Speech Understanding - Minor 2

Alli Khadga Jyoth - M23CSA003

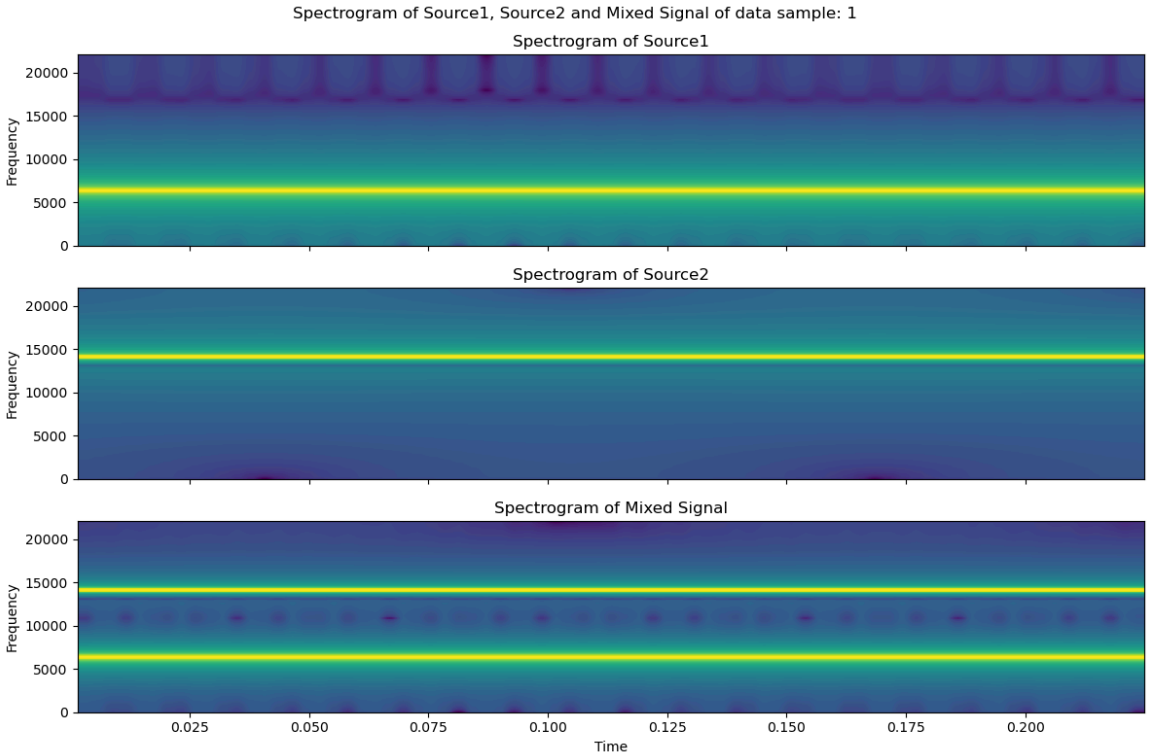
The code is provided in the zip file.

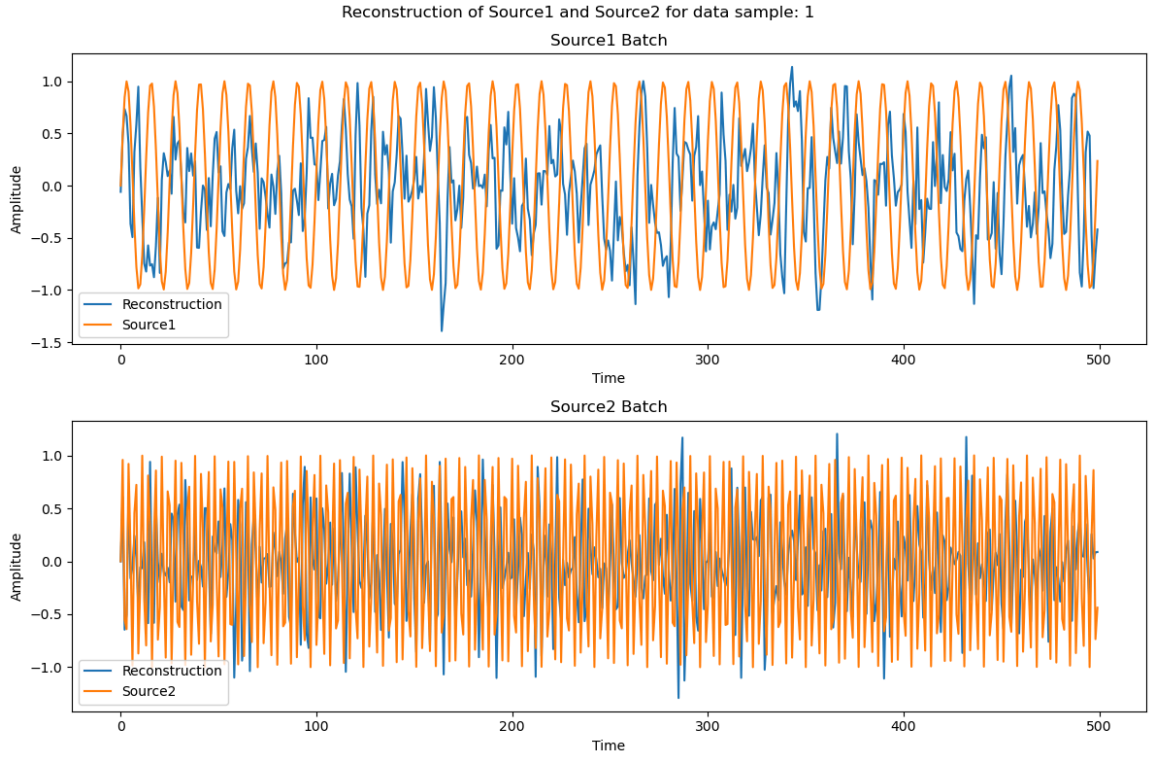
Q1:

This is a task of source signal separation using LSTM. In this task we create a mixed sinusoidal dataset by mixing two sinusoidal signals with different frequencies. One signal with frequency above the fundamental frequency of 8000 Hz and another with lower than fundamental frequency. The dataset contains 10000 timesteps.

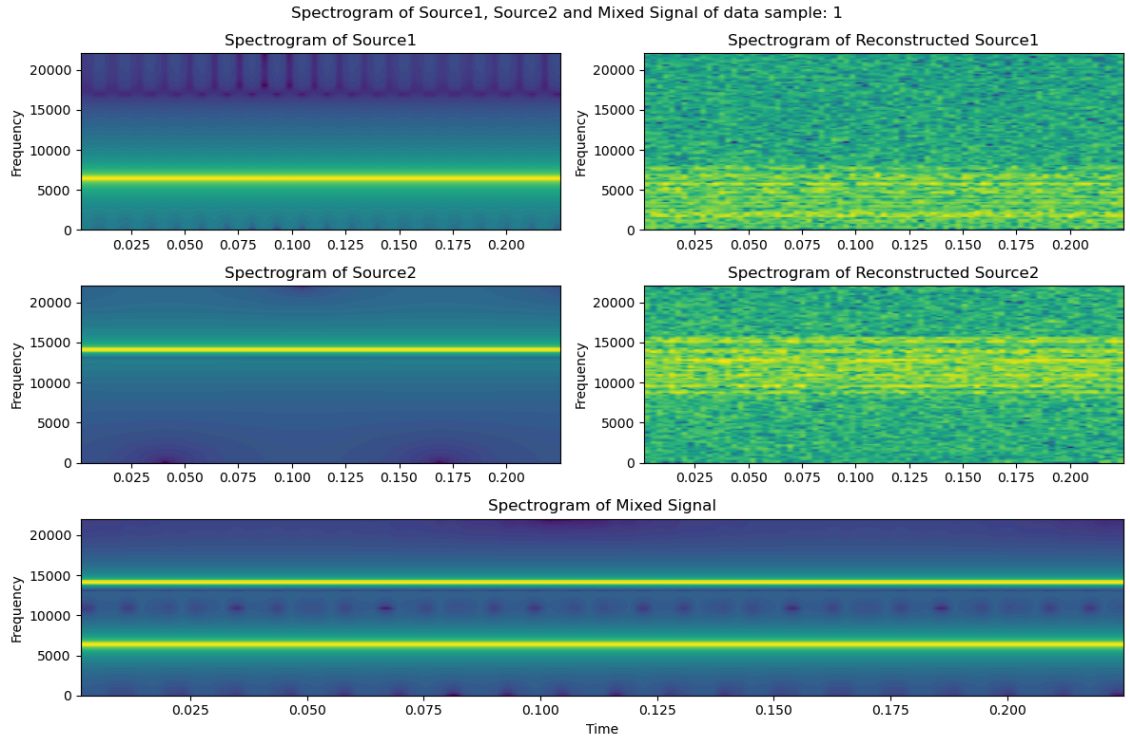
The model contains 1 LSTM layer with hidden layer size of 64 and a projection size of 32. This 32 unit output is passed through a fully connected layer to give us 20000 unit output. Out of which the first 10000 units corresponds to the source 1 and later 10000 units corresponds to the source 2. These outs are then compared with the true sources with respect to the L1 loss function.

The model is evaluated using the SI-SNS metric. The model is trained for 10 epochs using the ADAM optimizer with a learning rate of 0.01. The results are presented below.
Example datapoint





Model signal reconstruction



Model signal reconstruction using spectrogram

Training Metrics:

Training loss : 1.04
SI - SNR : - 6.11

Testing Metrics:

Loss : 1.43
SI - SNR: -49.14

Q2:

1.

The code for CTC is written in the Jupyter notebook file.

2.

Evaluating the custom CTC code by comparing it with pytorch implementation is also present in the notebook file.

Results I go for 1 runt:
PytorchCTC : 6.0781
Custom Implementation: 6.0781

References:

[1] numpy - How do I generate a sine wave using Python? - Stack Overflow
[2] aishoot/LSTM_PIT_Speech_Separation: Two-talker Speech Separation with LSTM/BLSTM by Permutation Invariant Training method. (github.com)
[3] vadimkantorov/ctc: Primer on CTC implementation in pure Python PyTorch code (github.com)