**Q.** The aim of this challenge is to create a machine learning model that can translate from English to German. Can you convince C-suite of an international company that your custom solution trained on company specific data is better than of-the-shelf solutions from big internet companies? Can you think of possible reasons why a company would hesitate to use a solution from big internet company and prefer an in-house solution? Use them together with results from your model to convince them to use your solution

**The objective is to convert an English sentence to its German counterpart using a Neural Machine Translation (NMT) system. I divided my assignment into different notebooks. If all work is done in one file, then it may get very memory heavy and it is not easy to make changes. I will explain content and role of each notebook briefly.**

# **1-Work 0:**

# **2-Combine:**

# **3-Preprocessing:**

The data we work with is more often than not unstructured so there are certain things we need to take care of before jumping to the model building part.

***3.1)-Text Cleaning***

* get rid of the punctuation marks and then convert all the text to lower case.

***3.2)-Text to Sequence Conversion***

* A Seq2Seq model requires that we convert both the input and the output sentences into integer sequences of fixed length.
* vectorize our text data by using Keras’s Tokenizer () class. It will turn our sentences into sequences of integers. We can then pad those sequences with zeros to make all the sequences of the same length.

# **4-Applied Models**

Sequence-to-Sequence (seq2seq) models are used for a variety of NLP tasks, such as text summarization, speech recognition, DNA sequence modeling, among others. Our aim is to translate given sentences from one language to another. We used seq2seq model for our problem as well.

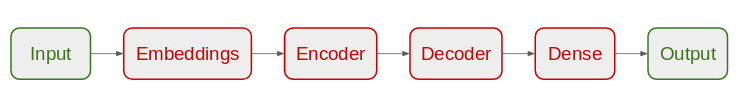
## ***4a)-Simple Model:***

**A typical seq2seq model has 2 major components.**

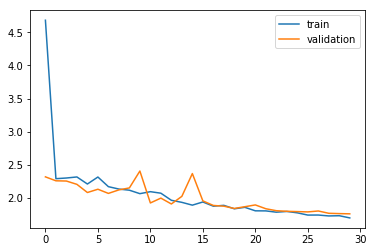
a) an encoder  
b) a decoder

Both these parts are essentially two different recurrent neural network (RNN) models combined into one giant network. Our simple model architecture consists of following;

* For the encoder, we will use an embedding layer and an LSTM layer
* For the decoder, we will use another LSTM layer followed by a dense layer



**This model gave not very impressive results implying that we may need to develop a better model.**

****

we have used ‘sparse\_categorical\_crossentropy‘as the loss function. This is because the function allows us to use the target sequence as is, instead of the one-hot encoded format**. One-hot encoding the target sequences using such a huge vocabulary might consume our system’s entire memory.**

## ***4b)-Encoder-decoder model***