Neural Networks and Deep Learning (2024)

Assignment (3):

Blind Source Separation Using Variational Autoencoders

Shiraz University

Due date: 8th of Tir 23:59 [28/June (2024)]

1 Introduction

Blind source separation (BSS) is a signal processing technique that aims to recover individual source signals from a mixture of signals, without any prior information about the source signals or the mixing process. It finds applications in various fields, including audio signal processing, biomedical signal analysis, and wireless communications. By combining signal processing principles with artificial intelligence (AI), BSS has become a powerful tool for extracting meaningful information from complex mixtures encountered in various domain.

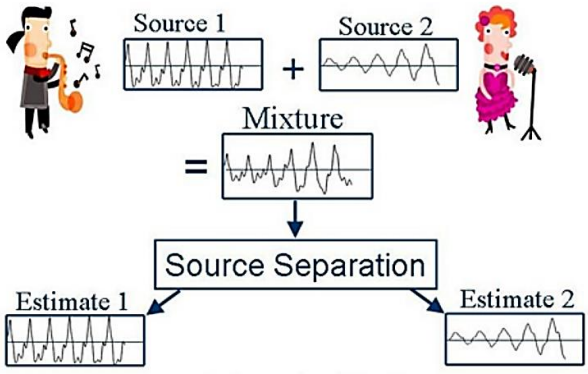

Figure 1: Illustration of vocal/music separation process

One of the applications of blind source separation is in separating vocal and background music components from audio recordings, a process known as vocal/music separation or singing voice separation. This process is illustrated in Figure 1.

2 Problem Statement

For this assignment, first, you are going to use Variational Autoencoders (VAEs) to separate mixed MNIST handwritten digit and Fashion MNIST images, and for the second part you are going to use VAEs to separate vocal and background music components from audio recordings with provided datasets.

2.1 Datasets

2.1.1 MNIST & Fashion MNIST datasets

MNIST dataset is collection of handwritten digits, available from this page, has a training set of 60,000 examples, and a test set of 10,000 examples.

Fashion-MNIST is a dataset of Zalando's article images—consisting of a training set of 60,000 examples and a test set of 10,000 examples. Each example is a 28x28 grayscale image, associated with a label from 10 classes.

You can download these datasets using built-in functions in libraries like TensorFlow, Pytorch, etc.

2.1.2 IRMAS dataset: [DOWNLOAD LINK]

IRMAS dataset includes musical audio excerpts with annotations of the predominant instrument(s) present. It is intended to be used for training and testing methods for the automatic recognition of predominant instruments in musical audio. The instruments considered are: cello, clarinet, flute, acoustic guitar, electric guitar, organ, piano, saxophone, trumpet, violin, and human singing voice.

For the second part of this assignment, you will work with a limited part of this dataset, human singing voice (voi). The sample rate for each audio is 44100 Hz.

2.2 Part I: Mixed MNIST images BSS

For this part, you are going to implement a VAE model to separate mixed image of random MNIST handwritten digit and Fashion MNIST images. Your VAE must accept an image, which is a mixture of one MNIST handwritten digit image and one Fashion MNIST image, as its input and return the 2 separated MNIST images along with the reconstructed mixed image itself.

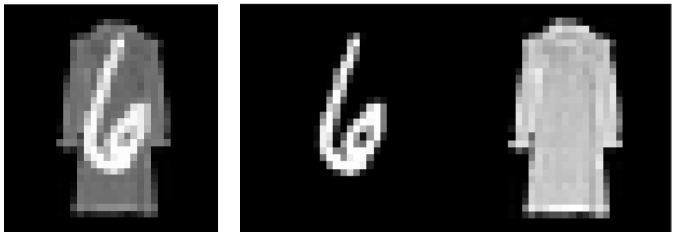

Figure 2: an example of input and output of VAE model for blind source separation of mixed MNIST image

2.3 Part 2: Music BSS

For this part, you'll have to modify your VAE to work on music blind source separation. Use the following tips to ease your work:

- You can use the spectrogram image of each audio as your model input.
  - Be very careful with parameters while computing the spectrogram.
- If your model doesn't perform well using dense layers try out convolutional layers.

3 Tasks

1. The performance of models like VAEs is highly dependent on the choice of loss function. Different losses (Cross-entropy, Kullback-Leibler divergence, etc.) might perform differently based on your data and application. Study these loss functions to choose the most appropriate ones for your implementation.

2. Display your results along with your loss value after each N epochs while your model is training. You can choose N freely.
3. Discuss how well VAEs perform in domain of blind source separation for each dataset. What models could outperform VAEs for each task.

Notes:

- Allowed programming languages: Python, MATLAB
- Any sign of cheating would result in a zero grade for this assignment.
- You should upload your submissions at:
  https://quera.org/course/add_to_course/course/16595/
  All of the files should be in a ZIP file named in this format:
  "FirstNameFamilyName-SudentNumber.zip"
  Ex: "AmirZamani-4023040.zip"
- Your reports should be in a PDF file including: key points of your implementation, explanation of your chosen approach, reports of your final results and answers of assignment questions (if given).
