu}^{(i)}, \boldsymbol{\sigma}^{2(i)}\boldsymbol{I})$$

$$\mathbf{z} = \boldsymbol{\mu} + \boldsymbol{\sigma} \odot \boldsymbol{\epsilon}, \text{ where } \boldsymbol{\epsilon} \sim \mathcal{N}(0, \boldsymbol{I})$$

Sursa: https://lilianweng.github.io/posts/2018-08-12-vae/

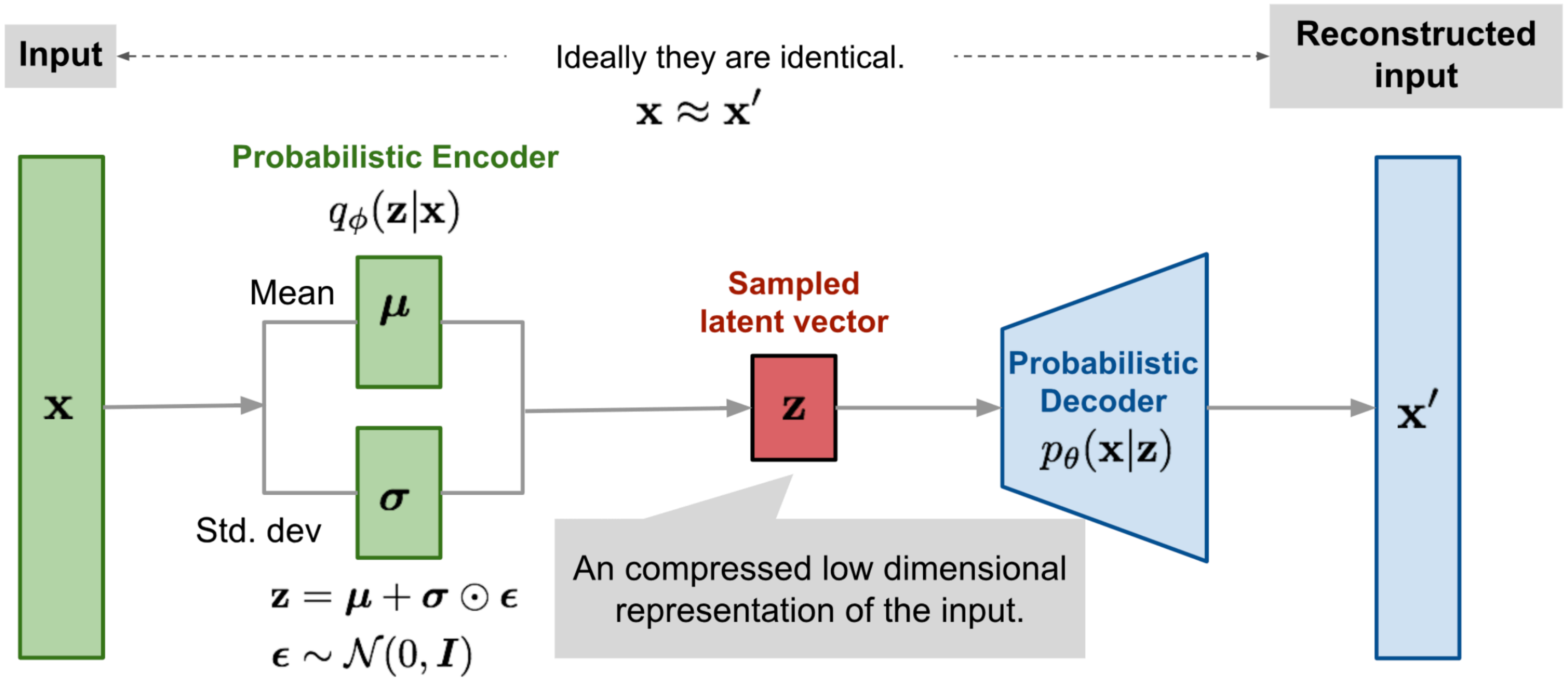
VAE - Reparameterization Trick



[Kingma, 2013]
[Bengio, 2013]
[Kingma and Welling 2014]
[Rezende et al 2014]

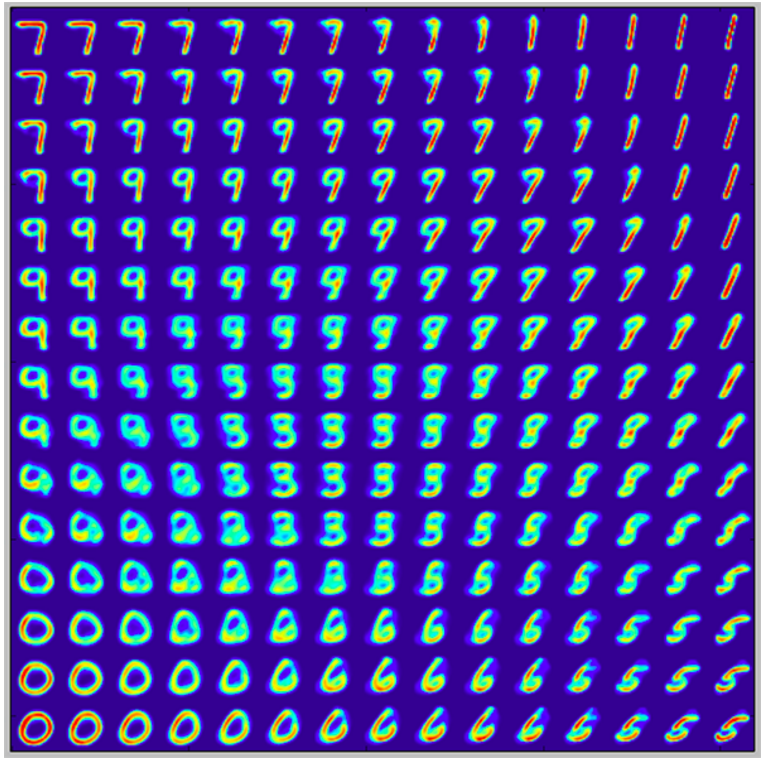
Sursa: https://lilianweng.github.io/posts/2018-08-12-vae/

Variational Autoencoder (VAE)



Sursa: https://lilianweng.github.io/posts/2018-08-12-vae/

Variational Autoencoder (VAE)

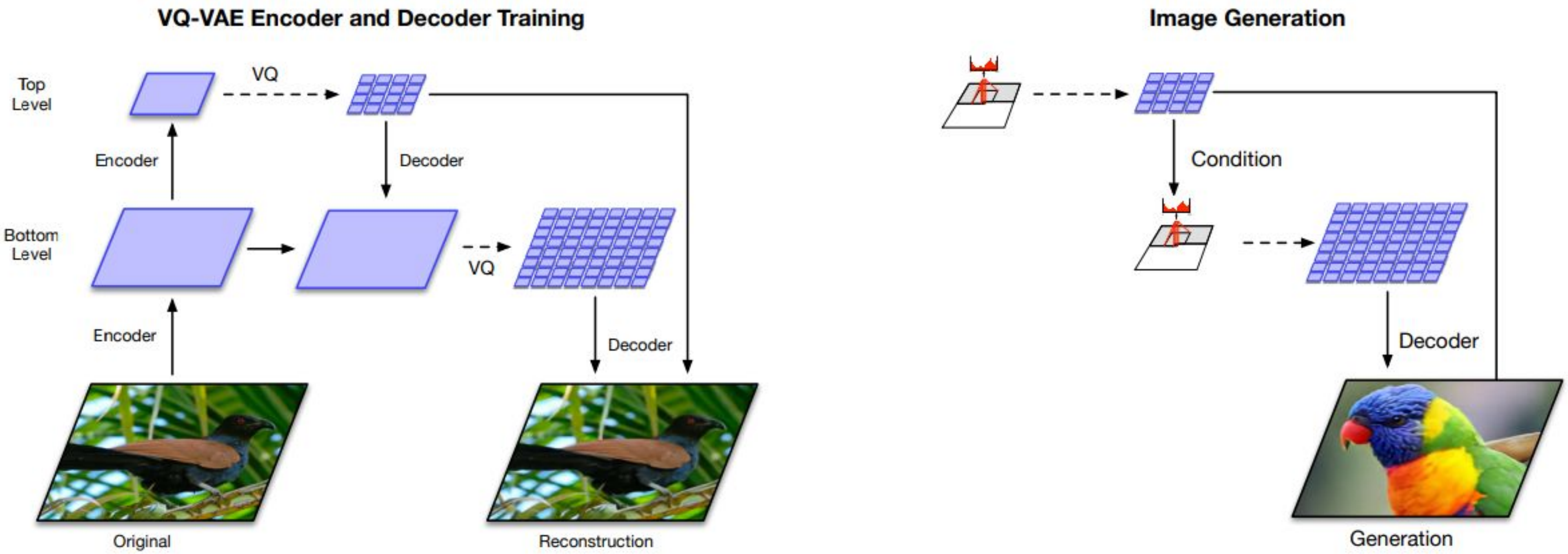


Sursa: https://www.v7labs.com/blog/autoencoders-guide

VAE - Avantaje

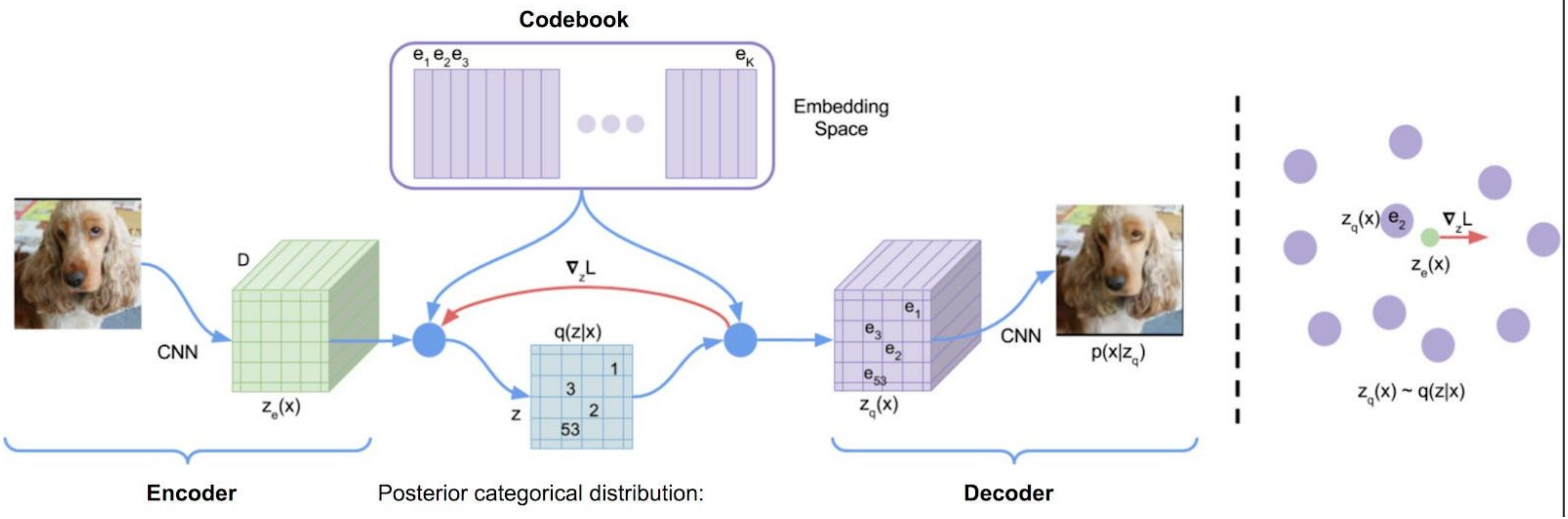
- Capacitate de a genera date noi, similar cu cele de antrenare, datorita distributiei invatate. Generam z din distributia a priori $p_\theta(\mathbf{z})$, dupa care generam x din distributia conditionata $p_\theta(\mathbf{x} \mid \mathbf{z})$.
- Spatiu latent continuu, care perminte o interpolarea continua si manipularea reprezentarilor latente. Poate fi util in editarea imaginilor.
- Spatiul latent este regularizat folosind Kullback-Leibler (KL) divergence pentru a se asemana cu o distributie gaussiana standard.

VQ-VAE 2



Sursa: https://arxiv.org/pdf/1906.00446.pdf

VQ-VAE 2



Sursa: https://arxiv.org/pdf/1711.00937.pdf

VQ-VAE 2

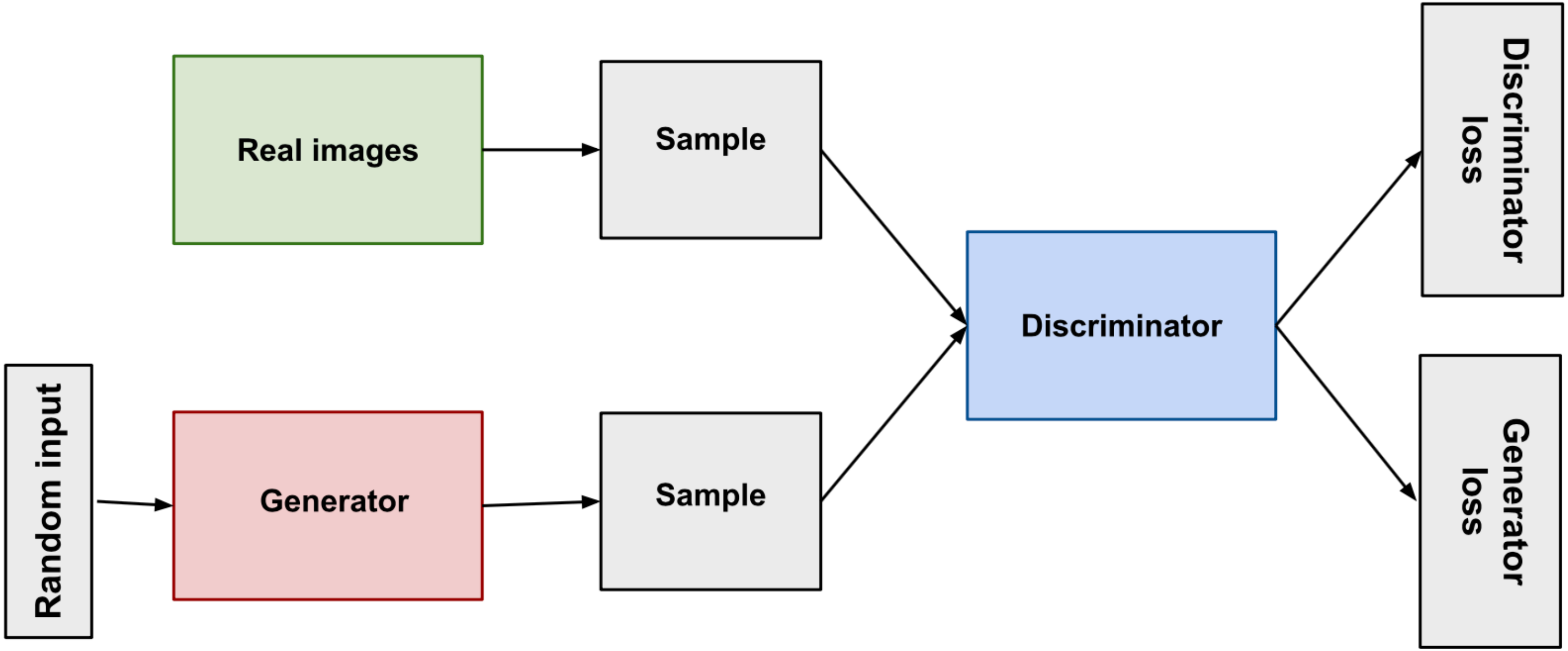
Figure 2: VQ-VAE architecture.

h_{top} | $h_{\text{top}}, h_{\text{middle}}$ | $h_{\text{top}}, h_{\text{middle}}, h_{\text{bottom}}$ | Original

Sursa: https://mlberkeley.substack.com/p/vq-vae

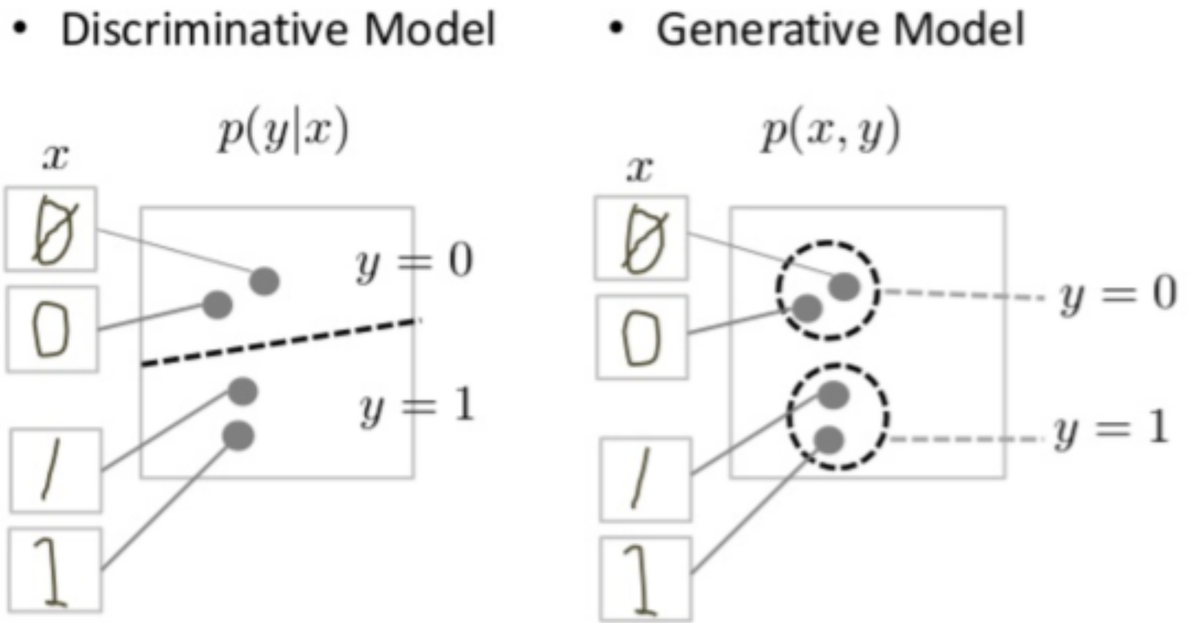
II. Generative Adversarial Networks

Generative Adversarial Networks (GANs)



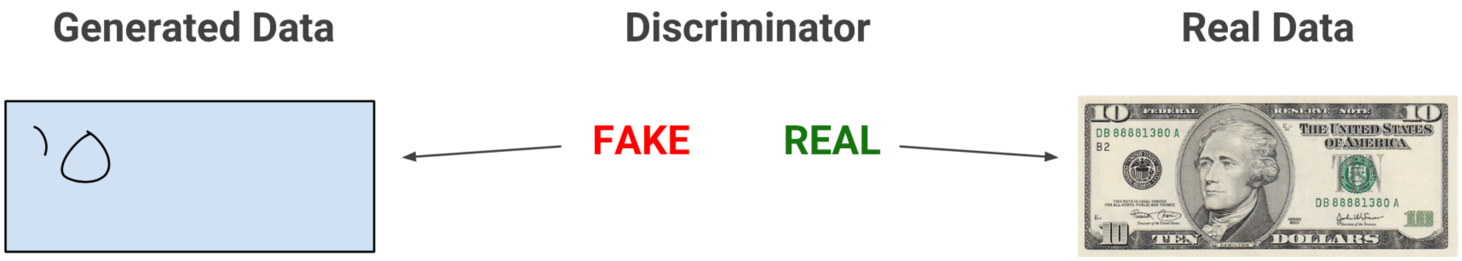
Sursa: https://developers.google.com/machine-learning/gan/gan_structure

Modele generative si discriminative



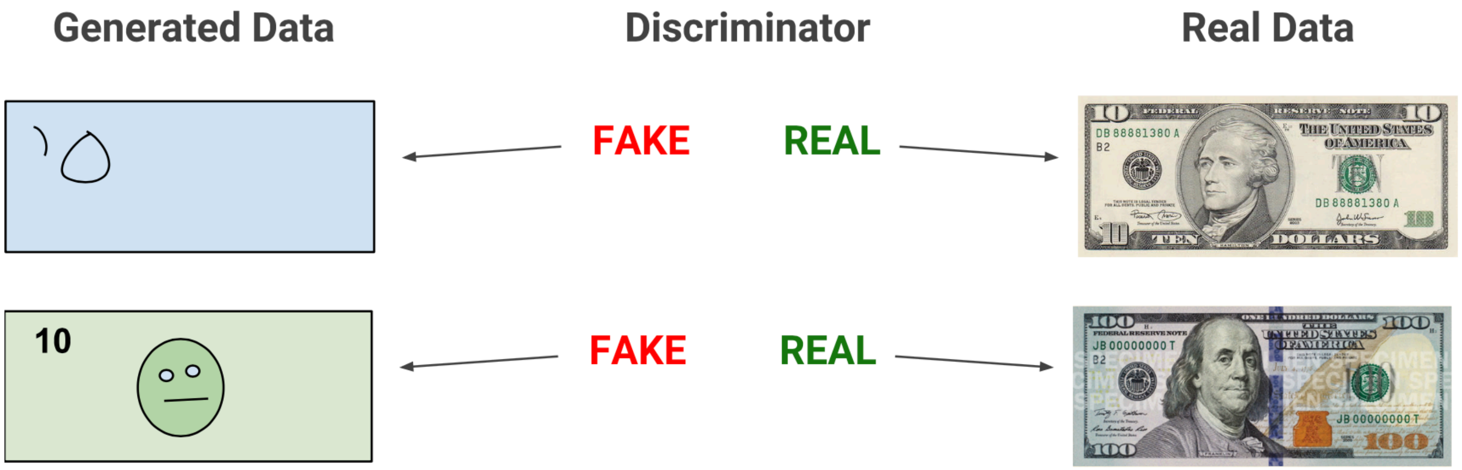
Sursa: https://developers.google.com/machine-learning/gan/generative

GANs - Intuitie



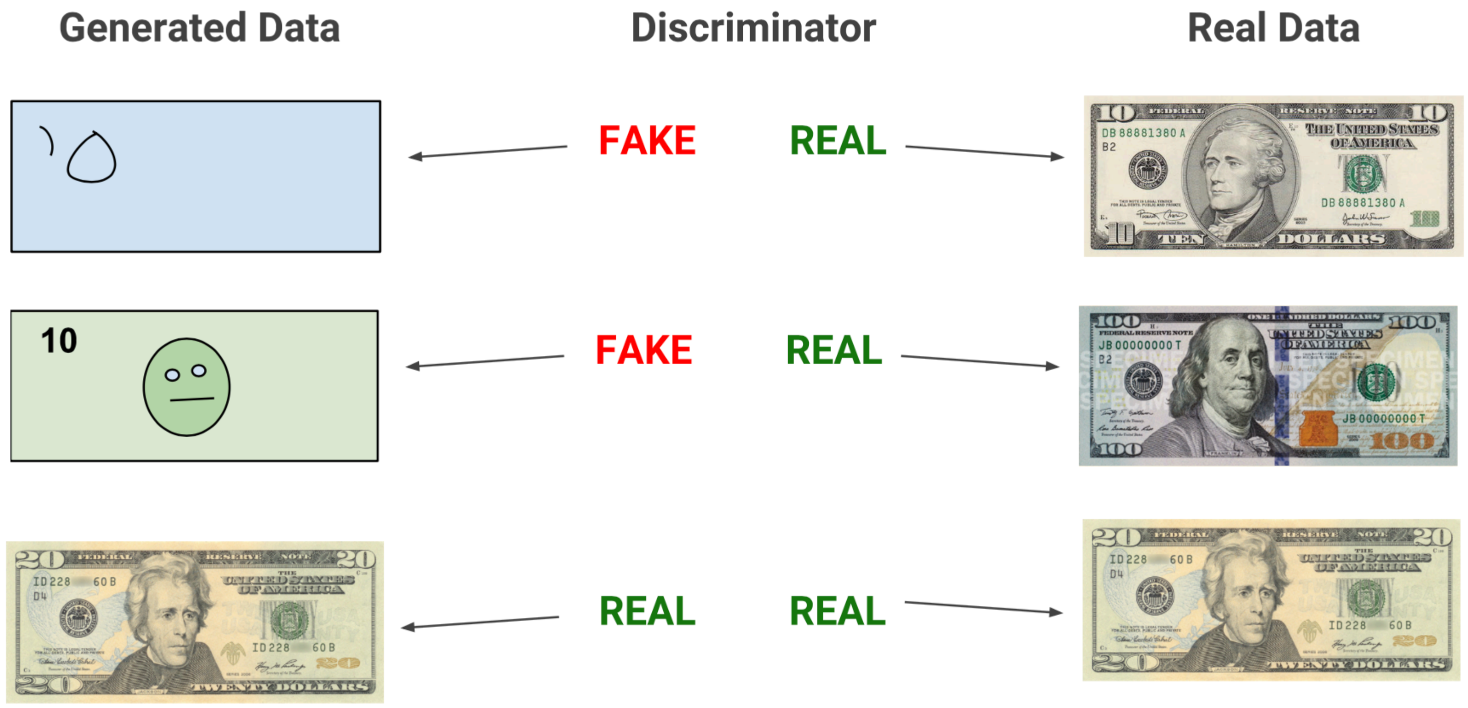
Sursa: https://developers.google.com/machine-learning/gan/gan_structure

GANs - Intuitie



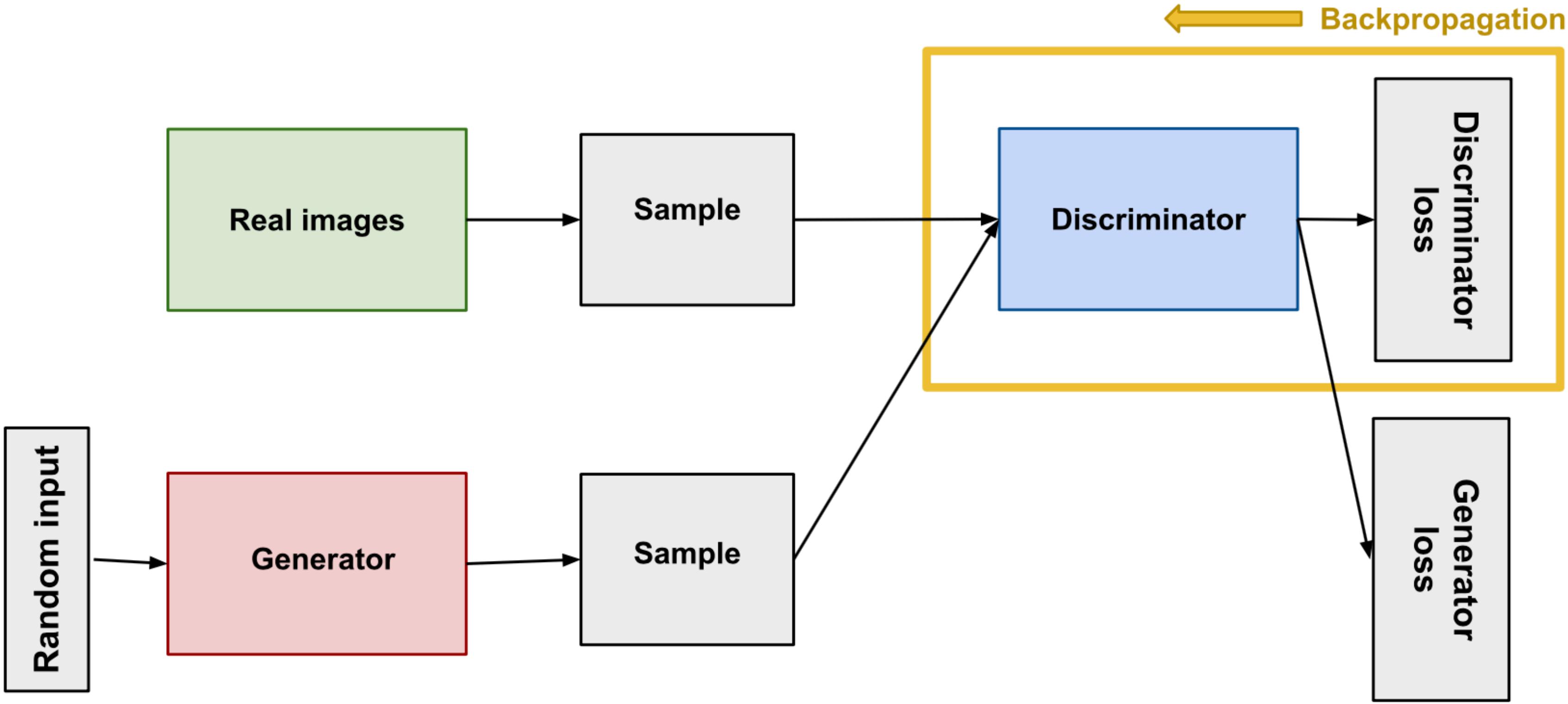
Sursa: https://developers.google.com/machine-learning/gan/gan_structure

GANs - Intuitie



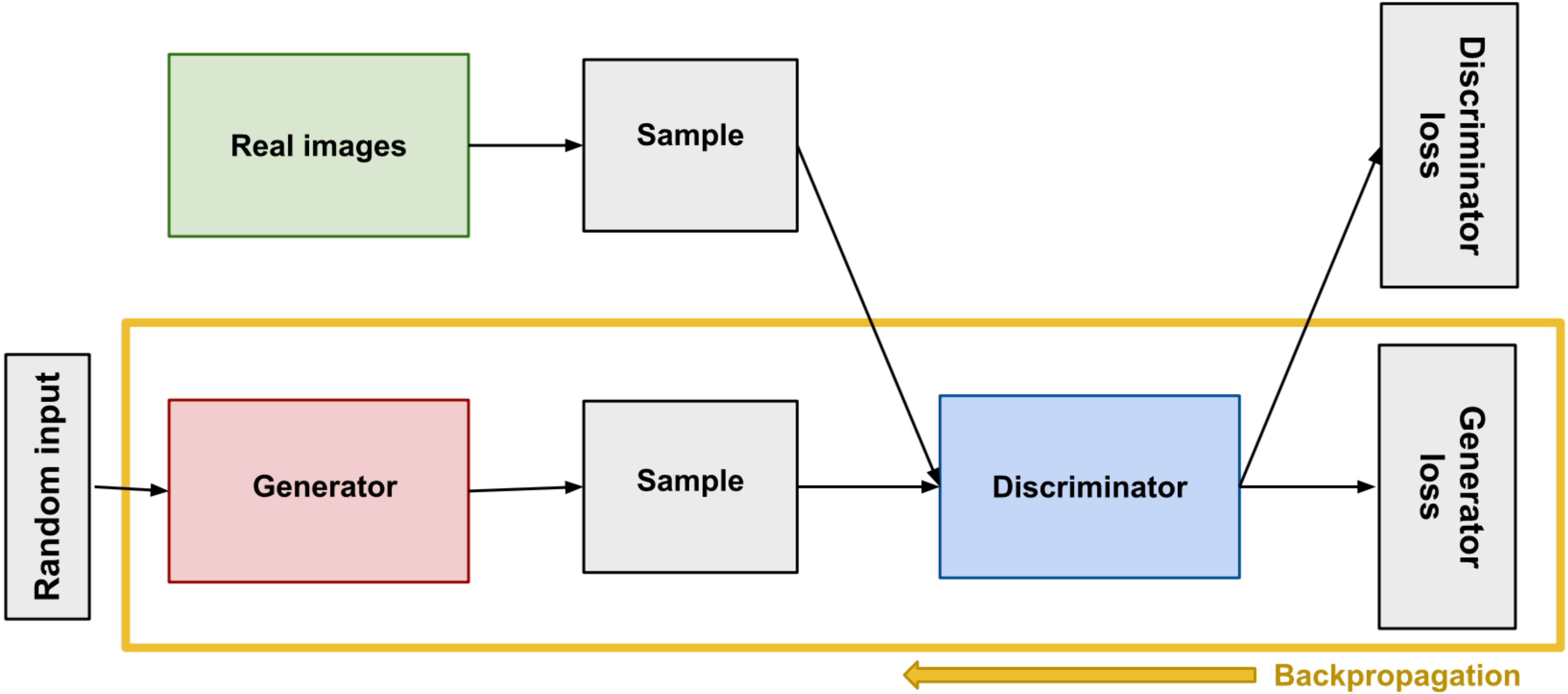
Sursa: https://developers.google.com/machine-learning/gan/gan_structure

GANs - Antrenare discriminator



Sursa: https://developers.google.com/machine-learning/gan/discriminator

GANs - Antrenare generator



Sursa: https://developers.google.com/machine-learning/gan/generator

GANs - Antrenare

- Antrenare alternativa intre generator si discriminator.
- Minmax loss: $E_x[log(D(x))] + E_z[log(1 - D(G(z)))]$
- Wasserstein GAN: generator-critic

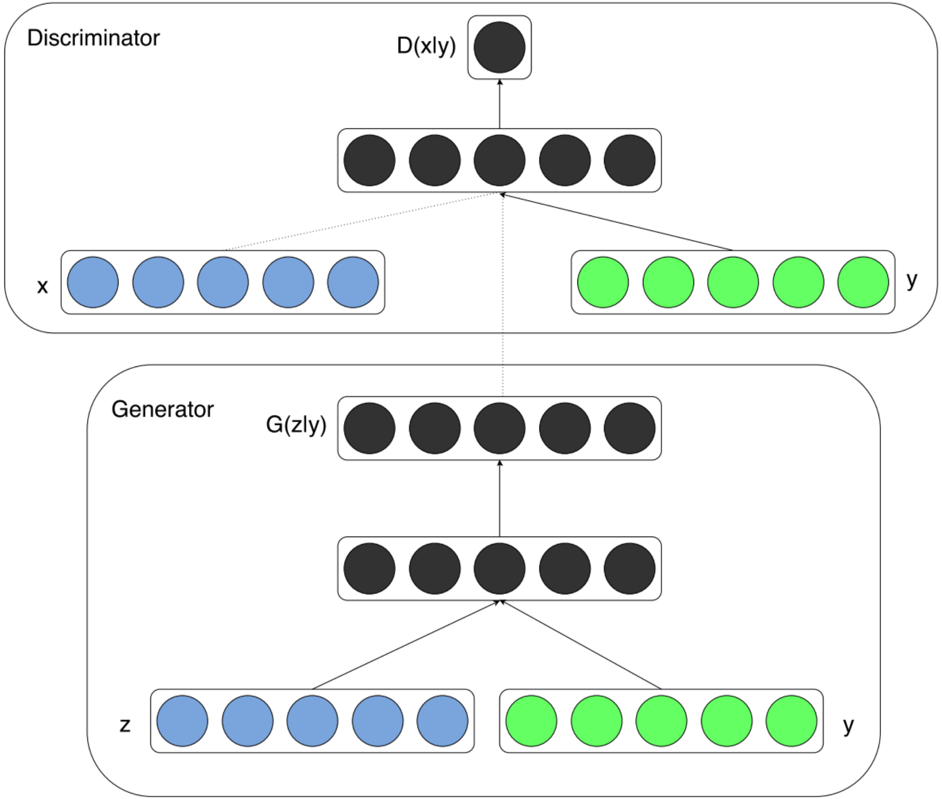
GANs - Probleme

- Convergenta este complicata
- Pe masura ce generatorul se imbunatateste, calitatea feedback-ul de la discriminator scade.
- Daca discriminatorul este prea bun => vanishing gradients.
- Mode Collapse - generatorul invata sa genereze mereu acelasi output.

GANs - Avantaje fata de VAE

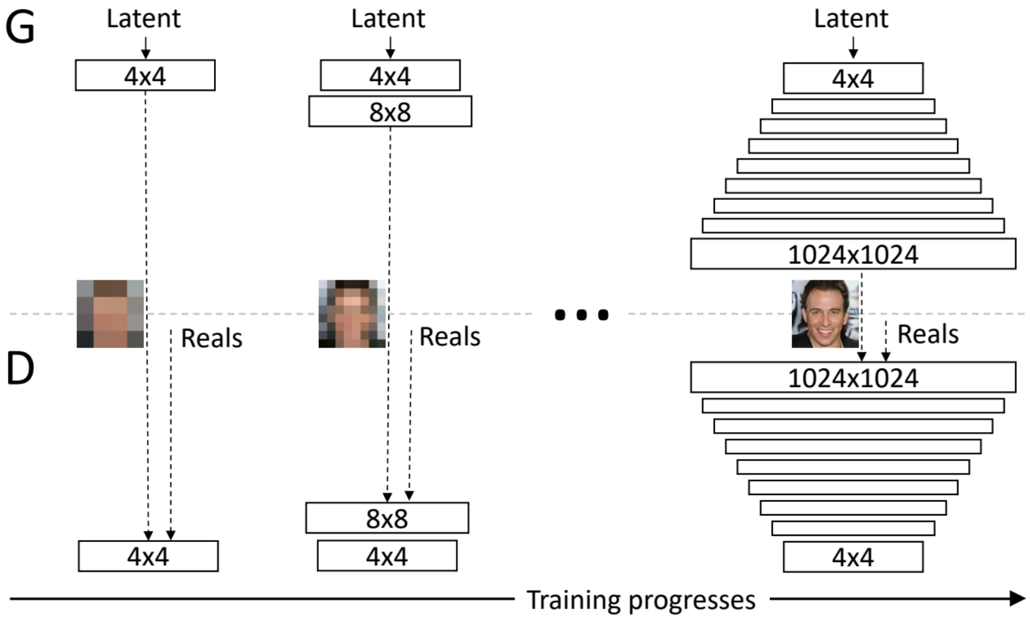
- GAN-urile pot genera imagini mai realiste si mai clare. Nu trebuie sa invete o aproximare continua a distributiei datelor.
- GAN-urile pot genera date mai diverse si mai complexe. Nu sunt limitate la o distributie normala a spatiului latent.

Conditional GANs



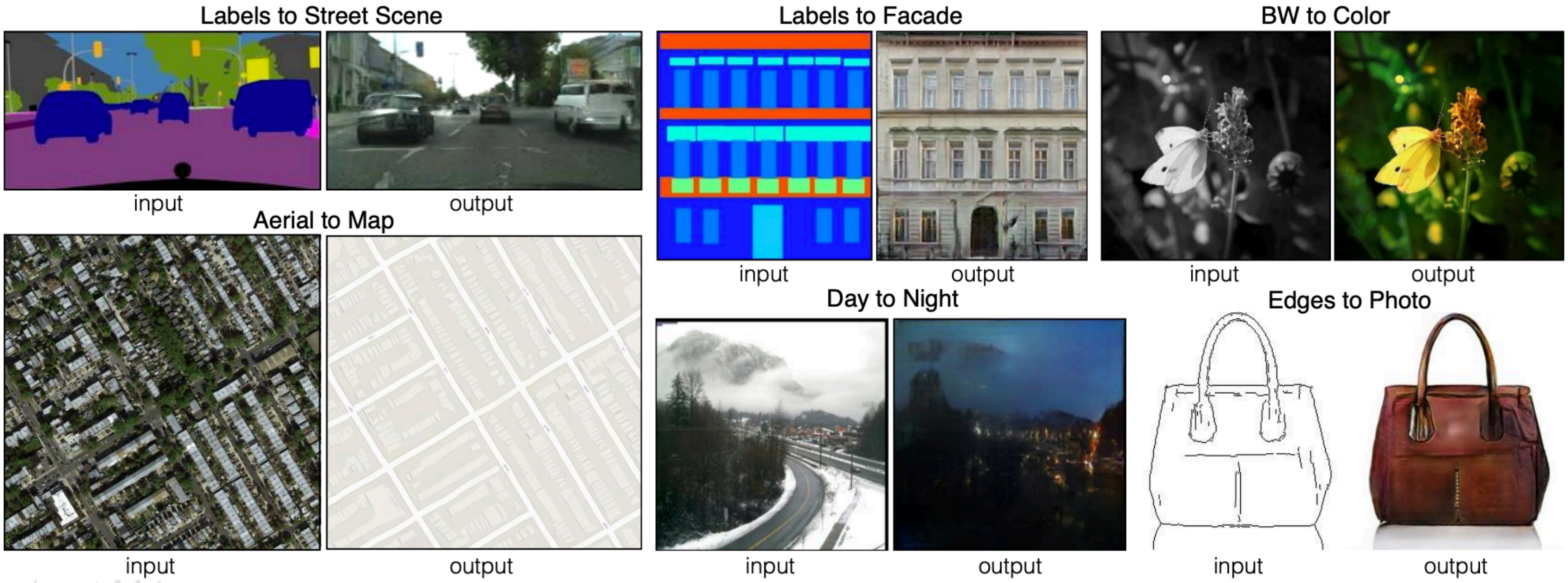
Sursa: Mirza, Mehdi, and Simon Osindero. "Conditional generative adversarial nets." arXiv preprint arXiv:1411.1784 (2014).

Progressive GANs



Sursa: Karras, Tero, et al. "Progressive growing of gans for improved quality, stability, and variation." arXiv preprint arXiv:1710.10196 (2017).

GANs - Image-to-Image Translation



Sursa: Isola, Phillip, et al. "Image-to-image translation with conditional adversarial networks." Proceedings of the IEEE conference on computer vision and pattern recognition. 2017.

GANs - Super-resolution

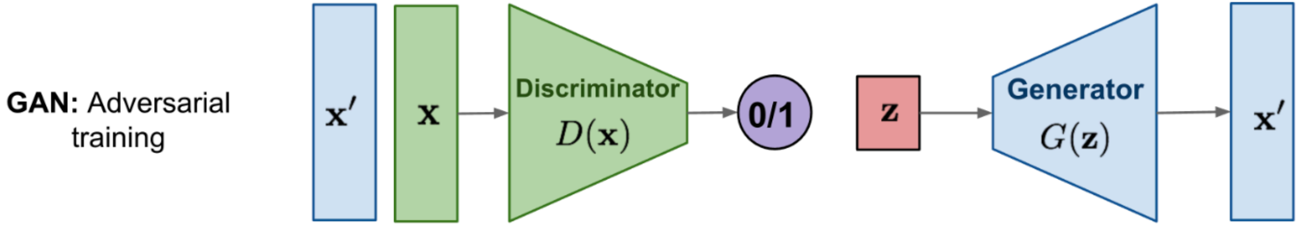
bicubic (21.59dB/0.6423) | SRResNet (23.53dB/0.7832) | SRGAN (21.15dB/0.6868) | original

Ledig, Christian, et al. "Photo-realistic single image super-resolution using a generative adversarial network." Proceedings of the IEEE conference on computer vision and pattern recognition. 2017.

GANs - this-person-does-not-exist.com

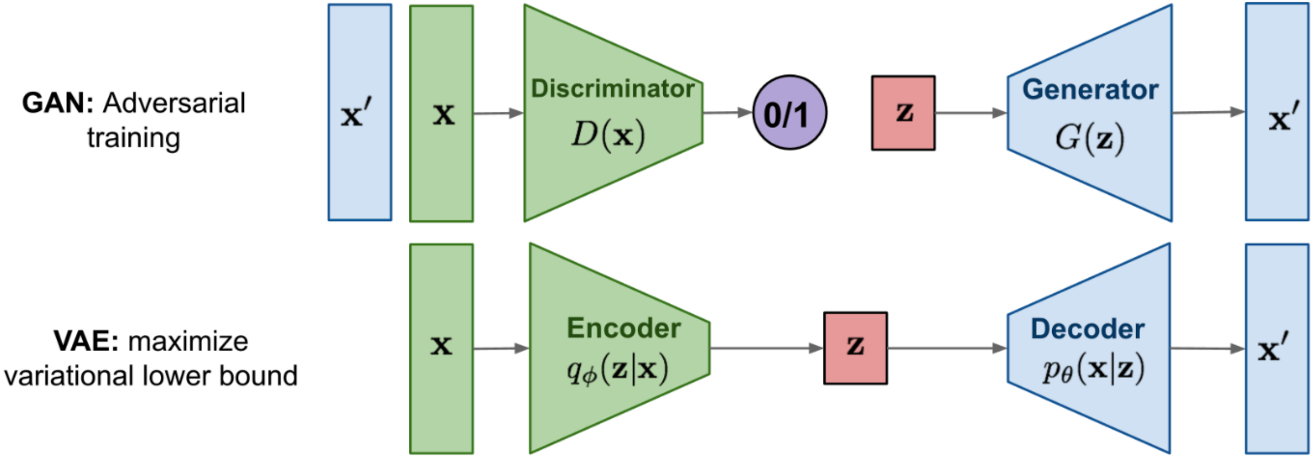
III. Diffusion Models

Diffusion Models - Overview



Sursa: https://lilianweng.github.io/posts/2021-07-11-diffusion-models/

Diffusion Models - Overview



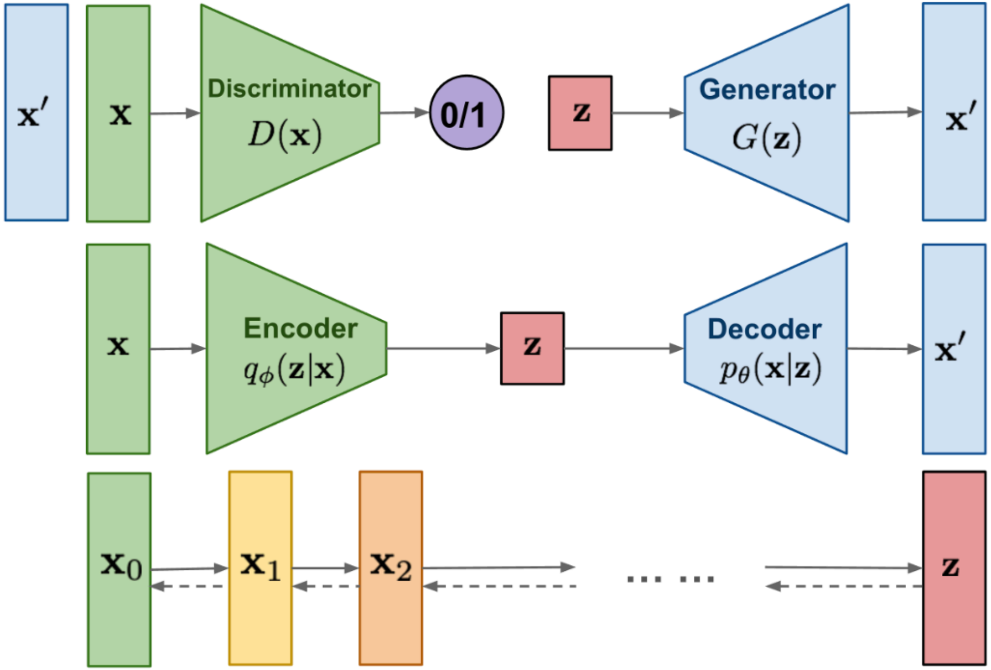
Sursa: https://lilianweng.github.io/posts/2021-07-11-diffusion-models/

Diffusion Models - Overview

GAN: Adversarial training

VAE: maximize variational lower bound

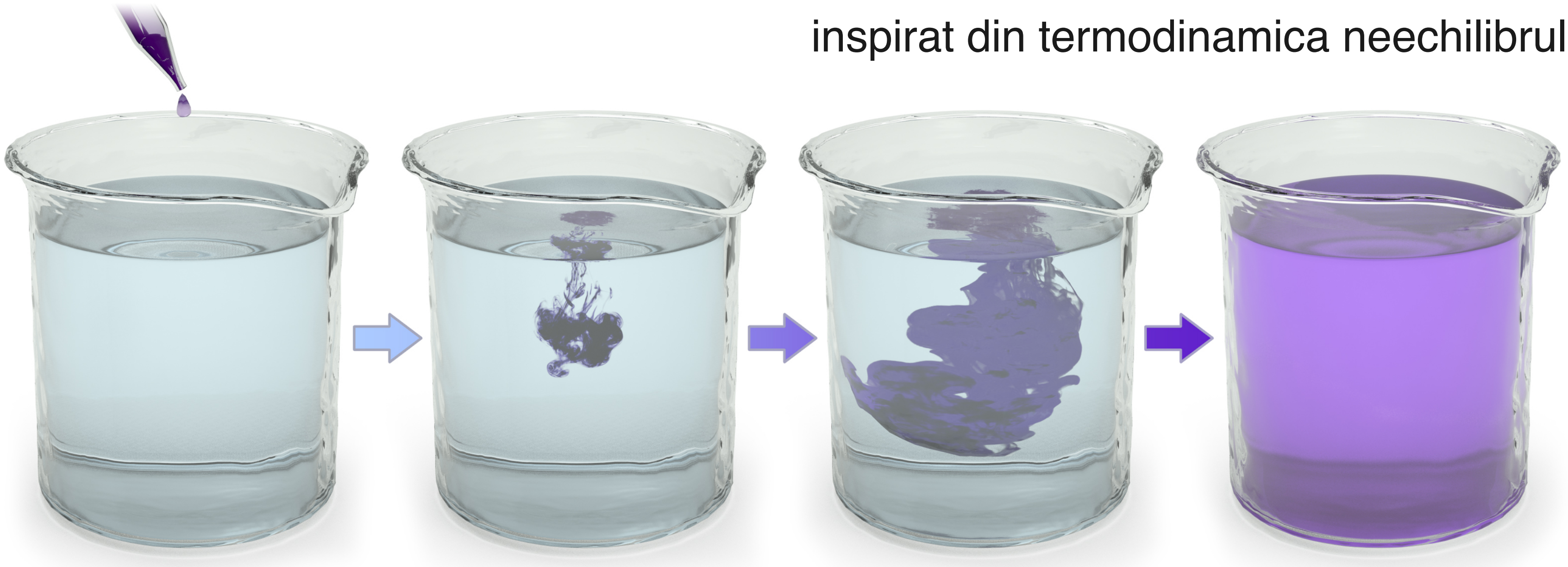
Diffusion models: Gradually add Gaussian noise and then reverse



Sursa: https://lilianweng.github.io/posts/2021-07-11-diffusion-models/

Diffusion Models

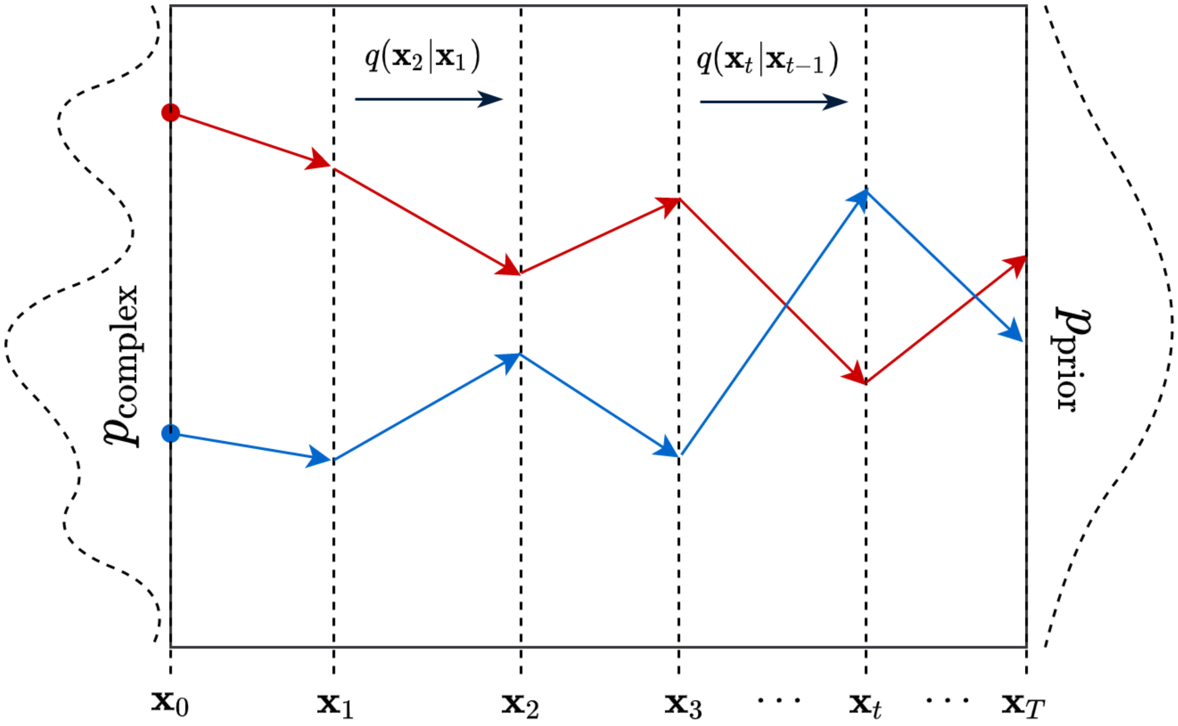
inspirat din termodinamica neechilibrului



Diffusion

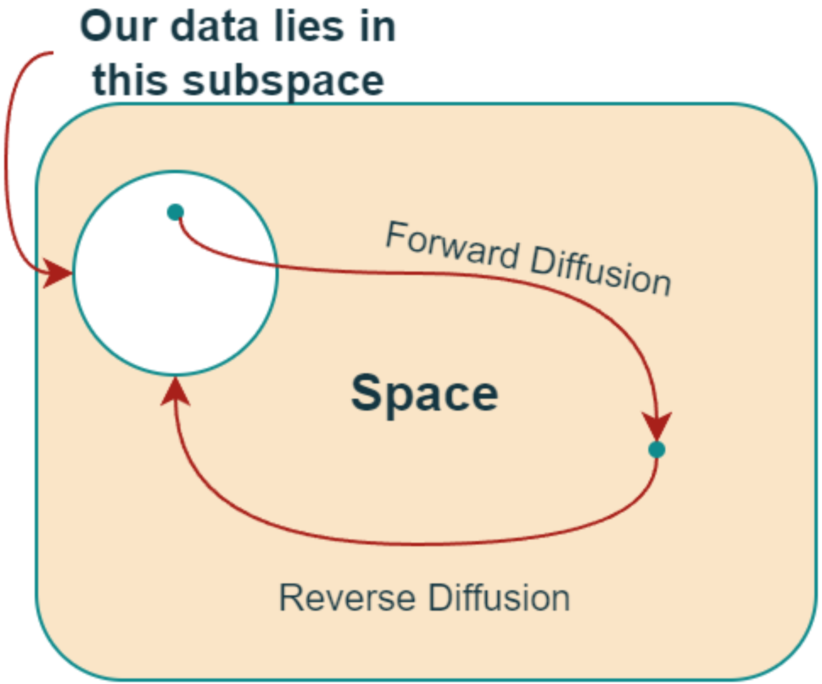
Sursa: https://commons.wikimedia.org/wiki/File:Blausen_0315_Diffusion.png#/media/Fișier:Blausen_0315_Diffusion.png

Diffusion Models



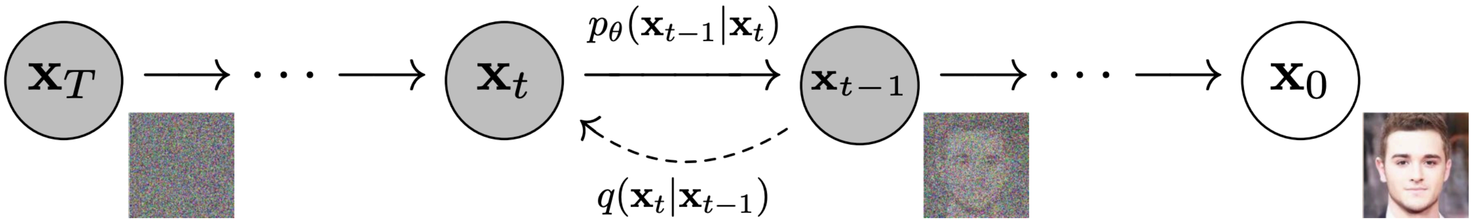
Sursa: https://ayandas.me/blog-tut/2021/12/04/diffusion-prob-models.html

Diffusion Models



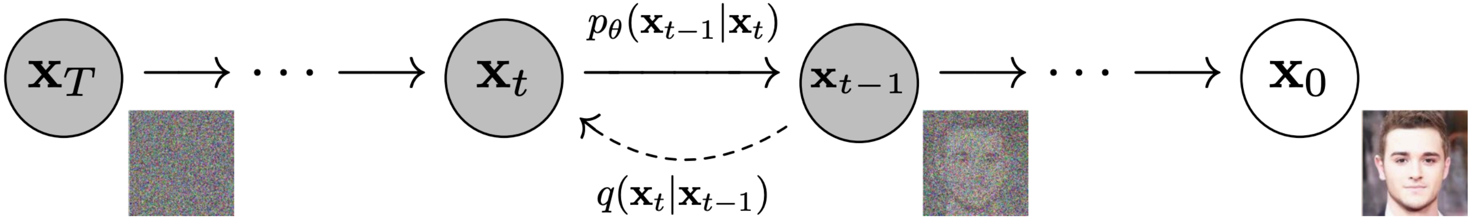
Sursa: https://learnopencv.com/denoising-diffusion-probabilistic-models/

Denoising Diffusion Probabilistic Models (DDPM)



Referinta: Ho, Jonathan, Ajay Jain, and Pieter Abbeel. "Denoising diffusion probabilistic models." *Advances in Neural Information Processing Systems* 33 (2020): 6840-6851.

Denoising Diffusion Probabilistic Models (DDPM)



Algorithm 1 Training

1: **repeat**
2: $\quad \mathbf{x}_0 \sim q(\mathbf{x}_0)$
3: $\quad t \sim \mathrm{Uniform}(\{1, \ldots, T\})$
4: $\quad \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
5: $\quad$ Take gradient descent step on

$$\nabla_\theta \left\| \boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta(\sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}, t) \right\|^2$$

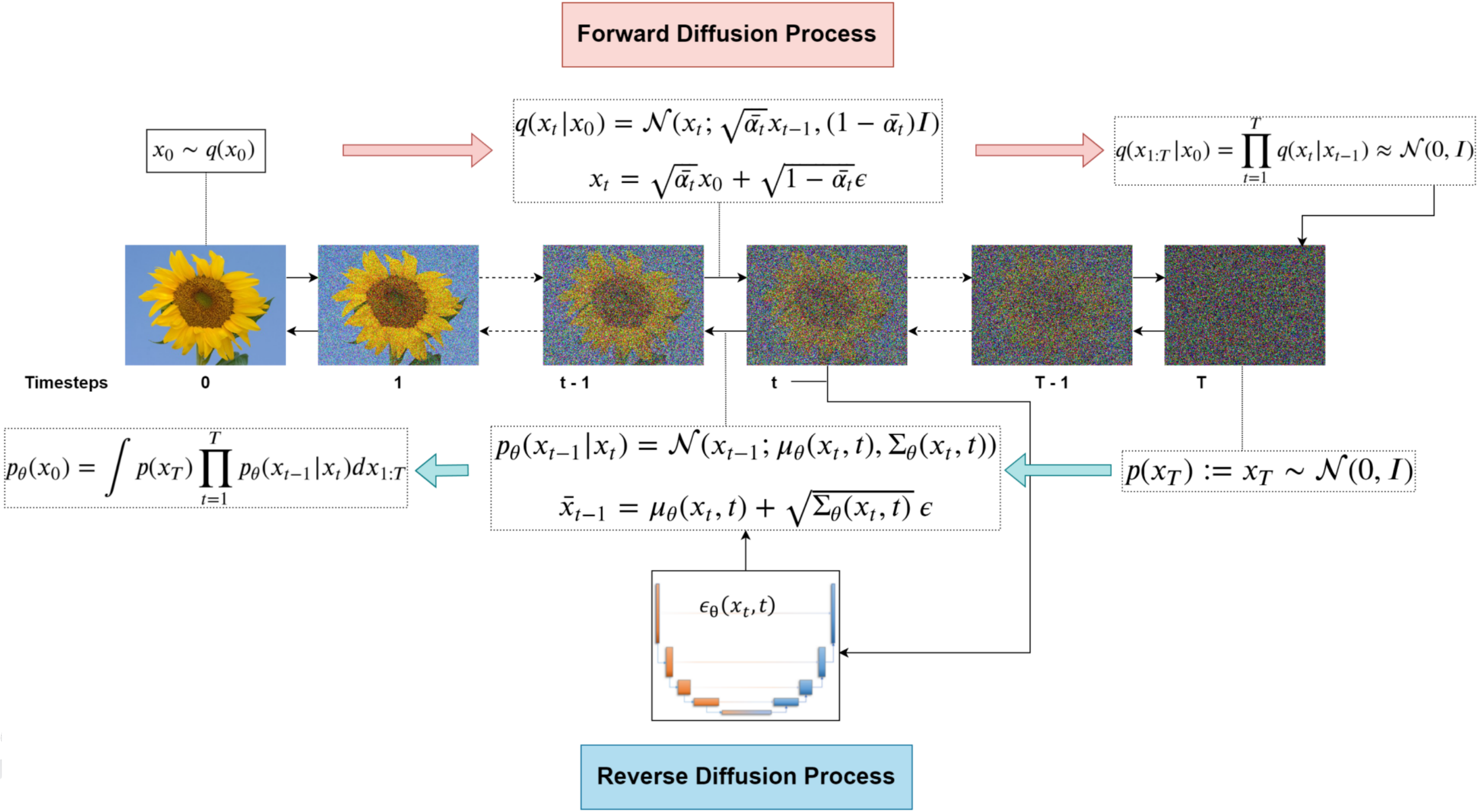
6: **until** converged

Algorithm 2 Sampling

1: $\mathbf{x}_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
2: **for** $t = T, \ldots, 1$ **do**
3: $\quad \mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ if $t > 1$, else $\mathbf{z} = \mathbf{0}$
4: $\quad \mathbf{x}_{t-1} = \frac{1}{\sqrt{\alpha_t}} \left(\mathbf{x}_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}} \boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t) \right) + \sigma_t \mathbf{z}$
5: **end for**
6: **return** $\mathbf{x}_0$

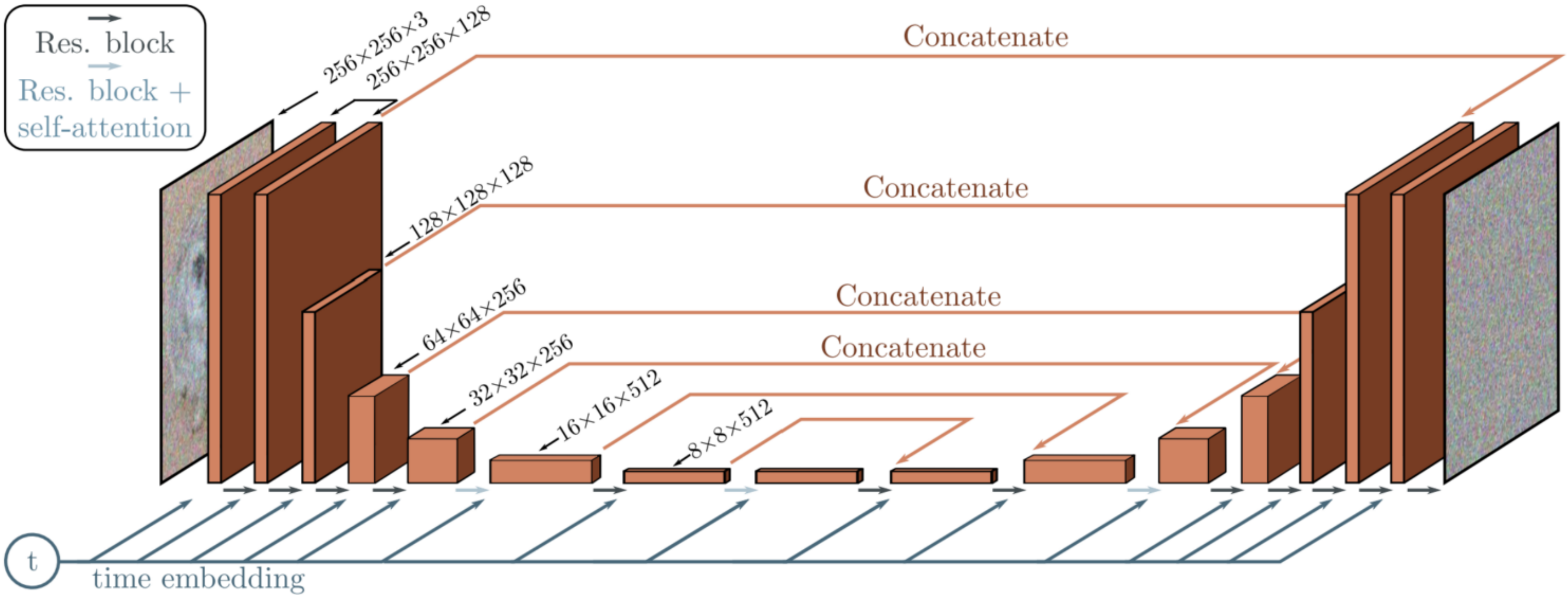
Referinta: Ho, Jonathan, Ajay Jain, and Pieter Abbeel. "Denoising diffusion probabilistic models." *Advances in Neural Information Processing Systems* 33 (2020): 6840-6851.

DDPM



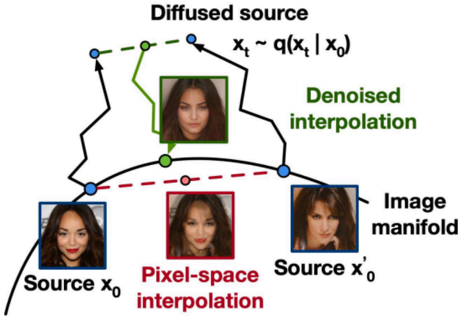
Sursa: https://learnopencv.com/denoising-diffusion-probabilistic-models/

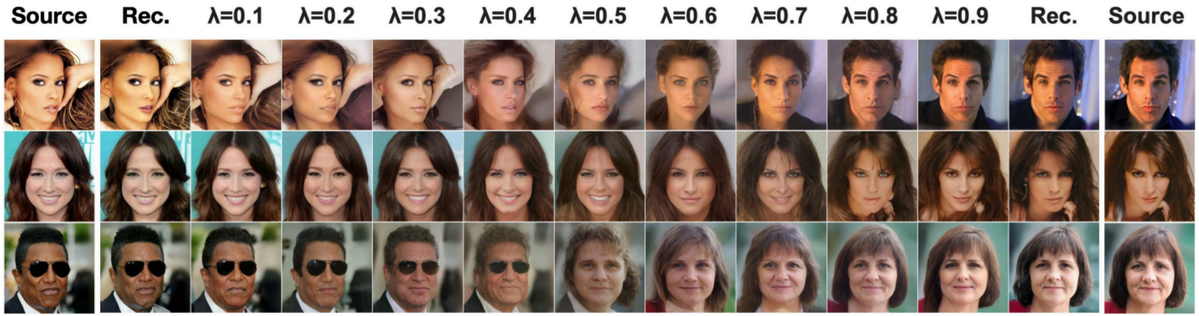
DDPM - U-Net conditionat de timp



Sursa: https://learnopencv.com/denoising-diffusion-probabilistic-models/

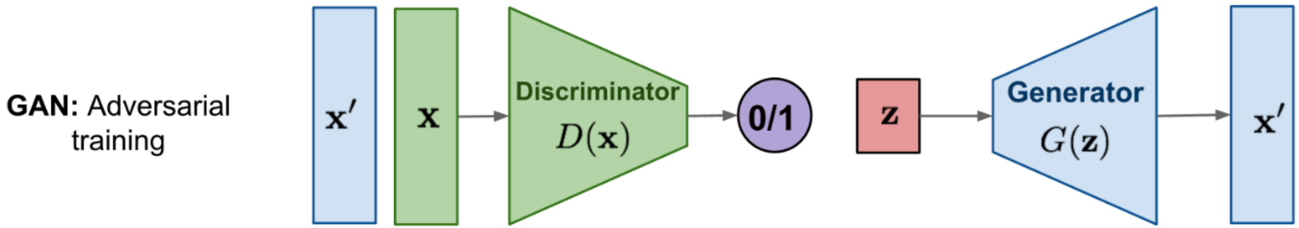
DDPM - Latent mixing



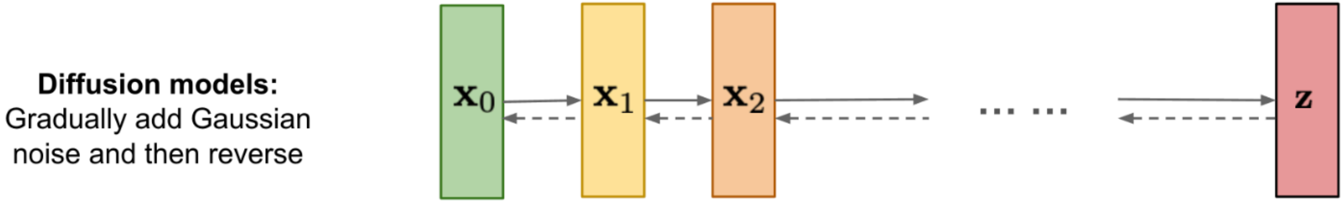


Referinta: Ho, Jonathan, Ajay Jain, and Pieter Abbeel. "Denoising diffusion probabilistic models." *Advances in Neural Information Processing Systems* 33 (2020): 6840-6851.

DDPM - Comparatie



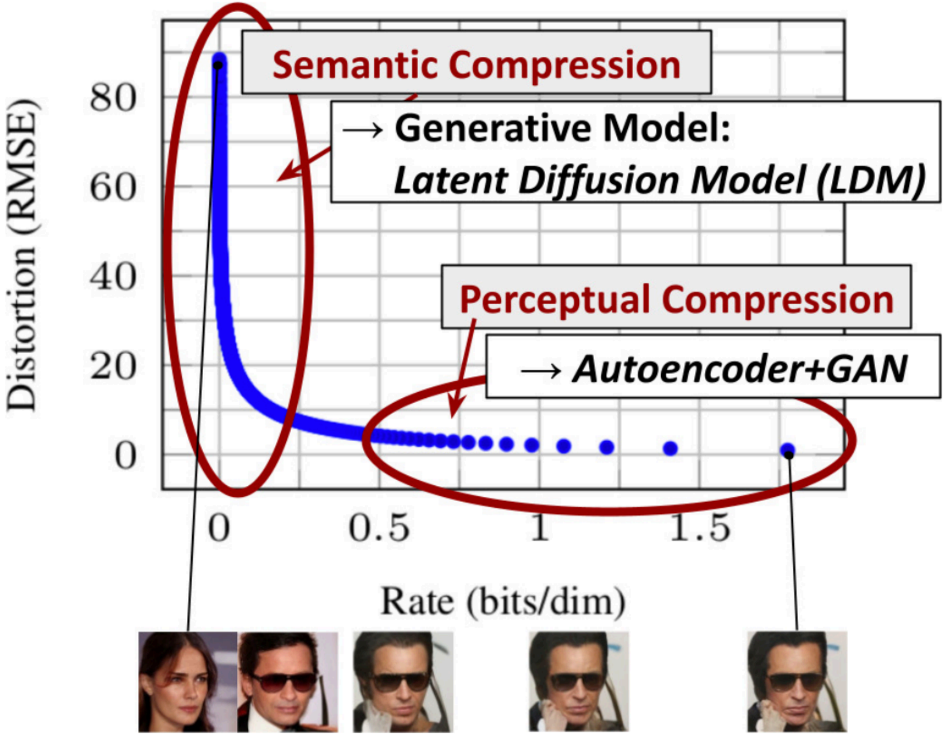
Mai usor de antrenat decat GAN-urile, datorita pasilor mici.



Inferenta mai lenta decat a GAN-urile.

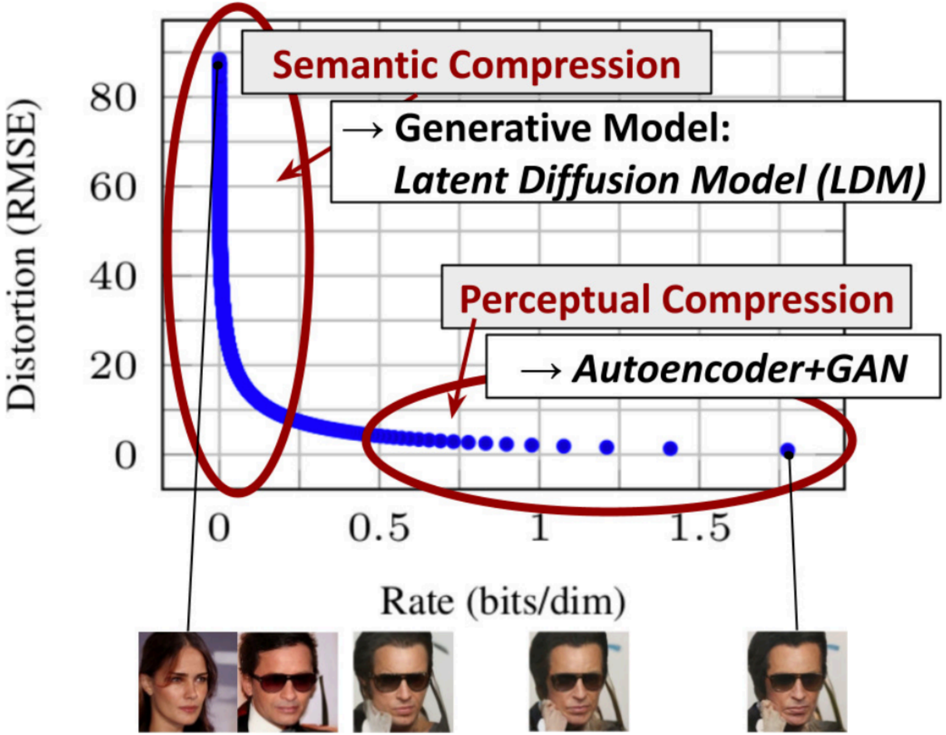
Sursa: https://lilianweng.github.io/posts/2021-07-11-diffusion-models/

Latent Diffusion Model (LDM) - Motivatie



Referinta: Rombach, Robin, et al. "High-resolution image synthesis with latent diffusion models." *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2022.

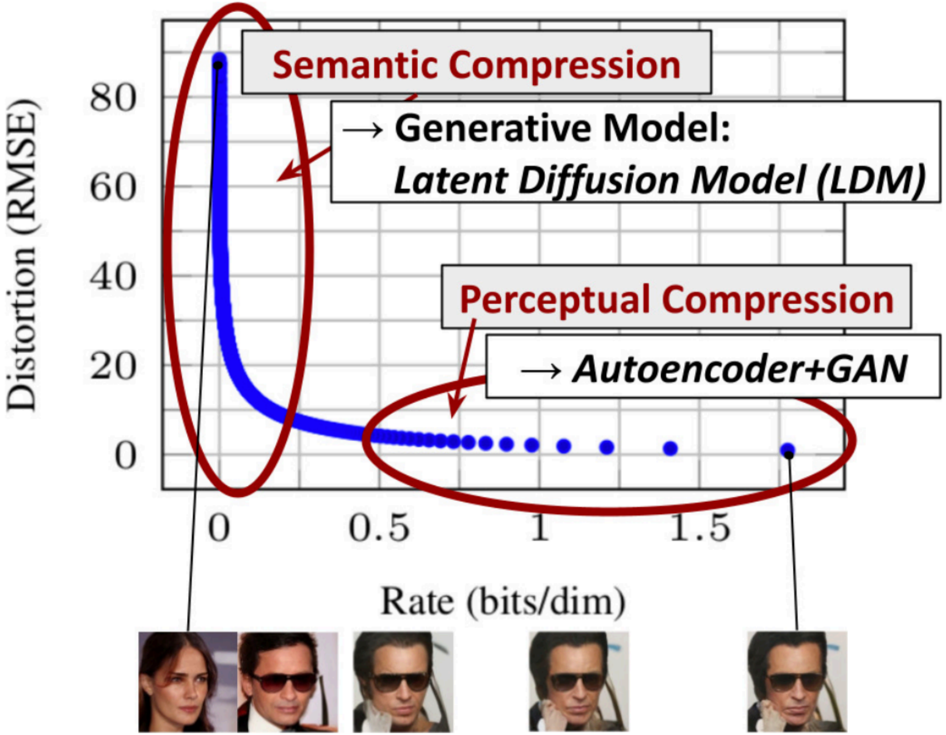
Latent Diffusion Model (LDM) - Motivatie



Idee: procesul de difuzie are loc in spatiul latent

Referinta: Rombach, Robin, et al. "High-resolution image synthesis with latent diffusion models." *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2022.

Latent Diffusion Model (LDM) - Motivatie

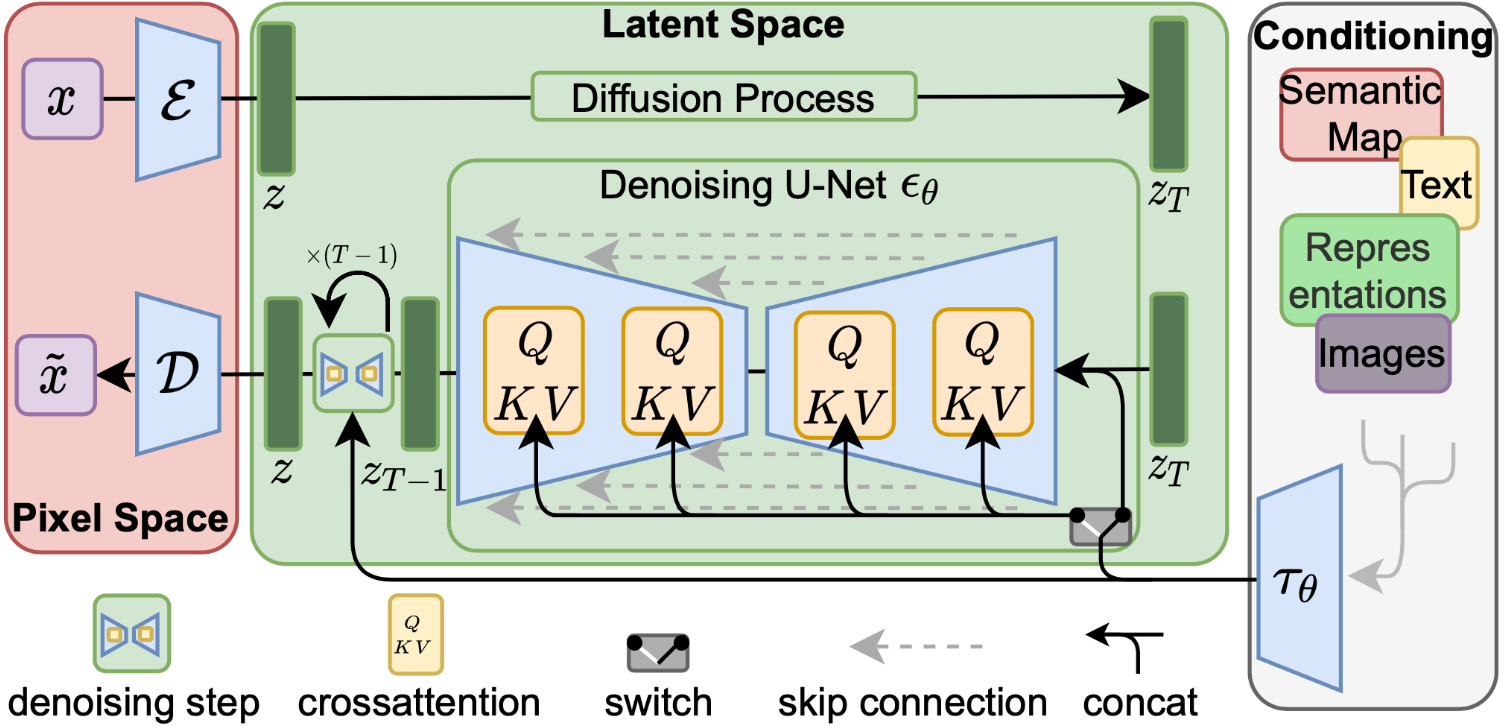


Idee: procesul de difuzie are loc in spatiul latent

Folsim un autoencoder pentru compresie

Referinta: Rombach, Robin, et al. "High-resolution image synthesis with latent diffusion models." *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2022.

Latent Diffusion Model (LDM)



Referinta: Rombach, Robin, et al. "High-resolution image synthesis with latent diffusion models." *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2022.

LDM - Exemple

'A street sign that reads "Latent Diffusion" ' | 'A zombie in the style of Picasso' | 'An image of an animal half mouse half octopus' | 'An illustration of a slightly conscious neural network' | 'A painting of a squirrel eating a burger' | 'A watercolor painting of a chair that looks like an octopus' | 'A shirt with the inscription: "I love generative models!" '

Referinta: Rombach, Robin, et al. "High-resolution image synthesis with latent diffusion models." *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2022.

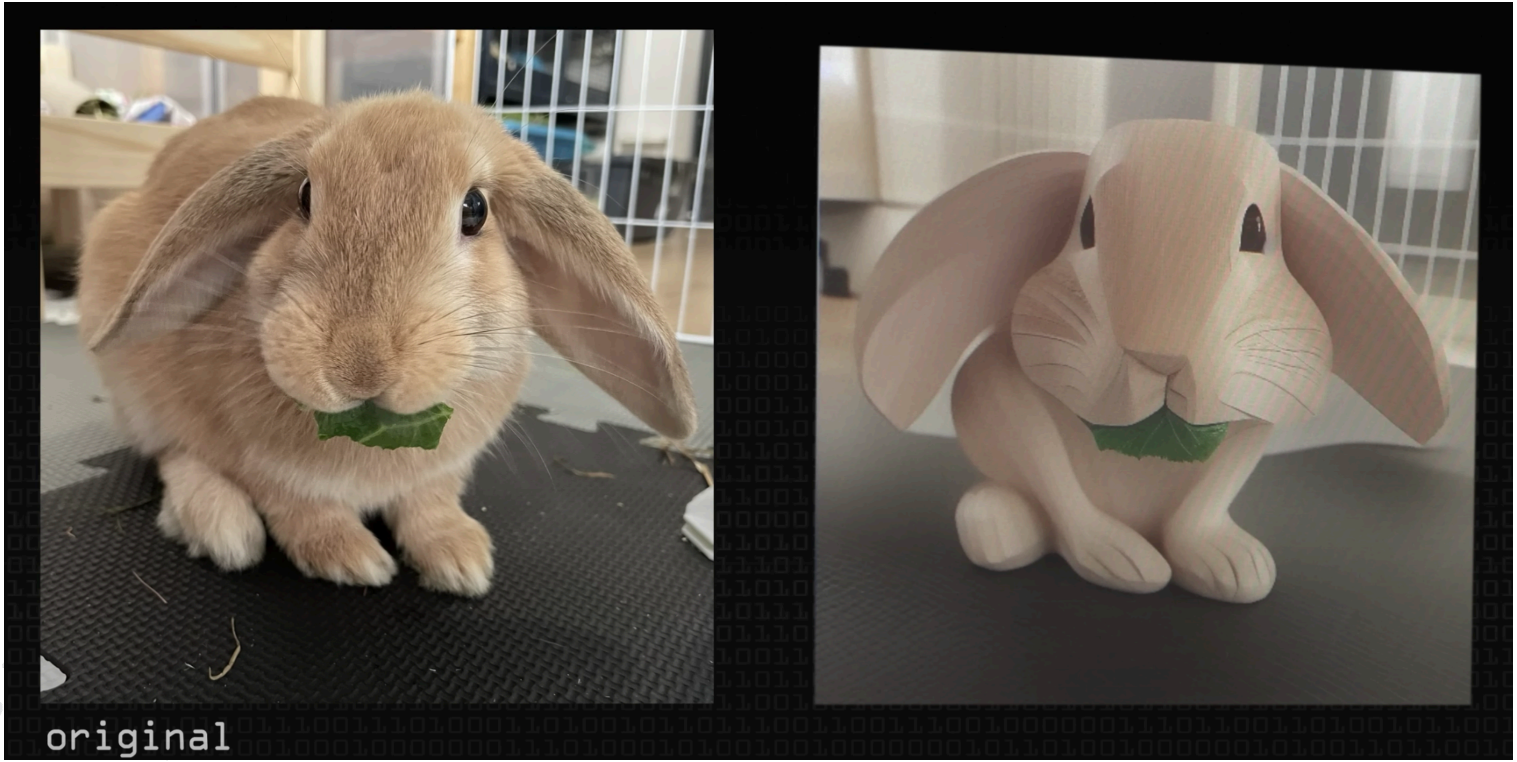
Classifier-Free Guidance

Amplificam diferenta dintre

$\epsilon_\theta(\mathbf{x}_t, t) = \epsilon_\theta(\mathbf{x}_t, t, y = \emptyset)$ si $\epsilon_\theta(\mathbf{x}_t, t, y)$

Referinta: Ho, Jonathan, and Tim Salimans. "Classifier-free diffusion guidance." arXiv preprint arXiv:2207.12598 (2022).

Diffusion Models - Image to image



Sursa: [Stable Diffusion in Code (AI Image Generation) - Computerphile]

Stable Diffusion V2

Sursa: https://github.com/Stability-AI/stablediffusion

Stable Diffusion V2

Cat de rapid ne-am fi putut imagina aceste imagini?

Sursa: https://github.com/Stability-AI/stablediffusion

Stable Diffusion - Demo

https://colab.research.google.com/drive/1roZqqhsdpCXZr8kgV_Bx_ABVBPgea3lX
by Jonathan Whitaker

Diffusion Models

DALL-E 2

Imagen

Stable Diffusion

Midjourney