**Language Transliteration Usage Guidelines**

Notebook demonstrates the transliteration of Indic words in to English; currently this model is being trained on Indian Language and get it transliterated into English.

Its bases on character level Sequence to Sequence Recurrent Neural Network(RNN) Encoder Decoder Deep Learning approach and this would be script independent i.e. output does not depends on linguistic of the script just we have to give large number of pairs of local language words with corresponding English words and train it character by character to create two RNN network (Encoder and Decoder), currently its being trained on Indian Language to English with character level mapping and it works because for Indian languages there is a character by character mapping with that of English , As this model is based on RNN we could further enhanced this model to extend the support to other foreign languages where one symbol/character corresponds to a word like we have in Korean/Japanese/Chinese .

We are publishing the source of this model so that it could be trained on language transliterated pair dataset. Input to the model would be comma separated local language words and English words e.g.

गणेश राजाभाऊ,ganesha rajabhau

अरुण रंगनाथ,aruna ranganatha

Below are usage steps

1. There are two notebooks **TrainTraslit-word.ipynb** and **TestTranslit-word.ipynb**
2. Install Python 3.6 and tensorflow 1.14(I used)
3. Start the jupyter notebook
4. Open the **TrainTraslit-word.ipynb,** give the dataset as in fist cell
5. Execute all the cells and install dependencies as per it asked for.
6. At last it will generate a model in the current directory **s2shin.h5**
7. Once the model trained open **TestTranslit-word.ipynb,** Give the inference file name in first cell
8. Execute all the cells make sure num\_samples should be same as given in training.
9. It will load the model saved in step 6 while training.
10. All inference output will saved in **TranscriptedNames.csv**