Diffusion Model with U-Net vs. Visual-Transformer

Samuil Ganev
siganev@fmi.uni-sofia.bg

Nikolay Ivanov
nikolayi1@fmi.uni-sofia.bg

Abstract

Language models and all other kinds of generative models are increasingly popular. Our task was to implement a diffusion model for generating images from text in two ways - the standard one using the U-Net architecture and a less popular variant using Visual Transformer - and as a final step to compare their performances. We've used the idea and implementation approach in the original 2020 paper as inspiration - https://arxiv.org/pdf/2006.11239.pdf

1. Dataset

The dataset supposedly influences the abilities of either approach, so we could not draw a general conclusion about which model is better by training on only one dataset. However, the one we have chosen consists of about 3000 images from old books and are 220x220 in size. Link - https://huggingface.co/datasets/gigant/oldbookillustrations?row=0.

Training on such a large set of images of similar size is not achievable with a standard computer, so we have chosen only a small part of these images to show the result of the training and it can be done relatively quickly.

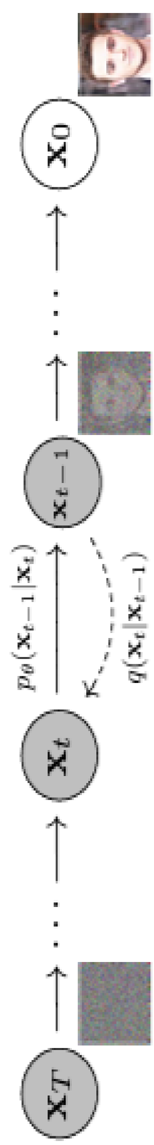
2. Background

The method of operation is a continuous Markov-type circuit whose states are different amounts of noise. Our goal is to use a conditional model (the condition is text) to "go" to a pure image state, starting from Gaussian noise.

We define a finite number of steps T and add noise for each step, where the transition probabilities from $t-1$ to t (more noise) are

$$q(x_t|x_{t-1}) := N(x_{t-1}, \sqrt{1-\beta_t}x_{t-1}, b_t I),$$

where β_t are numbers chosen by us - defining the transition speed from one state to another. The goal is to find the parameters of the distributions that allow us to "go backwards".



Since the steps represent a weak addition of a Gaussian distribution to the forward representation, it is assumed that we can model the transition probabilities in the opposite direction again by a normal distribution, and the problem reduces to finding the parameters but the normal distribution depending on the current state.

$$p_\theta(x_{t-1}|x_t) = N(x_{t-1}; \mu_\theta(x_t, t), \sigma_\theta(x_t, t))$$

Sampling:

$$q(x_{t-1}|x_t, x_0) = N(x_{t-1}; \mu_t^*(x_t, x_0), \beta_t^* I)$$

$$\mu_t^*(x_t, x_0) \coloneqq \frac{\sqrt{\alpha_{t-1}^*}\beta_t}{1-\alpha_t^*}x_0 + \frac{\sqrt{\alpha_t}(1-\alpha_{t-1}^*)}{1-\alpha_t^*}x_t$$

$$\beta_t^* \coloneqq \frac{1-\alpha_{t-1}^*}{1-\alpha_t^*}\beta_t$$

$$x_0^* = \frac{x_t - \sqrt{1-\alpha_t^*}\epsilon_\theta(x_t)}{\sqrt{\alpha_t^*}}$$

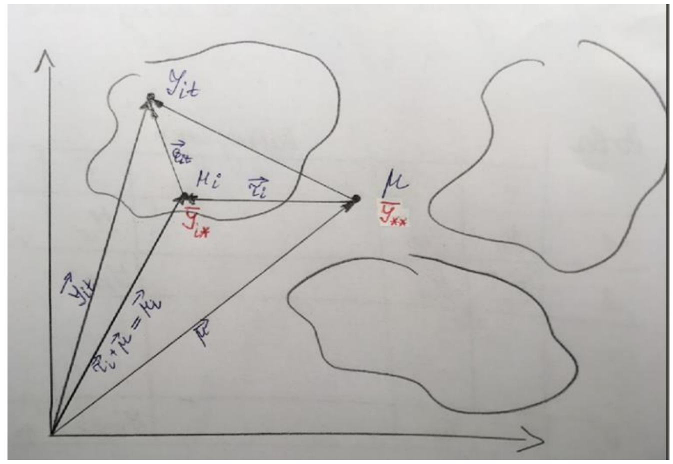
3. Results and comparisons

The following table shows the average training errors of the two models for epochs 10, 50, 100, 200, 300, 400, 500, 600, 700, 800, 900, 1000.

$$d_F(\mathcal{N}(\mu,\Sigma),\mathcal{N}(\mu',\Sigma'))^2 = \|\mu-\mu'\|_2^2 + \mathrm{tr}\left(\Sigma+\Sigma'-2(\Sigma\Sigma')^{\frac{1}{2}}\right)$$

EPOCH	U-NET	TRANSFORMER
10	169	146
50	141	148
100	134	150
200	137	151
300	129	145
400	122	141
500	117	137
600	108	132
700	102	134
800	87	126
900	83	124
1000	80	120

Analysis of Variance Test



$$Y_{1*}^{-} = \frac{1}{12}\sum_{i=1}^{12} UNet_i \qquad Y_{2*}^{-} = \frac{1}{12}\sum_{i=1}^{12} VT_i$$

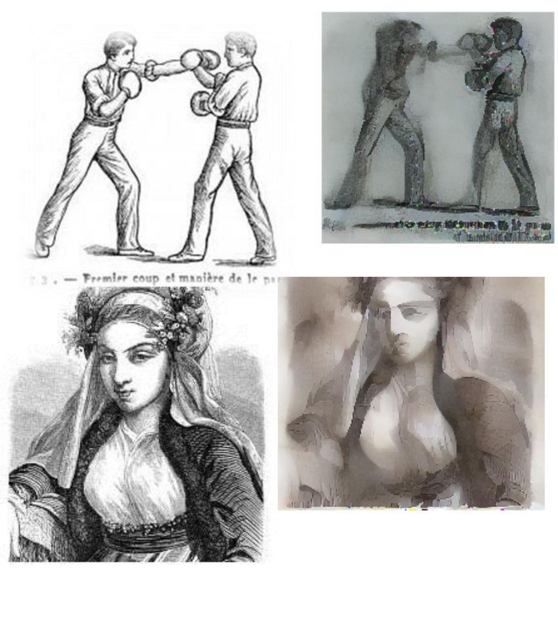
$$H_0: \tau_1 = \tau_2 = 0$$
$$H_1: \exists i: \tau_i \neq 0$$

$$SST \coloneqq 12\sum_{i=1}^{2}(Y_{i*}^{-} - Y_{**}^{-})^2 \sim X_{(1)}^2$$

$$SSE \coloneqq \sum_{i=1}^{2}\sum_{j=1}^{12}\left(Y_{ij}^{-} - Y_{i*}^{-}\right)^2 \sim X_{(22)}^2$$

$$F \coloneqq 22\frac{SST}{SSE} \approx 44.56$$

$$=> p_{value} = 5.53e-05$$



Comparison when generating with different number of inference steps:

20	50	100	200

And a combination of two images that we managed to get by combining the descriptions of the two images - we observe that it rather overlaps the two images into one, but the result is interesting:

+

=