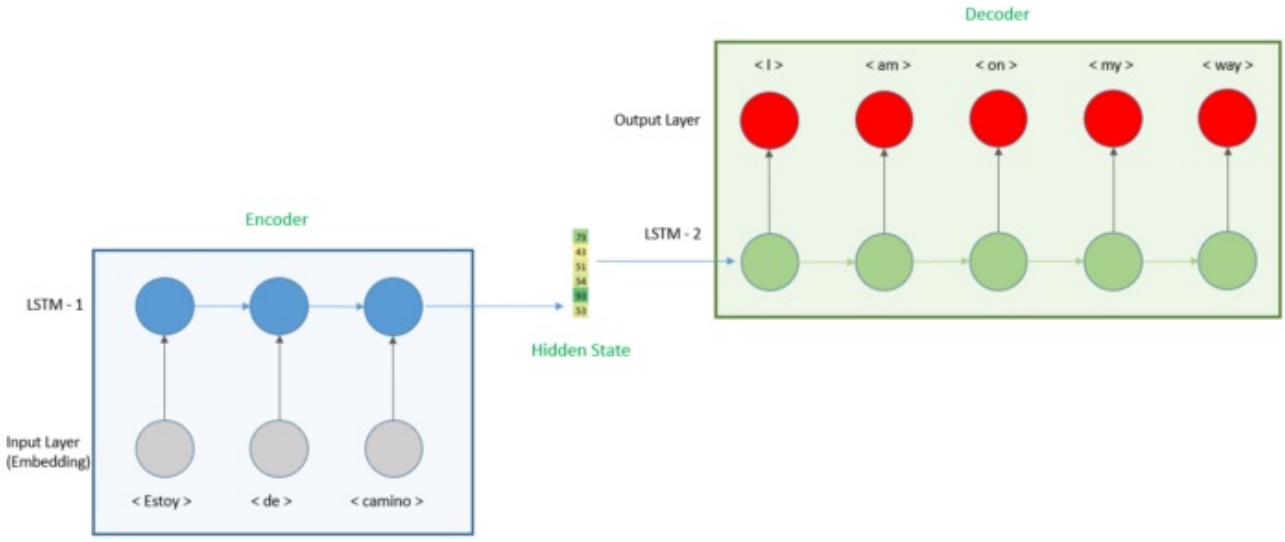
Assessment 6

GitHub Link: https://github.com/manansuthar55/CSE6037_20MAI0016/tree/main/Assessment_6

Problem : Encoder Decoder

Encoder Decoder translation model using LSTM with Python and Keras



In [1]:

```
import string
import numpy as np
from keras.preprocessing.text import Tokenizer
from keras.preprocessing.sequence import pad_sequences
from keras.models import Model
from keras.layers import LSTM, Input, TimeDistributed, Dense, Activation, RepeatVector,
Embedding
from keras.optimizers import Adam
from keras.losses import sparse_categorical_crossentropy

# Path to translation file
path_to_data = '/content/drive/MyDrive/Colab Notebooks/DL_LAB_6/spa.txt'

# Read file
translation_file = open(path_to_data,"r", encoding='utf-8')
raw_data = translation_file.read()
translation_file.close()

# Parse data
raw_data = raw_data.split('\n')
pairs = [sentence.split('\t') for sentence in  raw_data]
pairs = pairs[1000:20000]
```

In [2]:

```
def clean_sentence(sentence):
    # Lower case the sentence
    lower_case_sent = sentence.lower()
    # Strip punctuation
    string_punctuation = string.punctuation + "¡" + '¿'
    clean_sentence = lower_case_sent.translate(str.maketrans('', '', string_punctuation)
)

    return clean_sentence
```

In [3]:

```
def tokenize(sentences):
    # Create tokenizer
    text_tokenizer = Tokenizer()
    # Fit texts
    text_tokenizer.fit_on_texts(sentences)
    return text_tokenizer.texts_to_sequences(sentences), text_tokenizer
```

In [4]:

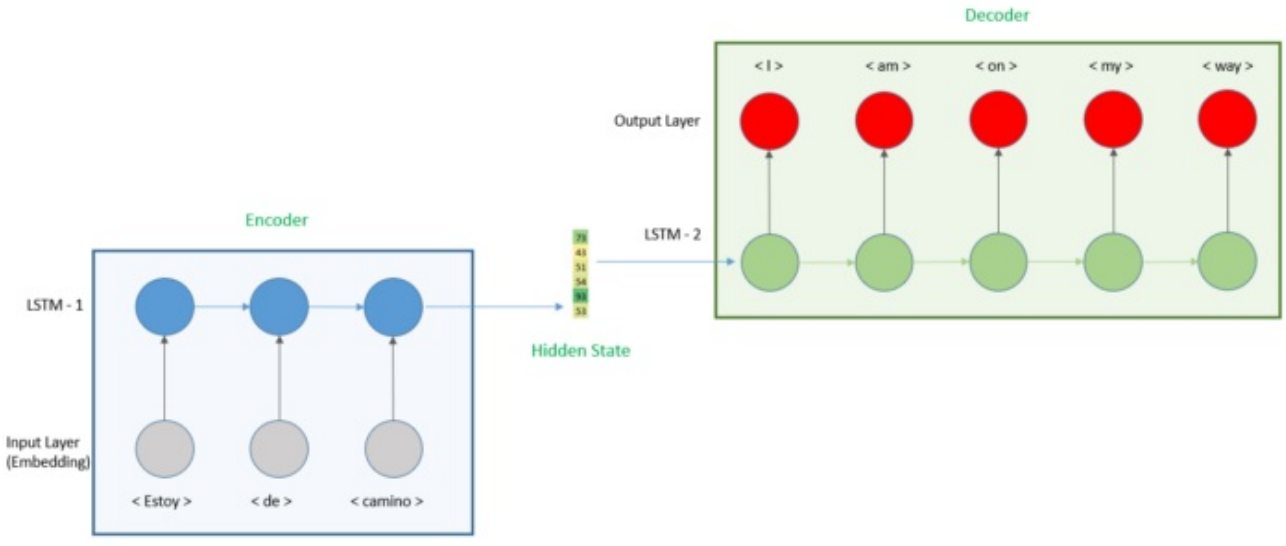
```
english_sentences = [clean_sentence(pair[0]) for pair in pairs]
spanish_sentences = [clean_sentence(pair[1]) for pair in pairs]

# Tokenize words
spa_text_tokenized, spa_text_tokenizer = tokenize(spanish_sentences)
eng_text_tokenized, eng_text_tokenizer = tokenize(english_sentences)

print('Maximum length spanish sentence: {}'.format(len(max(spa_text_tokenized,key=len))))
print('Maximum length english sentence: {}'.format(len(max(eng_text_tokenized,key=len))))


# Check language length
spanish_vocab = len(spa_text_tokenizer.word_index) + 1
english_vocab = len(eng_text_tokenizer.word_index) + 1
print("Spanish vocabulary is of {} unique words".format(spanish_vocab))
print("English vocabulary is of {} unique words".format(english_vocab))
```

```
Maximum length spanish sentence: 15
Maximum length english sentence: 6
Spanish vocabulary is of 7248 unique words
English vocabulary is of 3756 unique words
```

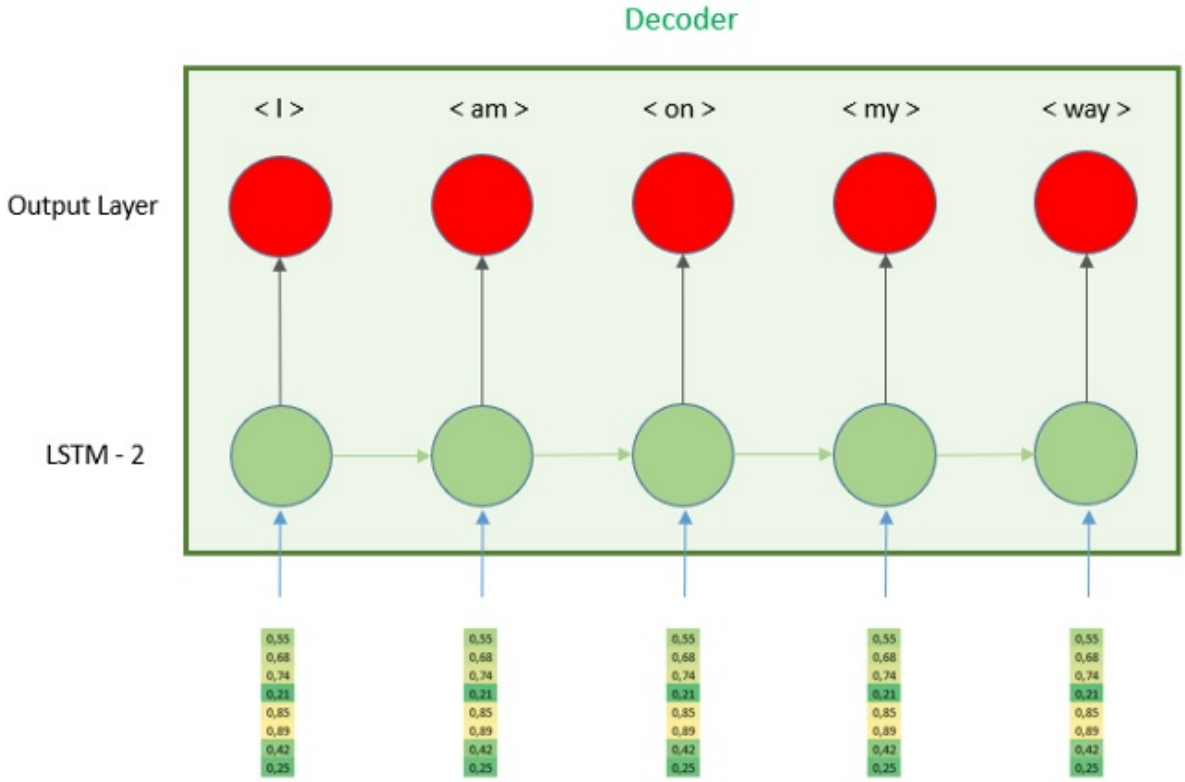
In [5]:

```
max_spanish_len = int(len(max(spa_text_tokenized,key=len)))
max_english_len = int(len(max(eng_text_tokenized,key=len)))

spa_pad_sentence = pad_sequences(spa_text_tokenized, max_spanish_len, padding = "post")
eng_pad_sentence = pad_sequences(eng_text_tokenized, max_english_len, padding = "post")

# Reshape data
spa_pad_sentence = spa_pad_sentence.reshape(*spa_pad_sentence.shape, 1)
eng_pad_sentence = eng_pad_sentence.reshape(*eng_pad_sentence.shape, 1)
```

Encoder



Decoder



In [6]:

```
input_sequence = Input(shape=(max_spanish_len,))
embedding = Embedding(input_dim=spanish_vocab, output_dim=128,)(input_sequence)
encoder = LSTM(64, return_sequences=False)(embedding)
r_vec = RepeatVector(max_english_len)(encoder)
decoder = LSTM(64, return_sequences=True, dropout=0.2)(r_vec)
logits = TimeDistributed(Dense(english_vocab))(decoder)
```

In [7]:

```
enc_dec_model = Model(input_sequence, Activation('softmax')(logits))
enc_dec_model.compile(loss=sparse_categorical_crossentropy,
              optimizer=Adam(1e-3),
              metrics=['accuracy'])
enc_dec_model.summary()
```

```
Model: "model"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
input_1 (InputLayer)         [(None, 15)]              0
_________________________________________________________________
embedding (Embedding)        (None, 15, 128)           927744
_________________________________________________________________
lstm (LSTM)                  (None, 64)                49408
_________________________________________________________________
repeat_vector (RepeatVector) (None, 6, 64)             0
_________________________________________________________________
lstm_1 (LSTM)                (None, 6, 64)             33024
_________________________________________________________________
time_distributed (TimeDistri (None, 6, 3756)           244140
_________________________________________________________________
activation (Activation)      (None, 6, 3756)           0
=================================================================
Total params: 1,254,316
Trainable params: 1,254,316
Non-trainable params: 0
_________________________________________________________________
```

In [8]:

```
model_results = enc_dec_model.fit(spa_pad_sentence, eng_pad_sentence, batch_size=30, epochs=100)
```

```
Epoch 1/100
634/634 [==============================] - 37s 53ms/step - loss: 4.7491 - accuracy: 0.4344
Epoch 2/100
634/634 [==============================] - 32s 50ms/step - loss: 3.4452 - accuracy: 0.4782
Epoch 3/100
634/634 [==============================] - 30s 48ms/step - loss: 3.3563 - accuracy: 0.4806
Epoch 4/100
634/634 [==============================] - 30s 47ms/step - loss: 3.3181 - accuracy: 0.4815
Epoch 5/100
634/634 [==============================] - 31s 50ms/step - loss: 3.3074 - accuracy: 0.4798
Epoch 6/100
634/634 [==============================] - 30s 48ms/step - loss: 3.3437 - accuracy: 0.4784
Epoch 7/100
634/634 [==============================] - 31s 48ms/step - loss: 3.3204 - accuracy: 0.4787
Epoch 8/100
634/634 [==============================] - 31s 49ms/step - loss: 3.2840 - accuracy: 0.4800
Epoch 9/100
634/634 [==============================] - 31s 49ms/step - loss: 3.2803 - accuracy: 0.4790
Epoch 10/100
634/634 [==============================] - 34s 53ms/step - loss: 3.2645 - accuracy: 0.4814
Epoch 11/100
634/634 [==============================] - 30s 47ms/step - loss: 3.2752 - accuracy: 0.4796
Epoch 12/100
634/634 [==============================] - 31s 49ms/step - loss: 3.2622 - accuracy: 0.4794
Epoch 13/100
634/634 [==============================] - 30s 48ms/step - loss: 3.1558 - accuracy: 0.4891
Epoch 14/100
634/634 [==============================] - 29s 45ms/step - loss: 3.0630 - accuracy: 0.4942
Epoch 15/100
634/634 [==============================] - 31s 48ms/step - loss: 2.9922 - accuracy: 0.5018
Epoch 16/100
634/634 [==============================] - 32s 51ms/step - loss: 2.8537 - accuracy: 0.5170
Epoch 17/100
634/634 [==============================] - 30s 47ms/step - loss: 2.7185 - accuracy: 0.5308
Epoch 18/100
634/634 [==============================] - 30s 48ms/step - loss: 2.6108 - accuracy: 0.5393
Epoch 19/100
634/634 [==============================] - 31s 48ms/step - loss: 2.5196 - accuracy: 0.5503
Epoch 20/100
634/634 [==============================] - 30s 47ms/step - loss: 2.4168 - accuracy: 0.5647
Epoch 21/100
634/634 [==============================] - 32s 50ms/step - loss: 2.3276 - accuracy: 0.5748
Epoch 22/100
634/634 [==============================] - 30s 47ms/step - loss: 2.2375 - accuracy: 0.5883
Epoch 23/100
634/634 [==============================] - 29s 46ms/step - loss: 2.1525 - accuracy: 0.5985
```

```
Epoch 24/100
634/634 [==============================] - 30s 47ms/step - loss: 2.0834 - accuracy: 0.607
6
Epoch 25/100
634/634 [==============================] - 30s 47ms/step - loss: 2.0209 - accuracy: 0.614
4
Epoch 26/100
634/634 [==============================] - 30s 48ms/step - loss: 1.9480 - accuracy: 0.623
2
Epoch 27/100
634/634 [==============================] - 29s 46ms/step - loss: 1.8919 - accuracy: 0.631
5
Epoch 28/100
634/634 [==============================] - 30s 48ms/step - loss: 1.8341 - accuracy: 0.638
8
Epoch 29/100
634/634 [==============================] - 32s 50ms/step - loss: 1.7761 - accuracy: 0.645
0
Epoch 30/100
634/634 [==============================] - 33s 53ms/step - loss: 1.7166 - accuracy: 0.654
5
Epoch 31/100
634/634 [==============================] - 30s 47ms/step - loss: 1.6625 - accuracy: 0.661
3
Epoch 32/100
634/634 [==============================] - 30s 48ms/step - loss: 1.6141 - accuracy: 0.668
2
Epoch 33/100
634/634 [==============================] - 30s 47ms/step - loss: 1.5677 - accuracy: 0.673
2
Epoch 34/100
634/634 [==============================] - 32s 51ms/step - loss: 1.5198 - accuracy: 0.680
6
Epoch 35/100
634/634 [==============================] - 30s 48ms/step - loss: 1.4755 - accuracy: 0.685
2
Epoch 36/100
634/634 [==============================] - 31s 49ms/step - loss: 1.4312 - accuracy: 0.693
3
Epoch 37/100
634/634 [==============================] - 29s 46ms/step - loss: 1.3918 - accuracy: 0.700
0
Epoch 38/100
634/634 [==============================] - 32s 50ms/step - loss: 1.3465 - accuracy: 0.704
0
Epoch 39/100
634/634 [==============================] - 30s 48ms/step - loss: 1.3124 - accuracy: 0.710
7
Epoch 40/100
634/634 [==============================] - 32s 50ms/step - loss: 1.2643 - accuracy: 0.720
5
Epoch 41/100
634/634 [==============================] - 31s 48ms/step - loss: 1.2356 - accuracy: 0.724
4
Epoch 42/100
634/634 [==============================] - 31s 49ms/step - loss: 1.2019 - accuracy: 0.728
2
Epoch 43/100
634/634 [==============================] - 31s 48ms/step - loss: 1.1565 - accuracy: 0.738
7
Epoch 44/100
634/634 [==============================] - 30s 47ms/step - loss: 1.1430 - accuracy: 0.740
4
Epoch 45/100
634/634 [==============================] - 30s 48ms/step - loss: 1.1016 - accuracy: 0.748
5
Epoch 46/100
634/634 [==============================] - 32s 50ms/step - loss: 1.0725 - accuracy: 0.754
7
Epoch 47/100
634/634 [==============================] - 31s 49ms/step - loss: 1.0438 - accuracy: 0.757
9
```

```
Epoch 48/100
634/634 [==============================] - 32s 50ms/step - loss: 1.0104 - accuracy: 0.764
2
Epoch 49/100
634/634 [==============================] - 32s 51ms/step - loss: 0.9892 - accuracy: 0.769
8
Epoch 50/100
634/634 [==============================] - 31s 48ms/step - loss: 0.9593 - accuracy: 0.775
9
Epoch 51/100
634/634 [==============================] - 31s 49ms/step - loss: 0.9306 - accuracy: 0.782
0
Epoch 52/100
634/634 [==============================] - 31s 49ms/step - loss: 0.9180 - accuracy: 0.785
3
Epoch 53/100
634/634 [==============================] - 31s 48ms/step - loss: 0.8877 - accuracy: 0.790
2
Epoch 54/100
634/634 [==============================] - 31s 48ms/step - loss: 0.8587 - accuracy: 0.797
4
Epoch 55/100
634/634 [==============================] - 31s 49ms/step - loss: 0.8375 - accuracy: 0.801
5
Epoch 56/100
634/634 [==============================] - 31s 50ms/step - loss: 0.8164 - accuracy: 0.806
0
Epoch 57/100
634/634 [==============================] - 30s 48ms/step - loss: 0.7897 - accuracy: 0.810
7
Epoch 58/100
634/634 [==============================] - 31s 48ms/step - loss: 0.7807 - accuracy: 0.814
9
Epoch 59/100
634/634 [==============================] - 31s 49ms/step - loss: 0.7587 - accuracy: 0.819
4
Epoch 60/100
634/634 [==============================] - 31s 49ms/step - loss: 0.7383 - accuracy: 0.822
5
Epoch 61/100
634/634 [==============================] - 32s 51ms/step - loss: 0.7281 - accuracy: 0.825
7
Epoch 62/100
634/634 [==============================] - 32s 50ms/step - loss: 0.6978 - accuracy: 0.832
9
Epoch 63/100
634/634 [==============================] - 33s 52ms/step - loss: 0.6797 - accuracy: 0.836
1
Epoch 64/100
634/634 [==============================] - 30s 47ms/step - loss: 0.6681 - accuracy: 0.836
8
Epoch 65/100
634/634 [==============================] - 32s 50ms/step - loss: 0.6573 - accuracy: 0.840
3
Epoch 66/100
634/634 [==============================] - 31s 50ms/step - loss: 0.6391 - accuracy: 0.845
1
Epoch 67/100
634/634 [==============================] - 31s 50ms/step - loss: 0.6294 - accuracy: 0.846
0
Epoch 68/100
634/634 [==============================] - 32s 50ms/step - loss: 0.6045 - accuracy: 0.853
8
Epoch 69/100
634/634 [==============================] - 32s 50ms/step - loss: 0.5898 - accuracy: 0.854
9
Epoch 70/100
634/634 [==============================] - 33s 52ms/step - loss: 0.5837 - accuracy: 0.856
0
Epoch 71/100
634/634 [==============================] - 32s 50ms/step - loss: 0.5751 - accuracy: 0.859
3
```

```
Epoch 72/100
634/634 [==============================] - 32s 51ms/step - loss: 0.5489 - accuracy: 0.865
6
Epoch 73/100
634/634 [==============================] - 31s 48ms/step - loss: 0.5480 - accuracy: 0.865
3
Epoch 74/100
634/634 [==============================] - 32s 50ms/step - loss: 0.5350 - accuracy: 0.865
7
Epoch 75/100
634/634 [==============================] - 32s 50ms/step - loss: 0.5221 - accuracy: 0.870
2
Epoch 76/100
634/634 [==============================] - 33s 52ms/step - loss: 0.5135 - accuracy: 0.871
4
Epoch 77/100
634/634 [==============================] - 32s 51ms/step - loss: 0.5013 - accuracy: 0.874
2
Epoch 78/100
634/634 [==============================] - 32s 51ms/step - loss: 0.4961 - accuracy: 0.875
1
Epoch 79/100
634/634 [==============================] - 32s 51ms/step - loss: 0.4913 - accuracy: 0.875
2
Epoch 80/100
634/634 [==============================] - 32s 51ms/step - loss: 0.4722 - accuracy: 0.882
3
Epoch 81/100
634/634 [==============================] - 30s 48ms/step - loss: 0.4696 - accuracy: 0.879
8
Epoch 82/100
634/634 [==============================] - 30s 48ms/step - loss: 0.4528 - accuracy: 0.883
0
Epoch 83/100
634/634 [==============================] - 31s 48ms/step - loss: 0.4531 - accuracy: 0.883
5
Epoch 84/100
634/634 [==============================] - 31s 48ms/step - loss: 0.4483 - accuracy: 0.885
2
Epoch 85/100
634/634 [==============================] - 33s 52ms/step - loss: 0.4371 - accuracy: 0.886
8
Epoch 86/100
634/634 [==============================] - 34s 53ms/step - loss: 0.4287 - accuracy: 0.889
5
Epoch 87/100
634/634 [==============================] - 32s 51ms/step - loss: 0.4271 - accuracy: 0.889
7
Epoch 88/100
634/634 [==============================] - 32s 51ms/step - loss: 0.4148 - accuracy: 0.891
9
Epoch 89/100
634/634 [==============================] - 34s 53ms/step - loss: 0.4017 - accuracy: 0.895
2
Epoch 90/100
634/634 [==============================] - 33s 52ms/step - loss: 0.3979 - accuracy: 0.897
2
Epoch 91/100
634/634 [==============================] - 33s 52ms/step - loss: 0.3917 - accuracy: 0.896
8
Epoch 92/100
634/634 [==============================] - 34s 53ms/step - loss: 0.3878 - accuracy: 0.897
3
Epoch 93/100
634/634 [==============================] - 32s 51ms/step - loss: 0.3781 - accuracy: 0.900
6
Epoch 94/100
634/634 [==============================] - 33s 53ms/step - loss: 0.3784 - accuracy: 0.898
3
Epoch 95/100
634/634 [==============================] - 31s 49ms/step - loss: 0.3689 - accuracy: 0.901
9
```

```
Epoch 96/100
634/634 [==============================] - 31s 49ms/step - loss: 0.3649 - accuracy: 0.903
5
Epoch 97/100
634/634 [==============================] - 30s 47ms/step - loss: 0.3577 - accuracy: 0.904
5
Epoch 98/100
634/634 [==============================] - 29s 46ms/step - loss: 0.3549 - accuracy: 0.905
0
Epoch 99/100
634/634 [==============================] - 32s 50ms/step - loss: 0.3547 - accuracy: 0.903
2
Epoch 100/100
634/634 [==============================] - 31s 49ms/step - loss: 0.3477 - accuracy: 0.906
4
```

In [9]:

```
def logits_to_sentence(logits, tokenizer):
    index_to_words = {idx: word for word, idx in tokenizer.word_index.items()}
    index_to_words[0] = '<empty>'
    return ' '.join([index_to_words[prediction] for prediction in np.argmax(logits, 1)])

index = 14
print("The english sentence is: {}".format(english_sentences[index]))
print("The spanish sentence is: {}".format(spanish_sentences[index]))
print('The predicted sentence is :')
print(logits_to_sentence(enc_dec_model.predict(spa_pad_sentence[index:index+1])[0], eng_
text_tokenizer))
```

```
The english sentence is: whats up
The spanish sentence is: qué hay
The predicted sentence is :
whats is <empty> <empty> <empty> <empty>
```