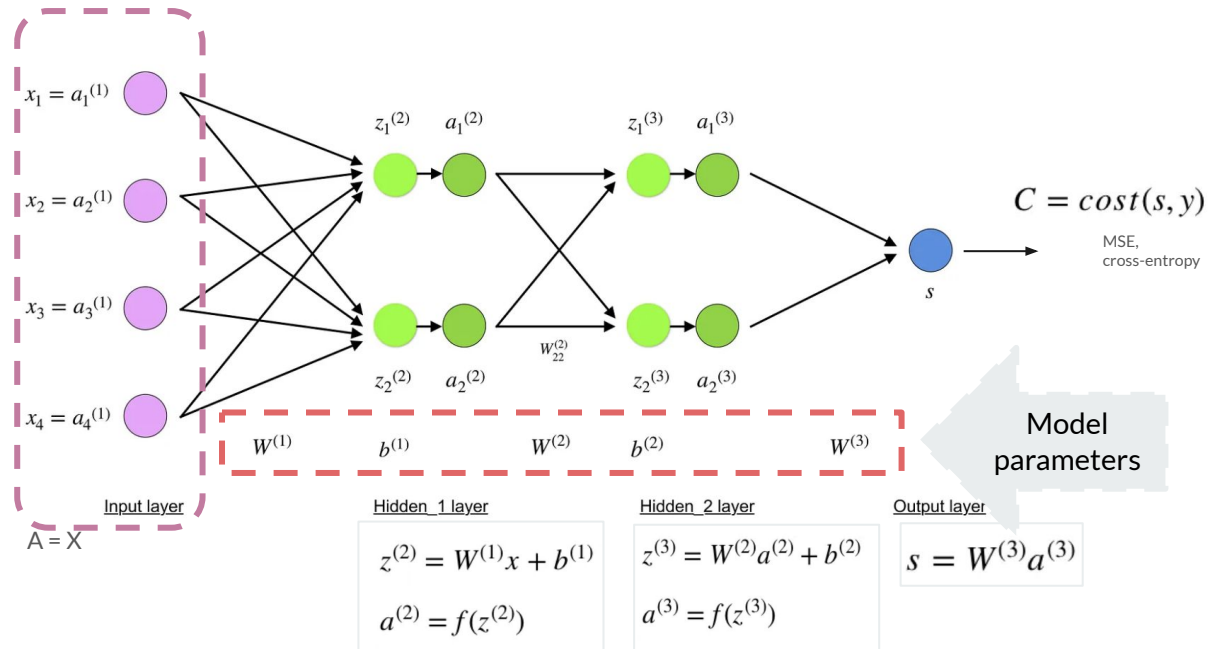# CIS 678 Machine Learning

Introduction to Neural Networks (cont.)

# Popular Neural Network (NN) Architectures
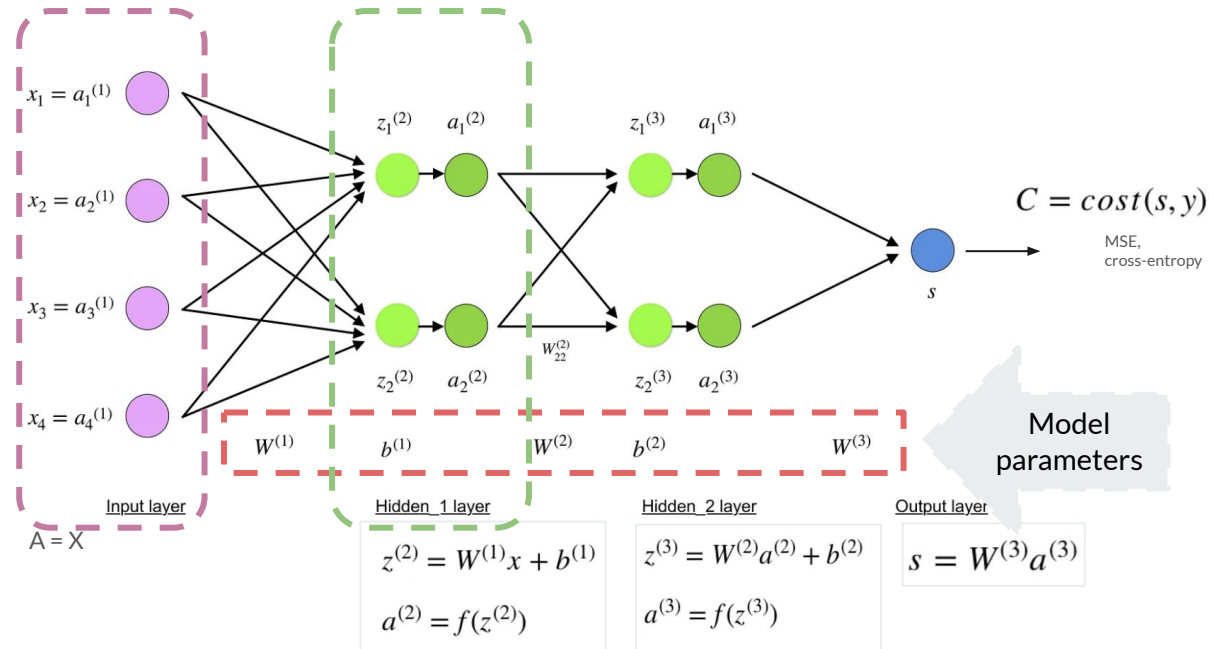
- Feed Forward NNs
- Convolutional NNs
- Sequence models
    - LSTM
    - Transformers
- Generative AI
    - GANs
    - VAEs
- Graph Neural Networks

# Feed forward NNs



$C = cost(s, y)$

MSE, cross-entropy

Model parameters

$A = X$

Hidden_1 layer
$$z^{(2)} = W^{(1)}x + b^{(1)}$$
$$a^{(2)} = f(z^{(2)})$$

Hidden_2 layer
$$z^{(3)} = W^{(2)}a^{(2)} + b^{(2)}$$
$$a^{(3)} = f(z^{(3)})$$

Output layer
$$s = W^{(3)}a^{(3)}$$

mainly adapted from

# Feed forward NNs

$x_1 = a_1^{(1)}$

$x_2 = a_2^{(1)}$

$x_3 = a_3^{(1)}$

$x_4 = a_4^{(1)}$

$z_1^{(2)}$  $a_1^{(2)}$

$z_2^{(2)}$  $a_2^{(2)}$

$z_1^{(3)}$  $a_1^{(3)}$

$z_2^{(3)}$  $a_2^{(3)}$

$W_{22}^{(2)}$

$C = cost(s, y)$

MSE,
cross-entropy

$s$

Model parameters

$W^{(1)}$  $b^{(1)}$  $W^{(2)}$  $b^{(2)}$  $W^{(3)}$

Input layer

A = X

Hidden_1 layer

Hidden_2 layer

Output layer

$z^{(2)} = W^{(1)}x + b^{(1)}$

$a^{(2)} = f(z^{(2)})$

$z^{(3)} = W^{(2)}a^{(2)} + b^{(2)}$

$a^{(3)} = f(z^{(3)})$

$s = W^{(3)}a^{(3)}$

mainly adapted from

# Feed forward NNs

$x_1 = a_1^{(1)}$

$x_2 = a_2^{(1)}$

$x_3 = a_3^{(1)}$

$x_4 = a_4^{(1)}$

$z_1^{(2)} \quad a_1^{(2)}$

$z_2^{(2)} \quad a_2^{(2)}$

$W_{22}^{(2)}$

$W^{(1)} \quad b^{(1)} \quad W^{(2)}$

Input layer

Hidden_1 layer

A = X

$z^{(2)} = W^{(1)}x + b^{(1)}$

$a^{(2)} = f(z^{(2)})$
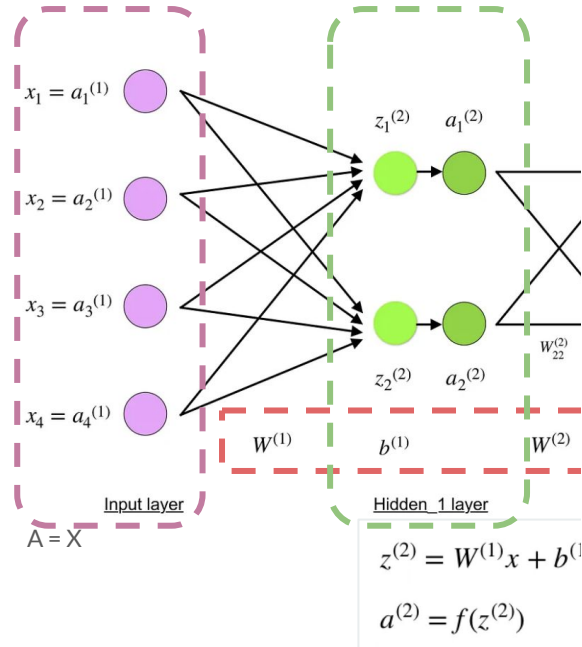
mainly adapted from

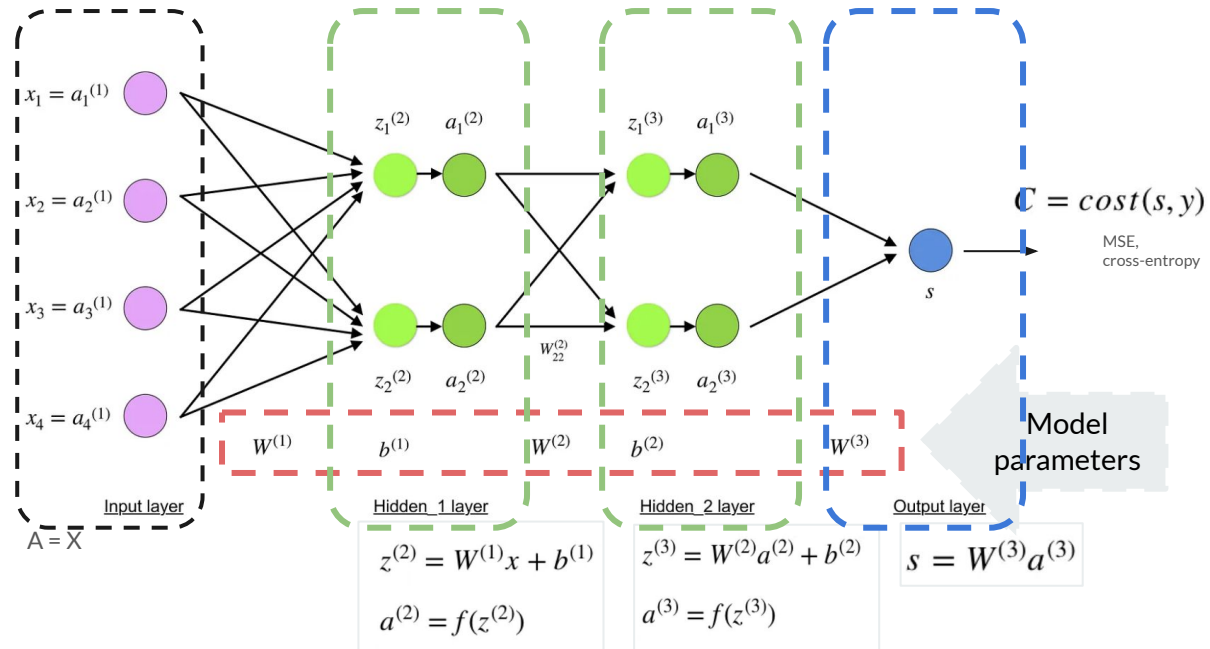| Name | Plot | Function, $g(x)$ |
|---|---|---|
| Identity | | $x$ |
| Binary step | | $\begin{cases} 0 & \text{if } x < 0 \\ 1 & \text{if } x \geq 0 \end{cases}$ |
| Logistic, sigmoid, or soft step | | $\sigma(x) \doteq \dfrac{1}{1 + e^{-x}}$ |
| Hyperbolic tangent (tanh) | | $\tanh(x) \doteq \dfrac{e^x - e^{-x}}{e^x + e^{-x}}$ |
| Soboleva modified hyperbolic tangent (smht) | | $\text{smht}(x) \doteq \dfrac{e^{ax} - e^{-bx}}{e^{cx} + e^{-dx}}$ |
| Rectified linear unit (ReLU)[8] | | $(x)^+ \doteq \begin{cases} 0 & \text{if } x \leq 0 \\ x & \text{if } x > 0 \end{cases}$ $= \max(0, x) = x\mathbf{1}_{x>0}$ |
| Gaussian Error Linear Unit (GELU)[2] | | $\dfrac{1}{2}x\left(1 + \text{erf}\left(\dfrac{x}{\sqrt{2}}\right)\right)$ $= x\Phi(x)$ |
| Softplus[9] | | $\ln(1 + e^x)$ |
| Exponential linear unit (ELU)[10] | | $\begin{cases} \alpha\left(e^x - 1\right) & \text{if } x \leq 0 \\ x & \text{if } x > 0 \end{cases}$ with parameter $\alpha$ |

# Feed forward NNs



$x_1 = a_1^{(1)}$

$x_2 = a_2^{(1)}$

$x_3 = a_3^{(1)}$

$x_4 = a_4^{(1)}$

$z_1^{(2)}$ $a_1^{(2)}$

$z_2^{(2)}$ $a_2^{(2)}$

$z_1^{(3)}$ $a_1^{(3)}$

$z_2^{(3)}$ $a_2^{(3)}$

$W_{22}^{(2)}$

$C = cost(s, y)$

MSE,
cross-entropy

$s$

Model
parameters

$W^{(1)}$ $b^{(1)}$ $W^{(2)}$ $b^{(2)}$ $W^{(3)}$

Input layer

A = X

Hidden_1 layer

Hidden_2 layer

Output layer

$$z^{(2)} = W^{(1)}x + b^{(1)}$$

$$a^{(2)} = f(z^{(2)})$$

$$z^{(3)} = W^{(2)}a^{(2)} + b^{(2)}$$

$$a^{(3)} = f(z^{(3)})$$

$$s = W^{(3)}a^{(3)}$$

mainly adapted from

# Feed forward NNs

$x_1 = a_1^{(1)}$

$x_2 = a_2^{(1)}$

$x_3 = a_3^{(1)}$

$x_4 = a_4^{(1)}$

$z_1^{(2)}$  $a_1^{(2)}$

$z_2^{(2)}$  $a_2^{(2)}$

$z_1^{(3)}$  $a_1^{(3)}$

$z_2^{(3)}$  $a_2^{(3)}$

$W_{22}^{(2)}$

$C = cost(s, y)$

MSE, cross-entropy

$s$

$W^{(1)}$   $b^{(1)}$   $W^{(2)}$   $b^{(2)}$   $W^{(3)}$

Model parameters

Input layer

A = X

Hidden_1 layer

Hidden_2 layer

Output layer

$$z^{(2)} = W^{(1)}x + b^{(1)}$$

$$a^{(2)} = f(z^{(2)})$$

$$z^{(3)} = W^{(2)}a^{(2)} + b^{(2)}$$

$$a^{(3)} = f(z^{(3)})$$

$$s = W^{(3)}a^{(3)}$$

mainly adapted from

# Convolutional NNs

Examples:

- Alexnet
- VGG
- ResNet
- GoogLeNet
- ..

$$(f * g)(t) := \int_{-\infty}^{\infty} f(\tau)g(t-\tau)\,d\tau.$$

ref



Image source
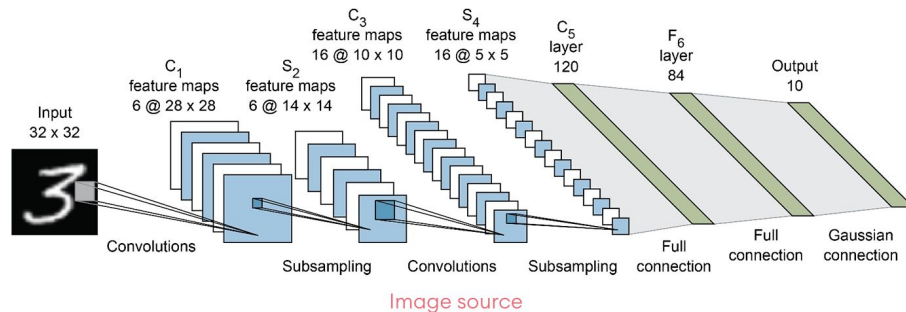
## The convolutional layer



Image patch
(Local receptive field)

Kernel
(filter)

Output

Input
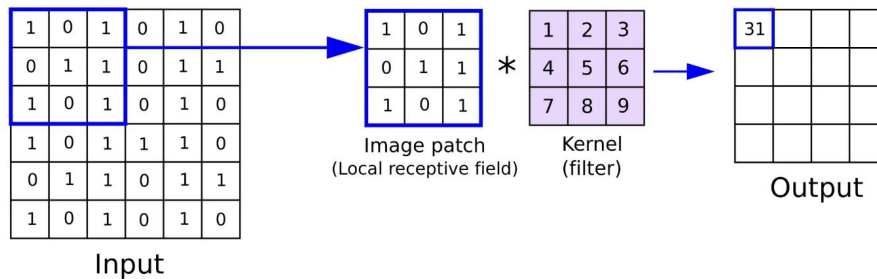
# Recurrent Neural Networks

Examples:

- LSTMs
- GRU
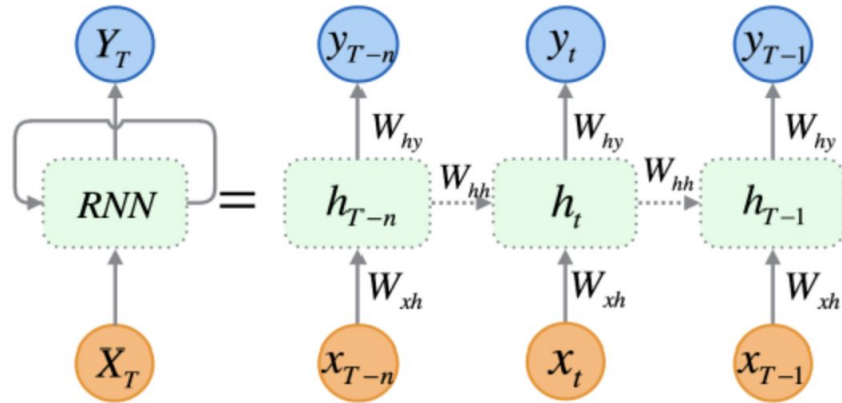


ref

# Transformers

Examples:

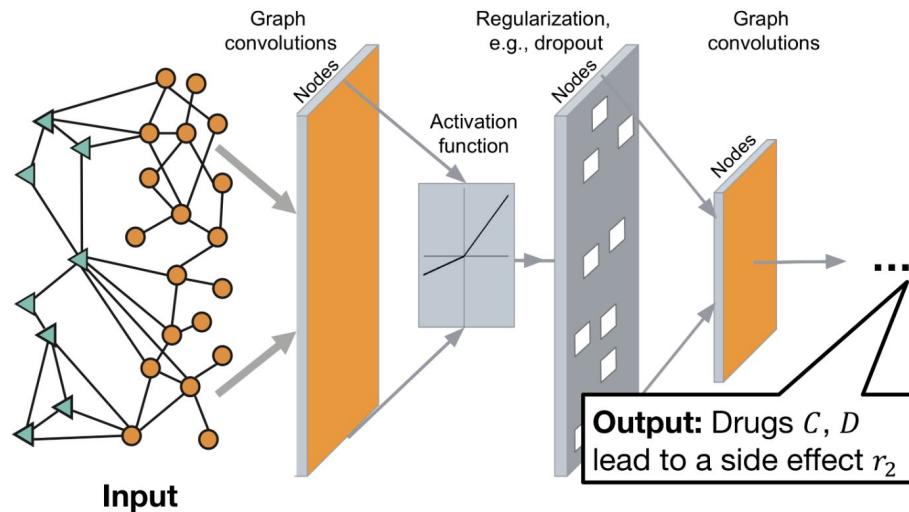- Encoder decoder pair
- GPT
- BERT

**BERT**

Encoder

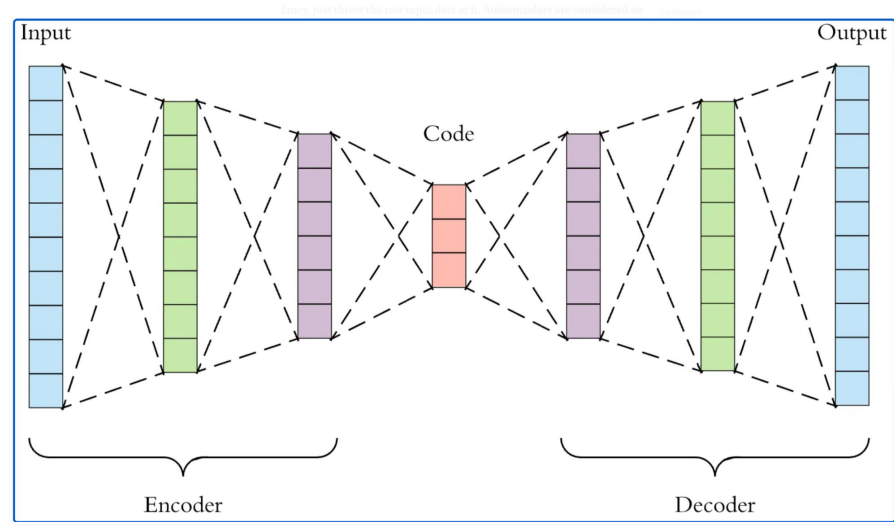**GPT**

Decoder
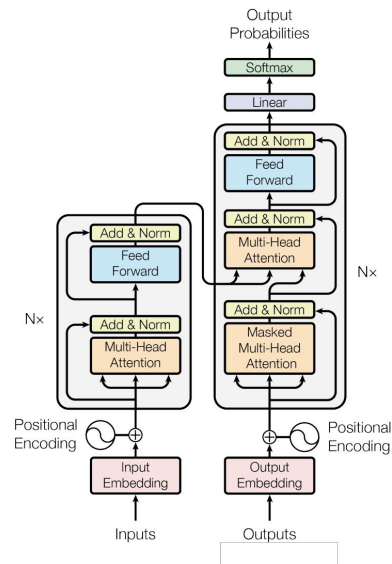
# Graph Neural Network

Examples:

-   Graph Convolutional NN



ref

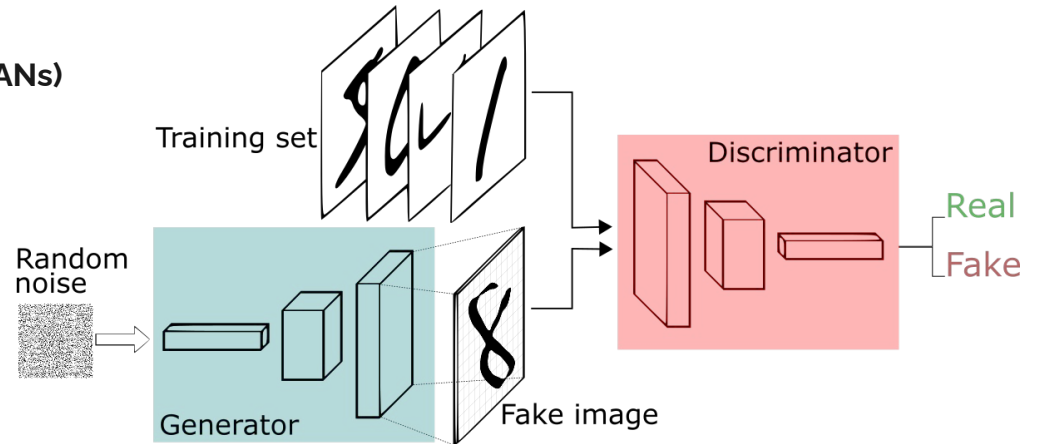# Unsupervised learning (nonlinear)

- Auto Encoders
- Restricted Boltzmann Machines (RBMs)

# Generative AI

Examples:

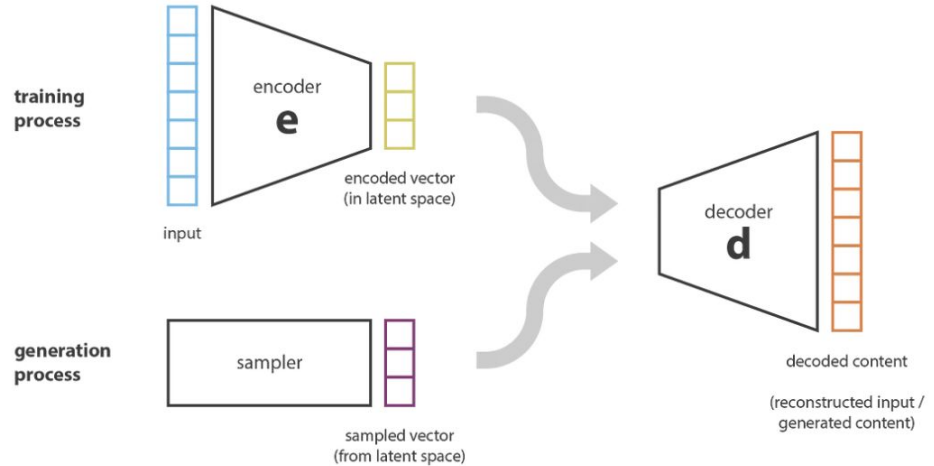- **Generative Adversarial Networks (GANs)**
- Variational Autoencoders (VAEs)



ref

# Generative AI

Examples:

- Generative Adversarial Networks (G
- **Variational Autoencoders (VAEs)**



training process

input → encoder **e** → encoded vector (in latent space)

generation process

sampler → sampled vector (from latent space)

decoder **d** → decoded content (reconstructed input / generated content)

ref

# QA