The goal of this mini project is to test the fine-tuning methods in Text2Image models. Fine-tuning is an important technique in NLP and related Machine Learning areas, allowing us to adapt the power of large models to perform well on specific tasks, rather than creating new models from scratch for these purposes. This approach both saves time and makes the results more accurate and according to our expectations. I decided to try to use the fine-tuning methods for image generation based on the textual prompt. As a result, we will be able to generate images aligning with a particular style. For doing this, we will use the following source: https://huggingface.co/blog/lora (LoRA for Stable Diffusion).

Basically, everything I needed to fine-tune the model was provided there. I decided to choose the old book illustrations dataset (https://huggingface.co/datasets/gigant/oldbookillustrations), because it seemed to be interesting. According to the description, 'The Old Book Illustrations contains 4172 illustrations scanned from old books'. Hence, it can be pretty much understood, what kind of images can be found there. The types of images presented in the files can be checked using the following link: https://huggingface.co/datasets/AdamLucek/oldbookillustrations-small?row=0.

I used one more source to help me better understand the provided pipeline. https://docs.google.com/document/d/1KRHOdVA75olgOQNJOSll29OYkfh-zTTXwaEssPlZ8rA/edit?tab=t.0#heading=h.yi9km4tgu2pk document (also, the YouTube video is available) describes in detail the pipeline, as well as the arguments that are passed while implementing the fine-tuning.

For fine-tuning, I took the CompVis/stable-diffusion-v1-4 model (the version of the model is not very important for this task, as we are mainly interested to observe the style changed through fine-tuning process).

After installing all the necessary things, I started the fine-tuning process. I used the following command to run it:

```
! accelerate launch train_text_to_image_lora.py \
  --pretrained_model_name_or_path=$MODEL_NAME \
  --dataset_name=$DATASET_NAME \
  --dataloader_num_workers=8 \
  --resolution=512 --center_crop --random_flip \
  --train_batch_size=1 \
  --gradient_accumulation_steps=4 \
  --max_train_steps=2501 \
  --learning_rate=3e-04 \
  --max_grad_norm=1 \
  --lr_scheduler="cosine" --lr_warmup_steps=0 \
  --output_dir="fine_tuned_model" \
  --mixed_precision="fp16" \
  --checkpointing_steps=500 \
```

```
--seed=1337
```

I did not change almost anything from the tutorial, except for several important things. First, I did not want to push my model to the HuggingFace. Hence, I removed all the parameters related to it. Second, instead of 15 000 steps as was in the initial guide, I did just 2500. The reason is that I did not want my model to train for too long. Meanwhile, to accelerate the learning process, I changed the learning rate from 1e-4 to 3e-4. We will later see that it is sufficient to observe the fine-tuning results.

After the fine-tuning process was finished, I checked if both the original model and the fine-tuned model work on inference. To do so, I passed an arbitrary prompt: 'A futuristic cityscape at sunset with flying cars and neon lights'. It worked. Obviously, with the case of old books, there is nothing to analyze, so at that moment I moved to the most interesting stage.

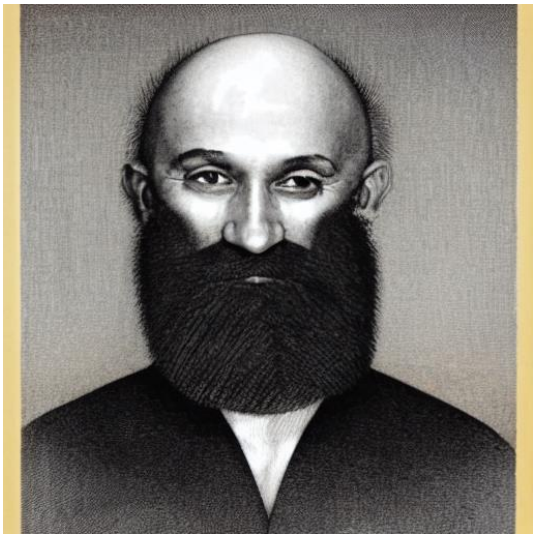
I got 15 prompts that can in theory have to do something with the old books. These prompts are:

```
prompts = ['A bald man with a long beard smiles faintly',
           'Vase with foliated ornaments',
           'A big black cat looking at the window',
           'A monkey wearing a feathered hat walks around the countryside',
           'A boy is reading a book in his bedroom',
           'A family is having a dinner',
          'A country cottage',
          'Wildflowers in a meadow',
          'A steam train on a bridge',
          'Children playing in a park',
          'A medieval castle',
          'A botanical illustration of a rose',
          'A map of Europe in the 18th century',
          'A caricature of a politician',
          'A scene from a Shakespeare play']
```

Then, I generated images using the original model, and using the fine-tuned one.

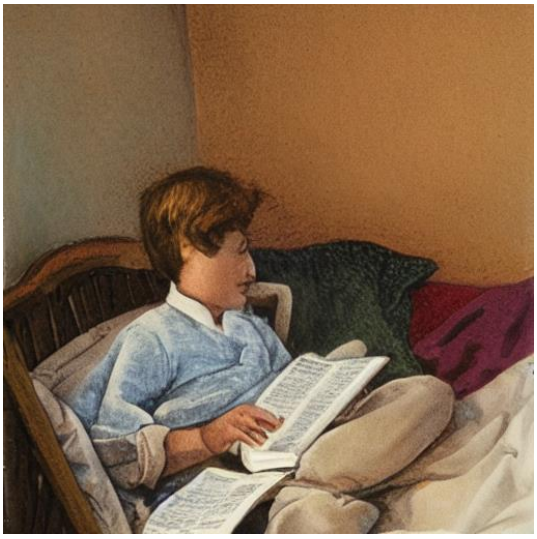
Here are some of the results (original image is on the left, the fine-tuned is on the right):

'A bald man with a long beard smiles faintly'



'A big black cat looking at the window'

'A boy is reading a book in his bedroom'

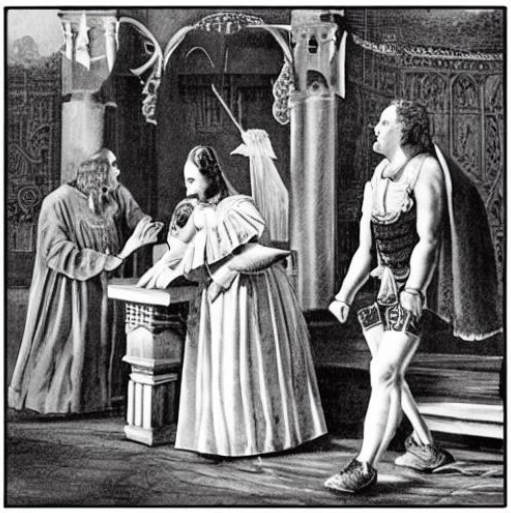


'A family is having a dinner'

'A medieval castle'

'A botanical illustration of a rose'

'A scene from a Shakespeare play'



These results are sufficient to show that the fine-tuning process was successful. In my opinion, most of the images generated by the fine-tuned model match the style of the old book dataset. Moreover, they are clearly different from the original images.

Note that in this report I analyze only some of the images. All the images are located in the Images Original Model and Images Fine-Tuned Model for the images generated by the original model, and the images generated by the fine-tuned model, respectively.

Note that all the images were generated for educational purposes only, and using the open-source model!