Fourth Assignment

Shahid Beheshti University - Bachelor’s Program

Artificial Neural Networks - Winter 2024

Theoretical Exercises

Exercise 1

Describe the differences between undercomplete, overcomplete, and exactly complete AutoEncoders. Provide examples of scenarios where each type would be beneficial.

Exercise 2

Contrast the training objectives of AutoEncoders with those of traditional supervised learning models. How does the absence of labeled data affect training strategies?

Exercise 3

Compare and contrast the reconstruction loss functions commonly used in AutoEncoder training, such as Mean Squared Error (MSE), Binary Cross-Entropy, and Kullback-Leibler Divergence.

Exercise 4

Discuss strategies for evaluating the quality of the learned representations in an AutoEncoder, particularly in the absence of labeled data.

Exercise 5

Explain the concept of the reparameterization trick in Variational AutoEncoders. How does it enable the backpropagation of gradients through the sampling process?

Exercise 6

Discuss strategies for regularizing the latent space of a Variational AutoEncoder to encourage the emergence of disentangled and interpretable representations.

Exercise 7

Describe the challenges of training Variational AutoEncoders on datasets with complex and high-dimensional input distributions. What techniques can be employed to address these challenges?

Exercise 8

Explain how Variational AutoEncoders can be extended to incorporate additional constraints or modifications, such as conditional generation or semi-supervised learning objectives. What benefits do these extensions offer in practical applications?

Exercise 9

What are the different types of convolutional layers? Explain each one of them and provide scenarios where they are used.

Exercise 10

Explain what dilated convolution is. Why is it being used? Describe real-life situations dilated convolution applies and why (At least three different scenarios).

Practical Exercises

Autoencoder Reconstruction of Mixed MNIST and CIFAR-10 Images

Introduction

In this practical assignment, you will delve into autoencoders, a fundamental technique in deep learning, to reconstruct images from two distinct datasets: MNIST and CIFAR-10. The objective is to create an autoencoder model capable of taking the mean of an MNIST and a CIFAR-10 image, feeding it into the model, and generating reconstructions of both MNIST and CIFAR-10 images.

Task Overview

1. Select a random image from the MNIST dataset and a random image from the CIFAR-10 dataset.
2. Compute the mean of these two images.
3. Prepare the MNIST image for input into the autoencoder:
    a. Resize the MNIST image to match the dimensions of CIFAR-10 images.
    b. Extend the channels of the MNIST image to match the channels of CIFAR-10 images.
4. Construct an autoencoder model in PyTorch.
5. Input the mean image into the autoencoder.
6. Obtain reconstructions of both MNIST and CIFAR-10 images from the autoencoder.
7. Evaluate the quality of reconstructions.

Detailed Instructions

1. **Dataset Preparation**

    a. Utilize the MNIST and CIFAR-10 datasets.
    b. MNIST images are grayscale (1 channel) and have dimensions of 28x28 pixels.
    c. CIFAR-10 images are RGB (3 channels) and have dimensions of 32x32 pixels.

2. **Mean Image Computation**

    a. Randomly select one image from the MNIST dataset and one from the CIFAR-10 dataset.
    b. Compute the mean of these two images element-wise.

3. **MNIST Image Preparation**

    a. Resize the MNIST image to match the dimensions of CIFAR-10 images (32x32 pixels).
    b. Extend the channels of the MNIST image to match the channels of CIFAR-10 images (from 1 to 3 channels).

4. **Autoencoder Model Construction**

    a. Design an autoencoder architecture in PyTorch.
    b. The encoder should compress the input image into a latent representation.
    c. The decoder should reconstruct the input image from the latent representation.

5. **Model Training**

    a. Train the autoencoder using the mean image computed earlier.
    b. Use appropriate loss functions such as Mean Squared Error (MSE).
    c. Optimize the model parameters using an optimizer like Adam.

6. **Reconstruction**
7.
    a. Input the mean image (prepared MNIST image) into the trained autoencoder.
    b. Obtain reconstructions of both MNIST and CIFAR-10 images from the autoencoder.

8. **Evaluation**

    a. Assess the quality of reconstructions visually.
    b. Compare the reconstructed MNIST and CIFAR-10 images with their originals.
    c. Consider metrics such as structural similarity index (SSIM) or peak signal-to-noise ratio (PSNR) for quantitative evaluation.