



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA



Máster en Inteligencia Artificial, Reconocimiento de Formas e Imagen Digital
Universitat Politècnica de València

Traducción basada en redes neuronales recurrentes

Traducción Automática

Autor: Juan Antonio López Ramírez

Curso 2019-2020

Introducción

El objetivo de esta práctica es construir sistemas de traducción basado en redes neuronales a partir de conjuntos de pares de frases bilingües. Para ello utilizaremos la herramienta *nmt-keras*, basada en Tensorflow y Keras.

Experimentos y resultados obtenidos

Ejercicio 1

Para este ejercicio se han probado distintos tamaños de embedding para la capa de entrada del encoder, además de los embeddings del decoder.

Los resultados obtenidos se pueden apreciar en la siguiente tabla:

Embedding de entrada	Target Embedding	BLEU obtenido
32	32	90.35
64	64	94.8
128	128	98.3
256	256	98.49
512	512	97.32

Como se puede observar, los mejores resultados se obtienen con un tamaño de embedding de 256, en el que el BLEU es de **98.49**. También apreciamos un decremento del BLEU cuando el Embedding es de tamaño 512, esto puede ser debido a que el conjunto de training es muy pequeño para entrenar embeddings que capturen la sintaxis y la semántica de los términos.

Ejercicio 2

Como en el anterior ejercicio obteníamos el mejor resultado con un embedding de tamaño 256, hemos probado otros tamaños para la red recurrente (modelo LSTM) con la entrada y salida a 256.

Los resultados obtenidos se pueden apreciar en la siguiente tabla:

LSTM de entrada	LSTM de salida	BLEU obtenido
32	32	96.45
64	64	98.50
128	128	95.30

Como se puede observar, el mejor BLEU, de **98.50**, se obtiene con 64 unidades LSTM. Un aumento en el número de unidades supone un decremento en el BLEU, lo que posiblemente se debe a que no existen suficientes datos como para realizar un buen entrenamiento y no caer en un sobreajuste.

Una vez realizados estos dos primeros ejercicios y viendo los resultados conseguidos, configuramos el modelo con embedding de tamaño 256 y 64 unidades de LSTM para entrenar el modelo hasta que el BLEU alcance el 100 % en training. El proceso ha convergido al llegar al epoch 25 y, tras realizar una evaluación sobre el conjunto de prueba, se ha obtenido un BLEU de **99.76**.

Ejercicio 3

En este ejercicio se pedían probar otros algoritmos de aprendizaje, como el Adagrad o el Adadelta. En nuestro caso, ambos han sido evaluados con 10 epochs. A nivel de coste temporal, la convergencia se obtiene de manera más lenta que su contraparte en ADAM y, a nivel de eficiencia, el BLEU que se obtiene es de **11.9** para Adagrad y **1.5** para Adadelta.

Ejercicio 4

Para probar el Transformer, hemos tenido que ajustar el tamaño del modelo y el de los embeddings para que ambos coincidan.

Los resultados obtenidos se pueden apreciar en la siguiente tabla:

Tamaño del modelo	BLEU obtenido
32	64.25
64	87.40
128	92.50
256	88.14

Como se puede apreciar, el modelo basado en transformer obtiene el mejor resultado cuando es de tamaño 128, consiguiendo un BLEU de **92.50**, lo que hace que no supere al modelo de Encoder-Decoder que habíamos planteado anteriormente.