Introduction

1. Neural Machine Translation (NMT) is the task of using artificial neural network models for translation from one language to the other.
2. The NMT model generally consists of an encoder that encodes a source sentence into a fixed-length vector from which a decoder generates a translation.
3. This problem can be thought as a prediction problem, where given a sequence of words in source language as input, task is to predict the output sequence of words in target language.
4. The dataset comes from http://www.manythings.org/anki/, where you may find tab delimited bilingual sentence pairs in different files based on the source and target language of your choice.
5. For this project, you need to use French - English language pairs just to evaluate the projects uniformly for all students.

Step-1: Download and clean the data

1. Download the data as zip file and extract it to corresponding txt file. Read this txt file and prepare the list of pairs of language phrases.
2. Now, we will nedd to clean these pairs. For cleaning the text, some of the operations for cleaning are:

- Remove the non printable charaters, if any
- Remove punctuations and non-alphabetic charaters
- Convert to lowercase

Step-2: Split and prepare the data for training the model

1. After cleaning the data, next you need to split the data in train and test.
2. Then, you need to create separate tokenizer for both source language and target language.
3. After creating the tokenizer, you need to encode and pad the input (source language) and output(target language) sequences w.r.t. their individual tokenizers and maximum sequence lengths.
4. Here, in this problem you will essentially be predicting the words in target language, therefore output seuences will need to be converted in one hot encoding.

Step-3: Define and train the RNN based Encoder-Decoder model

1. First, you need to define the sequential model consisting mainly of two parts Encoder and Decoder
2. In Encoder, the input sequence shall be passed through an Embedding layer (to train the word embeddings for source language) and then the output from the Embedding layer may be passed through one or more RNN/LSTM layers.
3. Now, to connect this Encoder to Decoder (yet to be defined), we can use RepeatVector layer. (This is because the shape of the output by Encoder is not same as expected shape of Input by Decoder)
4. Now, stack up the Decoder, wherein you may add one or more RNN/LSTM layers and finally the output TimeDistributed Dense layer to get output separately by timesteps.
5. Now, you have defined the model and now this can be trained on the training data, you prepared in last step. Here, you may play with the number of epochs, optimizer, batch size to get the optimum results.

Step-4: Evaluating the model

Use BLEU score for evaluating your model using NLTK library

For .ipynb notebook of this file, right click following link and choose 'save link as'

https://files.codingninjas.in/0000000000003927.ipynb