|  |  |
| --- | --- |
| Language Transliteration - Solution Description  Language Transliteration is a Deep learning approach of transliterating a text in one language into another language for Indian languages Applicable for Solutions Sets <clarifies/lists the applicable solution sets>  No valid solution set found Type <specifies type: keep only the applicable type and delete the others in following list. For more detailed type description, reference the concepts page in TCM methodology Guide [here](https://stbeehive.oracle.com/content/dav/st/TCM_HTML/Public%20Documents/html/concept_assets.html), that covers blueprints, solution sets and solutions.>  Integration (application)  Localisation (geographic) Purpose This an implementation of Language Transliteration i.e. transforms the text in a local indic language and get it transliterate in to English, saying we need to train the model with desired language, this is genric code which could be trained in local specific language with pair of texts on both language, solution takes input text in the local language and get it translate in to English. Purpose of this need is we have to transliterate the Named Entities from one language to another without going in to meaning of that. Scope Right now solution is being trained and tested in indic languages, which could map character to character to English and could be easily adapt for all such languages, which support character-to-character mapping. Technology/pre-requisites In order to enable this Solution, following technology must be available:   * Python 3.6 * Tensorflow 1.14 * Keras * Jupyter Notebook(Anaconda)   In addition, following pre-requisites must be fulfilled:   * System must have installed python , tensorflow and Jupyter notebook * Not mandatory, model could be trained on GPU more efficiently if the underline hardware has GPU installed for that CUDA and CuDNN need to be installed. * If you are training on GPU version, install GPU supported tensorflow using pip install tensorflow-gpu  Deployment duration This is Jupyter notebook source code, from deployment perspective we have to install Anaconda => Create Environment => Install Python => Install tensorflow, once these installation is done we can open up the notebook and train the model.  Training time varies from size of dataset and whether trained on CPU or GPU | **COUNTRIES SUPPORTED BY THIS SOLUTION:**  India  **LANGUages supported by this solution:**  Python  **TCM DOCUMENT ID:**  <SOL\_3461\_001\_Solution\_Description>  **last updated:**  Date: 07/11/2019 |
| Detailed description This is source of Language Literation model which is being written in python using deep learning framework tensor flow and keras.  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  Algorithm/ Approach used Sequence to Sequence Encoder Decoder LSTM Recurrent Neural Network, in this we created two RNN, the encoder network is that part of the network that takes the input sequence and maps it to an encoded representation of the sequence. The encoded representation is then used by the decoder network to generate an output sequence.  Once the model is being trained to form a encoder and decoder network, we can give input text which generates the output text.  This model takes input the form of a file with pair of text in both language in a line with many such entries and gives the out exported in a file, |
| Supporting document links No supporting document part from Asset |

Appendix 1 – Additional Information required to publish IP

# Short descriptive text for the Innovation Portal

This an implementation of Language Transliteration i.e. transforms the text in a local indic language and get it transliterate in to English, saying we need to train the model with desired language, this is genric code which could be trained in local specific language with pair of texts on both language, solution takes input text in the local language and get it translate in to English. Purpose of this need is we transliterate the Named Entities from one language to another without going in to meaning of that.

# Long descriptive text for the Innovation Portal

There are several requirements where we need to transform named entities from source language to target language without translating it to the meaning into the target language. For example, we need to transform person name, places, organizations written in Hindi and get it transliterated to English without making any meaning translation

This an implementation of Language Transliteration i.e. transforms the text in a local indic language and get it transliterate in to English, saying we need to train the model with desired language, this is genric code which could be trained in local specific language with pair of texts on both language, solution takes input text in the local language and get it translate in to English. Purpose of this need is we transliterate the Named Entities from one language to another without going in to meaning of that.

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 .

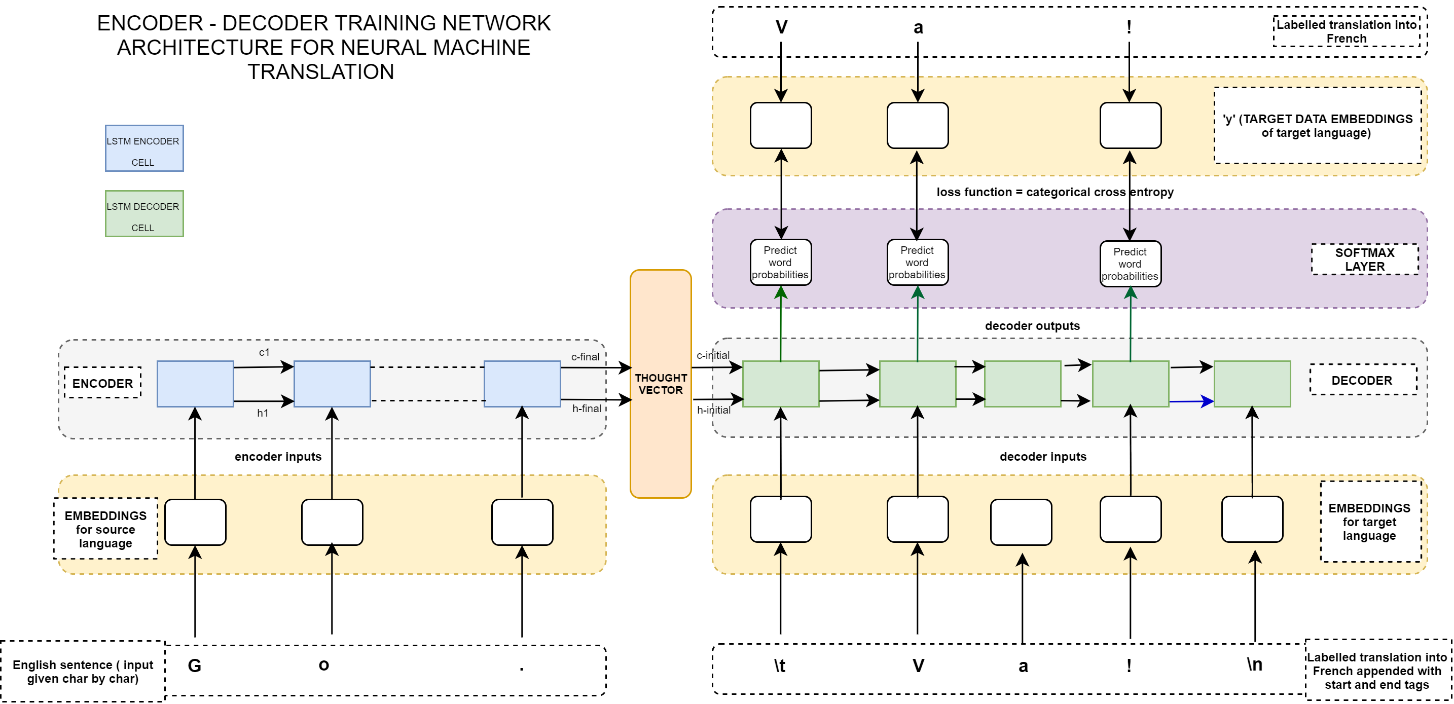
This is source of Language Literation model which is being written in python using deep learning framework tensor flow and keras.

Algorithm/ Approach used Sequence to Sequence Encoder Decoder LSTM Recurrent Neural Network, in this we created two RNN, the encoder network is that part of the network that takes the input sequence and maps it to an encoded representation of the sequence. The encoded representation is then used by the decoder network to generate an output sequence.

Once the model is being trained to form a encoder and decoder network, we can give input text which generates the output text.

This model takes input the form of a file with pair of text in both language in a line with many such entries and gives the out exported in an file,

# ,Insert descriptive picture



Appendix 2 – <name appendix topic>

# Subtopic 1

# Subtopic 2