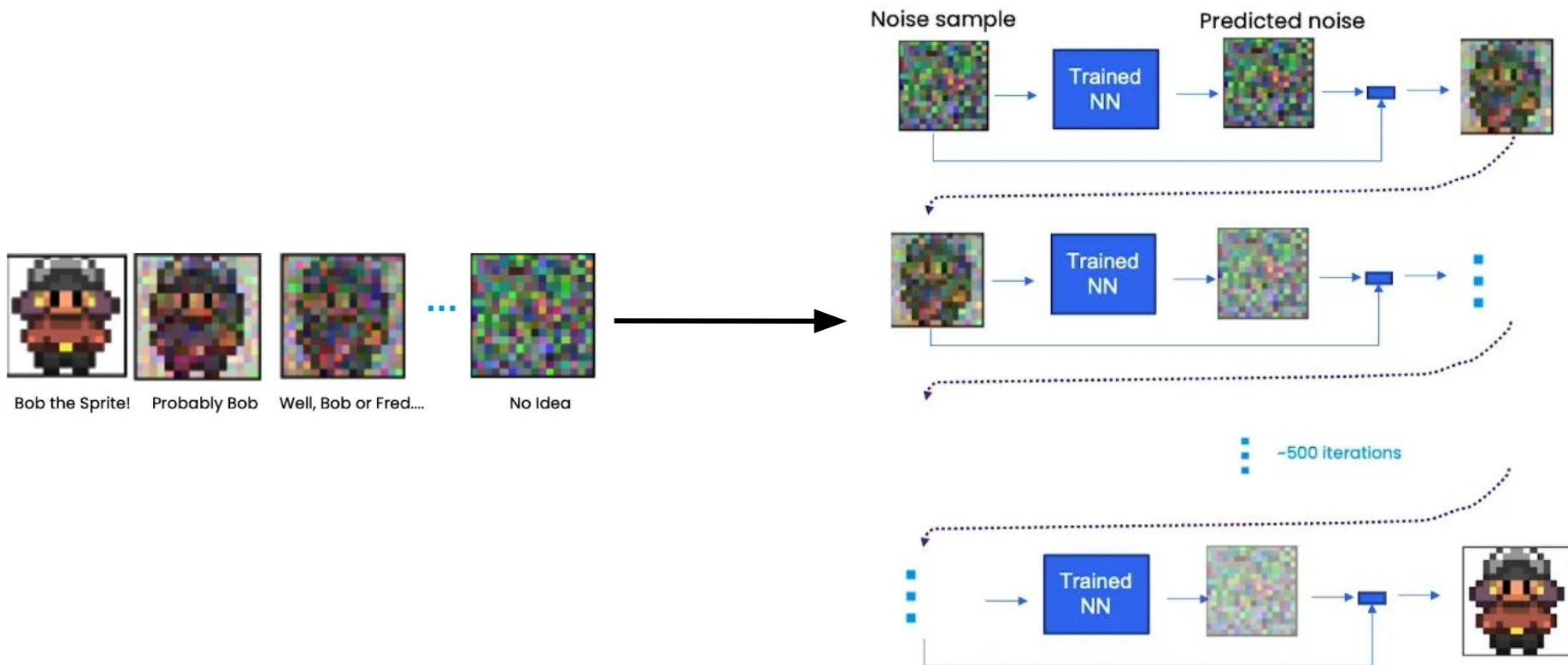# Image Generation with a Diffusion Model

Zeynep Gültuğ Aydemir
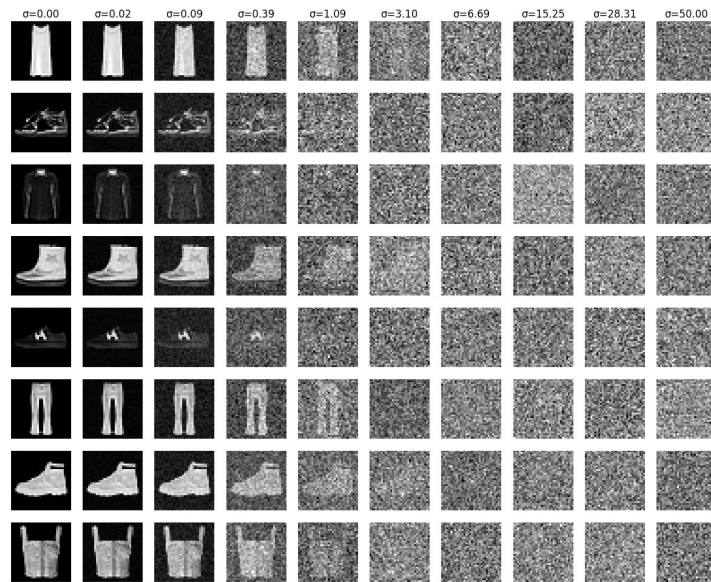Deep Learning 14x050 - Mini Project

How Diffusion Models Work

Noising Procedure

- Gradually add Gaussian noise as $y + \sigma \cdot \epsilon$, where:
  - σ: Noise level (increases from left to right)
  - ε: Random noise sampled from a Gaussian distribution $N(0, I)$
- At high noise levels, the images should be unrecognizable and resemble random noise.
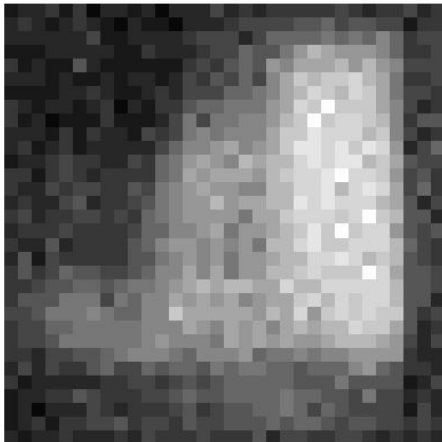

Data ⟶ Noise

Training Pipeline

- Pass the noisy image x through the model:
  - Model learns to predict the residual noise.
- Compute the target, output and MSE loss:
  - Target: Residual noise that was added to y
  - Output: Predicted noise component that needs to be removed from x
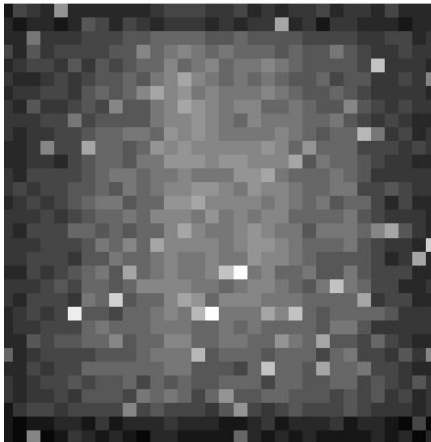- Optimize the model using backpropagation.

Training and Validation Loss Over 100 Epochs

Training and Validation Loss Over 100 Epochs
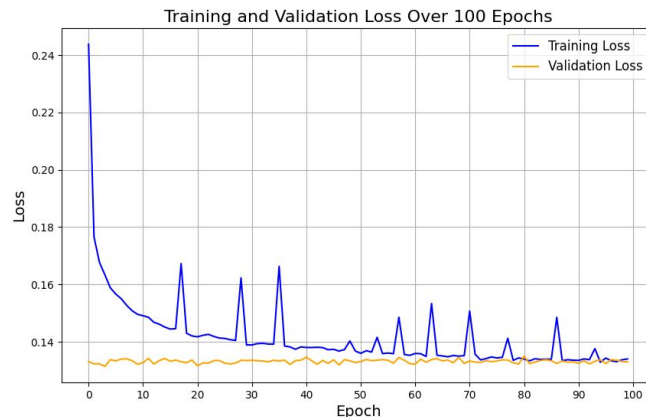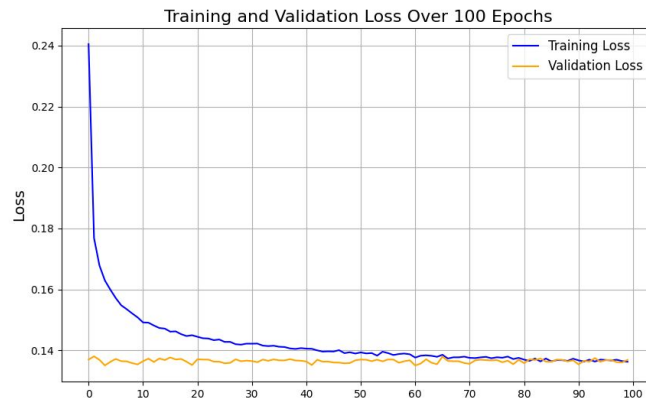
Sampling Procedure
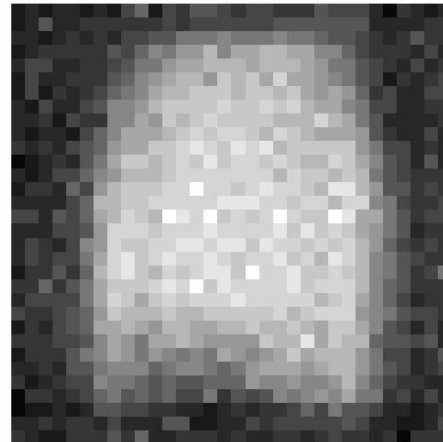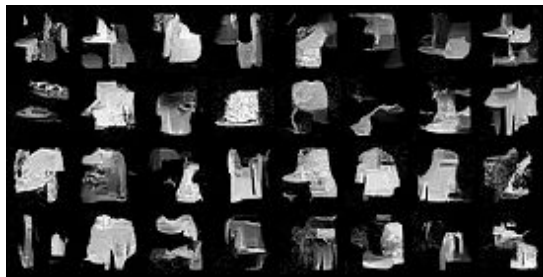

Step 1   Step 1   Step 1

Model Improvements

Base Model - no noise conditioning

Conditional BatchNorm2d - noise conditioning + improved initialization

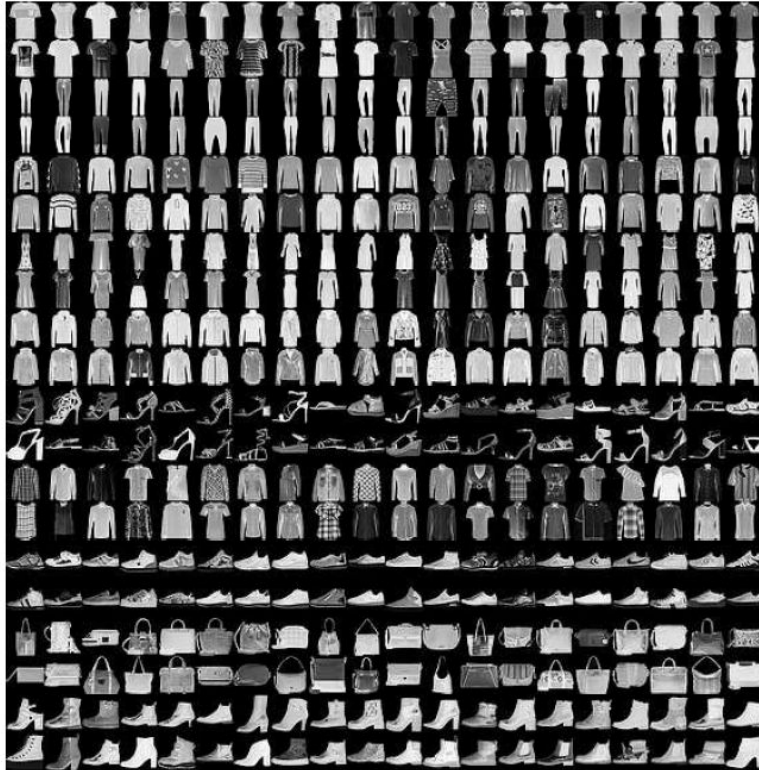# Class-Conditioning

T-Shirt/Top
Trouser
Pullover
Dress
Coat
Sandals
Shirt
Sneaker
Bag
Ankle boots

Model Evaluation

Model	Parameters	Average FID
Base Model	nb_channels=64, num_blocks=8, cond_channels=32 (50ep)	127.18
Base Model	nb_channels=128, num_blocks=8, cond_channels=64 (50ep)	113.27
Cond. BatchNorm2d 1	nb_channels=128, num_blocks=6, cond_channels=32 (50ep)	113.73
Cond. BatchNorm2d 2	nb_channels=128, num_blocks=8, cond_channels=64 (50ep)	107.05
Class Conditional 1	nb_channels=128, num_blocks=6, cond_channels=32 (50ep)	89.56
Class Conditional 2	nb_channels=128, num_blocks=8, cond_channels=64 (50ep)	87.83
Class Conditional 3	nb_channels=128, num_blocks=6, cond_channels=32 (100ep)	89.88
Class Conditional 4	nb_channels=128, num_blocks=8, cond_channels=64 (100ep)	87.86

Class Conditional Model Evaluation

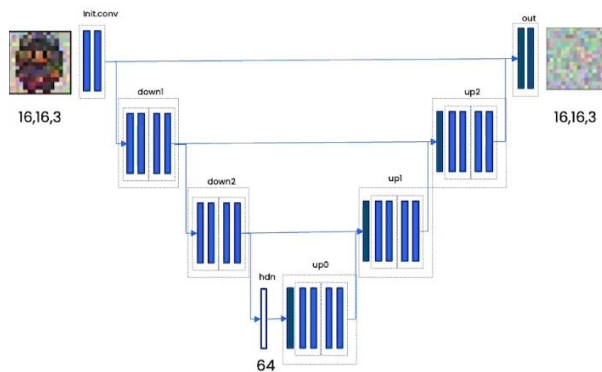| Model/Class	T-shirt	Trouser	Pullover	Dress	Coat	Sandal	Shirt	Sneaker	Bag	Boot
Model 1	133.04	53.28	80.15	75.14	82.55	113.3	86.5	83.14	120.5	68.08
Model 2	137.15	46.14	83.54	79.88	91.1	113.13	80.36	76.29	106.8	63.89
Model 3	135.52	55.20	90.01	73.28	78.87	114.8	91.26	84.59	104.88	70.23
Model 4	143.98	46.52	86.13	78.49	87.51	116.65	85.5	69.1	102.39	62.43

Final Results
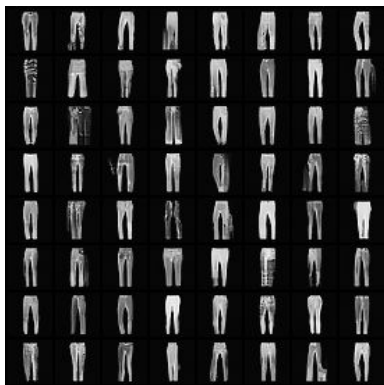
Possible Improvements

- Investigate low performing classes
- Stabilize loss curves with learning rate scheduler
- Better architecture with UNet model
- Face generation using CelebA

UNet