

## **Project Title : Music to Music Translation**

**Project Group : Priyam Khandelwal , Appoose MD**

Our Project aims to translate music produced by one instrument to the music of another instrument. For translation purpose, Generative Adversarial Networks (GAN's) are correct and relevant choice in deep learning. As many other applications like image to image translation has been done using GAN's successfully. GAN's are computationally intensive.

For capturing the sequential information / temporal dependencies Recurrent Neural Networks (RNNs) are very effective. Since Music has temporal dependencies, so the basic elements which we have used in our model are Long Short Term Memory Networks (LSTM : variant of RNN and more powerful than RNN).

Since paired Dataset is not available, so model consists of cyclic GAN which consists of 2 Generators and 2 Discriminators. Each of them is a LSTM with at least 3 hidden layers. One LSTM itself is a folded Neural Network which when unfolded is a sequence of 256 steps ( 256 Neural Networks , in this case ).

So, the training of our model consists of training 4 different Neural Networks, each having complex structure as described above. Along with this, training of GAN's itself is a difficult task as it's very hard to stabilize it.

For creating this network, tensorflow is being used which is a primary software tool of deep learning.

### **Why Tensorflow ?**

- When it comes to Neural machine translation, TensorFlow reduces errors by 55%-85%.
- TensorFlow allows coders to iterate quickly, train models faster and run more experiments
- With TensorFlow, we can design a multilayered model easily.
- With this, we can build a computational graph and then feed it with data to train.
- Tensorflow provides various Optimizers which if coded manually, cannot be guaranteed to provide accuracy to such an extent which it gives.

Tensorflow uses GPU (given priority over CPU) for performing the calculations ( a lot of numpy operations all at single time ) so that network can be trained faster. It needs CUDA ( is a parallel computing platform and application programming interface (API) model created by Nvidia ). The computational graph which it creates, itself requires a good amount of memory.

If it is setup on a normal Laptop having an Nvidia Graphics card, then we can setup the pipeline for project but it starts throwing errors like Out Of Memory, etc for even normal batch sizes. It also restricts the user to do other tasks as many times it gets hanged. So, training such a deep learning model is very difficult on a normal machine.

That is why, a powerful GPU server/machine is needed for training such models.