Project Planning Report - VAE-based Medical Image Generator

Leon De Andrade, Bjarne Gerdes, Simon Schmalfuß
12. Juni 2023

1 Introduction

This planning report outlines a bonus coding project conducted within the „Deep Generative Models“ module. The project focuses on applying learned knowledge to code Variational Autoencoders (VAEs) for generating medical diagnostic images. This report provides an overview of our project plan in terms of objects and planning milestones. This bonus project offers an opportunity to apply acquired knowledge and expand our understanding of deep generative models.

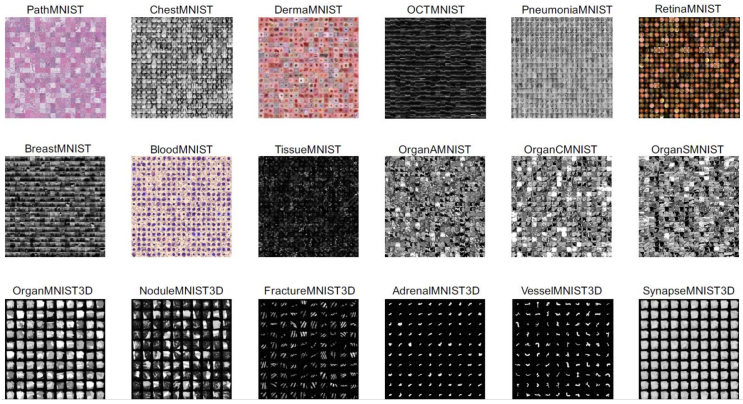

Abbildung 1: The picture showcases a collection of medical images from the MedMNIST dataset, that we will use in our project [1]

2 Objectives

1. Basic Task: Implement and train a VAE to generate medical images from the MedMNIST dataset using PyTorch. By successfully training the VAE, we aim to generate realistic and diverse synthetic medical images across different modalities, such as X-ray, Pathology, and Dermatology.

2. Extension (a): Implement a variant of the VAE that incorporates disentanglement techniques. Explore and experiment with methods to disentangle the latent representation of the generated images. Evaluate the disentangled VAE on at least three different modalities to assess its ability to capture the unique characteristics of each modality.

3. Extension (b): Alternatively, implement a VAE variant for conditional training. Condition the VAE on additional information, such as specific medical conditions or demographics, to generate condition-specific medical images. Evaluate the conditioned VAE on multiple modalities, including X-ray, Pathology, and Dermatology, to assess its effectiveness in producing tailored images.

Our objective is to gain experience in implementing VAEs using PyTorch and to demonstrate our ability to generate high-quality medical images across various modalities. Additionally, exploring the extensions of disentanglement and conditional training will allow us to learn and apply advanced techniques in deep generative models for medical image generation. By achieving these goals, we aim to develop our skills and knowledge in this field, and make a contribution to the growing area of medical image generation.

3 Planning

To build a VAE-based medical image generator, we will complete the following tasks:

1. **Data exploration and preprocessing**: Explore the MedMNIST dataset to understand its characteristics and challenges. Preprocess the data to prepare it for training the VAE model.
2. **VAE theory and architecture**: Study the theory behind Variational Autoencoder (VAE) and its architecture, including the encoder and decoder networks, the loss function, and optimization techniques.
3. **Basic VAE implementation**: Implement a basic VAE model using pytorch, including designing the encoder and decoder networks, defining the loss function, and specifying the optimization algorithm.
4. **Evaluation metrics selection**: Select appropriate evaluation metrics for our VAE-based medical image generator, such as reconstruction error, image quality, and diversity. Develop a method for visualizing the generated images.
5. **Training and hyperparameter tuning**: Train the VAE model on the MedMNIST dataset and perform hyperparameter tuning to optimize its performance. Experiment with different training strategies, regularization techniques, and optimization algorithms.
6. **Extension task exploration**: Explore possible extensions to our VAE-based medical image generator, such as disentanglement or conditional VAE. Compare the advantages and limitations of each extension and select the most relevant and feasible one for our project.
7. **Extension task implementation**: Implement the chosen extension to our VAE-based medical image generator, modifying the existing VAE architecture and defining additional loss functions and optimization strategies. Evaluate the performance of the extended model using the selected evaluation metrics.

By completing these tasks, we will develop a VAE-based medical image generator that can generate high-quality and diverse medical images.

We plan to meet once a week in person to discuss our progress and any challenges we may be facing. Additionally, we will use Google Colab as our development environment, allowing us to work in parallel and access CPU/GPU resources as needed. By using Google Colab, we can easily share our code and notebooks, and collaborate in real-time. This will enable us to work more efficiently towards our project goal.

Literatur

[1] *MedMNIST*. `https://medmnist.com/`. (Accessed on 06/05/2023).