

# CSE 5370 Bioinformatics Homework 4

**Name:** Venkat Ankit Gundala

**Student ID:** 1002069069

## 1 Introduction to VAE

VAE stands for Variational Autoencoder, which is a type of neural network architecture used for unsupervised learning of complex data representations. VAE is composed of two main components: an encoder network that maps the input data to a latent space, and a decoder network that maps the latent space back to the original data space. The encoder network takes the input data and maps it to a mean and variance in the latent space. Then, a random sample is drawn from the latent space based on the mean and variance, and passed to the decoder network to generate a reconstruction of the input data.

## 2 PyTorch Implementation

The assignment is implemented on Google Colab, offering free access to GPUs and pre-installed Python packages like PyTorch. A framework with code placeholders is provided, and it is recommended to understand each module's function for filling the placeholders.

### 2.1 Setting Up the Framework

I have Downloaded the vae.zip file which was attached to the assignment. I have copied the vae file to MyDrive folder on the Google Drive. The image below is the reference for the vae directory.



## 2.2 Downloading your Dataset

After downloading the vae.zip file from the assignment and copied it to the google drive we can find all the files present in the vae directory and we have to open main.ipynb using colab to access the file. After opening the notebook we have to ensure that the notebook has GPU access if it doesn't have the access to GPU do this go to Runtime → Change runtime type → Hardware Accelerator and select "GPU". In the fourth block make sure to update the random student id with the original student id. Run the first 7 blocks of code so the data gets downloaded. To check whether the images are downloaded check the data folder on the google drive if the folder has 3 ".svs" files in it then the images are correctly downloaded into the folder.

## 2.3 dataset.py Placeholders

We have been asked to implement 6 lines of code so the function returns index'th item in our dataset and its size. The first line extracts the file\_path and the (x,y) coordinates from the "coords" attribute of the class instance at the specified index. The second line uses a private method "-load-file" to load the image at the specified file\_path. The third line uses another private method "\_image\_to\_tensor" to convert the image patch specified by the (x,y) coordi-

nates to pixel values and stores the result in a variable called "out". The fourth line checks if there are any transformations specified in the "transformations" attribute of the class instance. If transformations are present, they are applied to "out" using the transformations function. Finally, the method returns "out" and its size as a tuple.

We have been asked to implement 1 line of code so it returns the length of "self.coords".s, I have used len() function to return the length.

## 2.4 datamodule.py Placeholders

We have been asked to implement 1 line of code so it returns Dataloader for self.train\_dataset, batch size of self.batch\_size, number of workers of self.num\_dataloader\_workers and drop\_last to be True.

We have been asked to implement 1 line of code so it returns Dataloader for self.val\_dataset, batch size of self.batch\_size, number of workers of self.num\_dataloader\_workers and drop\_last to be True.

We have been asked to implement 1 line of code so it returns Dataloader for self.test\_dataset, batch size of self.batch\_size, number of workers of self.num\_dataloader\_workers and drop\_last to be True.

## 2.5 task.py Placeholders

We have been asked to implement 3 line of code so it returns function that is taking in a batch of data and performing a step of training on it. The output of this step is the calculated loss and some logs. Next, the logs are being formatted with a prefix "train\_" added to each key, indicating that these logs are specific to the training step. The logs are then passed to the log\_dict method which logs the values in a structured format. Finally, the function returns the calculated loss. It's worth noting that this function is likely part of a larger training loop, and it's possible that the logs and loss are being used for further analysis or reporting on the model's performance.

We have been asked to implement 5 line of code the function that is taking in a batch of data and performing a validation step on it. The output of this step is the calculated loss and some logs. Logs are being formatted with a prefix "val\_" added to each key, indicating that these logs are specific to the validation step. The logs are then passed to the log\_dict method which logs the values in a structured format. If batch\_idx is equal to 0, then the method is storing the batch in self.val\_outs. Function returns the calculated loss. Used to report model's performance during Validation.

unction that is taking in a batch of data and performing a testing step on it. The output of this step is the calculated loss and some logs. The logs are being formatted with a prefix "test\_" added to each key, indicating that these logs are specific to the testing step. The logs are then passed to the log\_dict method which logs the values in a structured format. If batch\_idx is equal to 0, then the method is storing the batch in self.test\_outs. Function returns the calculated loss. Used to report model's performance during testing.

## 2.7 main.ipynb Notebook

After filling all the place holders from datamodule.py, dataset.py, task.py save them to Google Drive and open main.ipynb notebook and run the remaining cells after cell 7. Then we get some outputs on the Tensor Board from which we can analyze the how the images are processed and what output is displayed at the end of each epoch. After running the cell which has hyperparameters in it displays a lot of details about trainable parameters, encoder, decoder, In Sizes, Out Sizes etc.

## 4 Difficulty Adjustment

It took 10 plus hours for me to complete the assignment.

I was confused how to edit the datamoudule.py, dataset.py, task.py files.