.UBAfiuba
FACULTAD DE INGENIERÍA

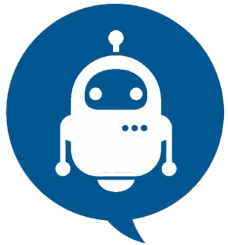
NLP

Sequence to Sequence (seq2seq)

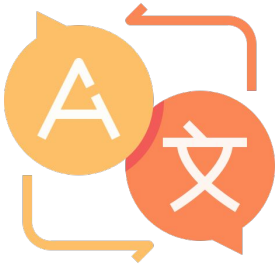
Dr. Rodrigo Cardenas Szigety
rodrigo.cardenas.sz@gmail.com

Soluciones Seq2Seq

Trabaja principalmente con el concepto many-to-many en formato "codificador" a "decodificador", en donde la secuencia de entrada se codifica a una representación intermedia y se decodifica al espacio de salida. **Es útil para traducir o transferir representaciones entre distintos dominios de datos o modalidades.**

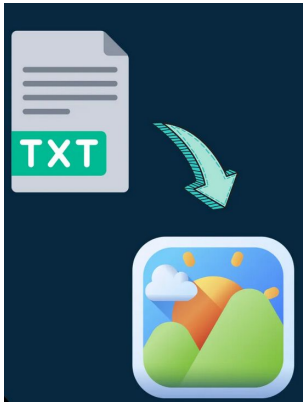


Bots Conversacionales (modelos de lenguaje)



Traducción de idiomas

Completar una imagen

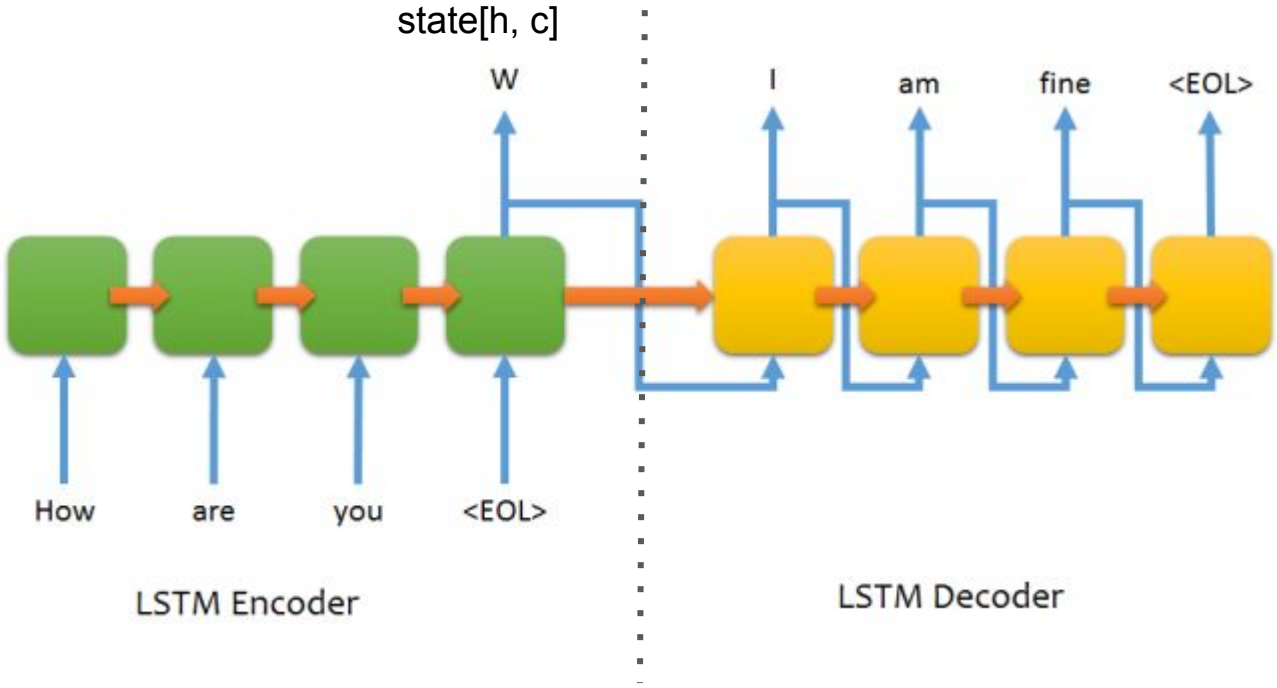


Text2img

Seq2Seq con encoder-decoder

"Modelo basado en dos partes, la primera genera un “espacio latente” o “contexto” que alimenta a la segunda parte, la cual realiza una inferencia realimentada de la última salida." (simil one-to-many)

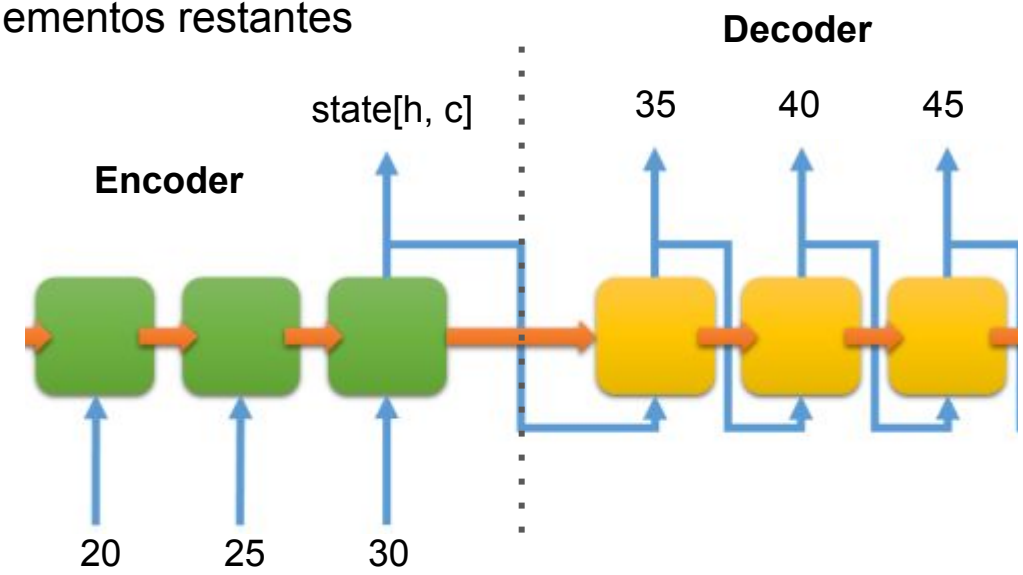
La primera inferencia depende del encoder y su estado final reemplaza en el decoder el estado inicial h_{t0}. En el decoder el modelo es auto-regresivo.



Encoder-Decoder en secuencia numérica

Al igual que en el ejemplo de many to many, el encoder procesará toda la secuencia de entrada produciendo un estado oculto que se pasará como primer estado al decoder.

El decoder utiliza ese estado oculto y su propia realimentación de salida para producir los elementos restantes



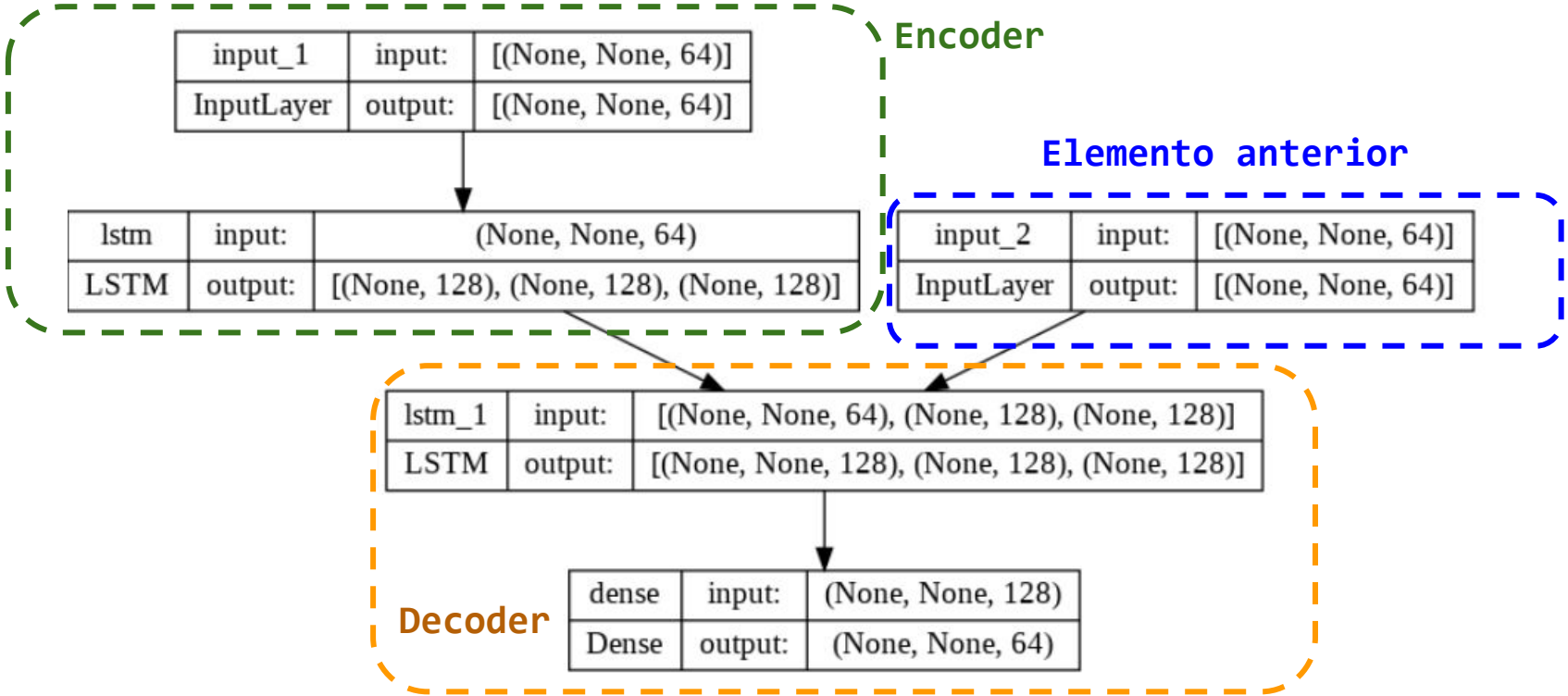
***Nota**

Tener en cuenta que el encoder-decoder recibirá un vector onehotEncoding que represente (embedding) a los números, ya que el espacio de posibilidades debe ser acotado (discreto)

LSTM encoder-decoder

LINK

El modelo que se entrena es el "completo", con el encoder y decoder.

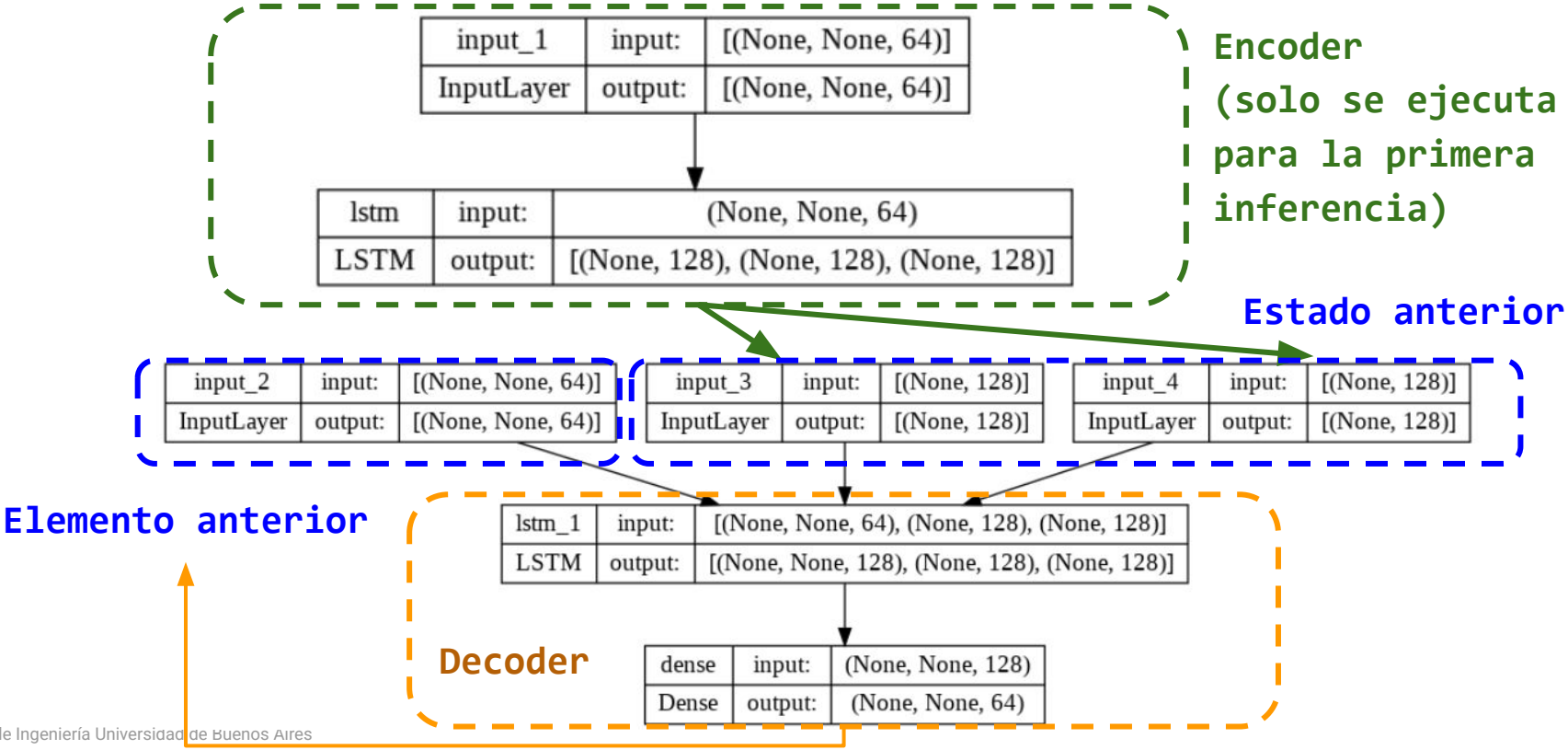


LSTM Decoder

LINK

Para la inferencia se utiliza por separado el encoder y el decoder.

El decoder funciona de manera auto-regresiva.



Encoder-decoder

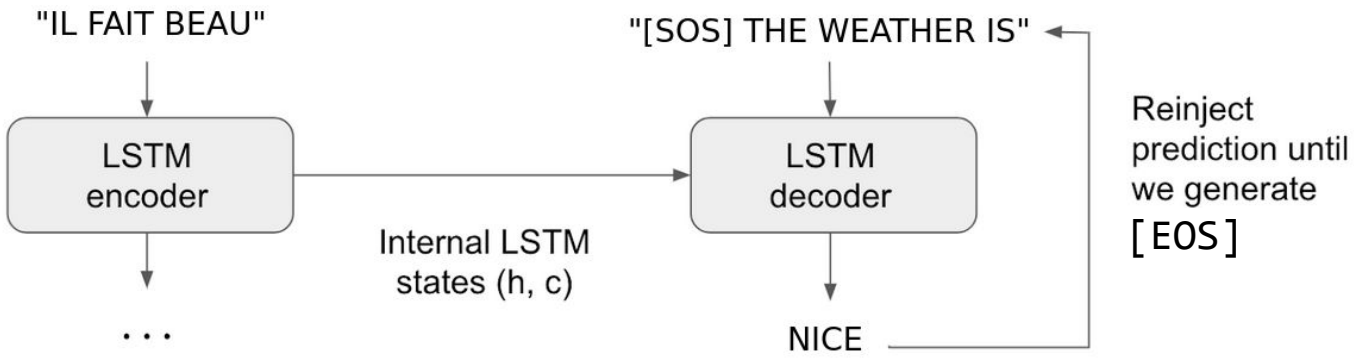
Link al Colab

LINK

NLP Encoder-decoder

LINK

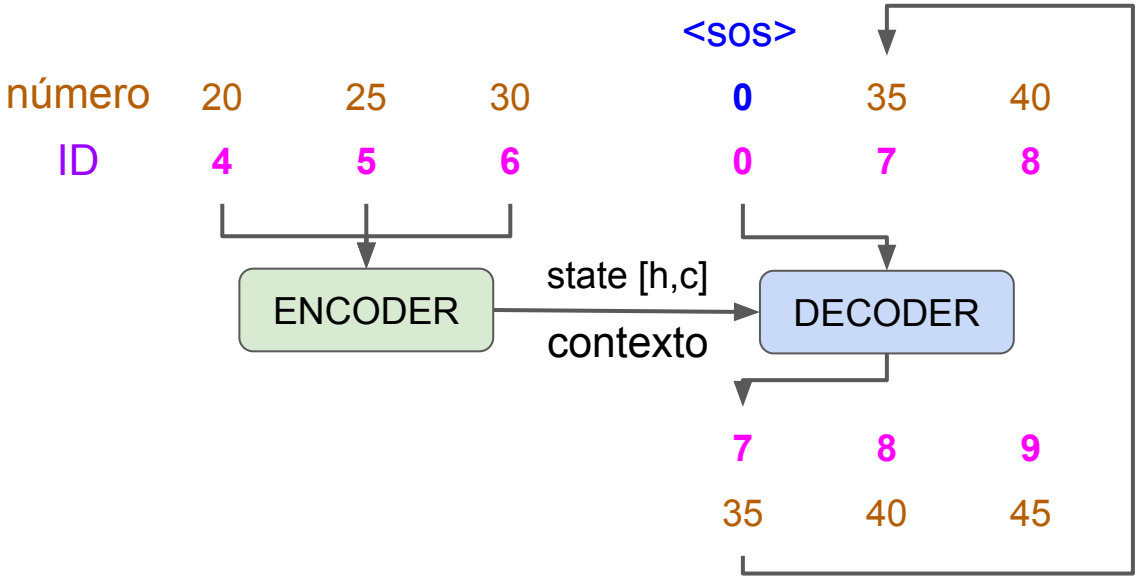
Cuando hablamos de un encoder-decoder NLP se agrega un grado de dificultad más, ya que las secuencias no necesariamente tienen el mismo tamaño y que hay que vectorizar las sentencias de entrada



Para solucionar el problema de secuencias de distinto tamaño se define una máxima longitud y luego se acota con los tokens de inicio y fin de sentencia (<sos>/<eos>)

Tokens especiales <SOS> & <EOS>

Tokens que se reservan para indicarle al modelo el comienzo (Start Of Sequence) o el fin (End Of Sequence) de la secuencia.



Las **palabras/números** se transforman en tokens (**ids**) con el Tokenizador o LabelEncoder

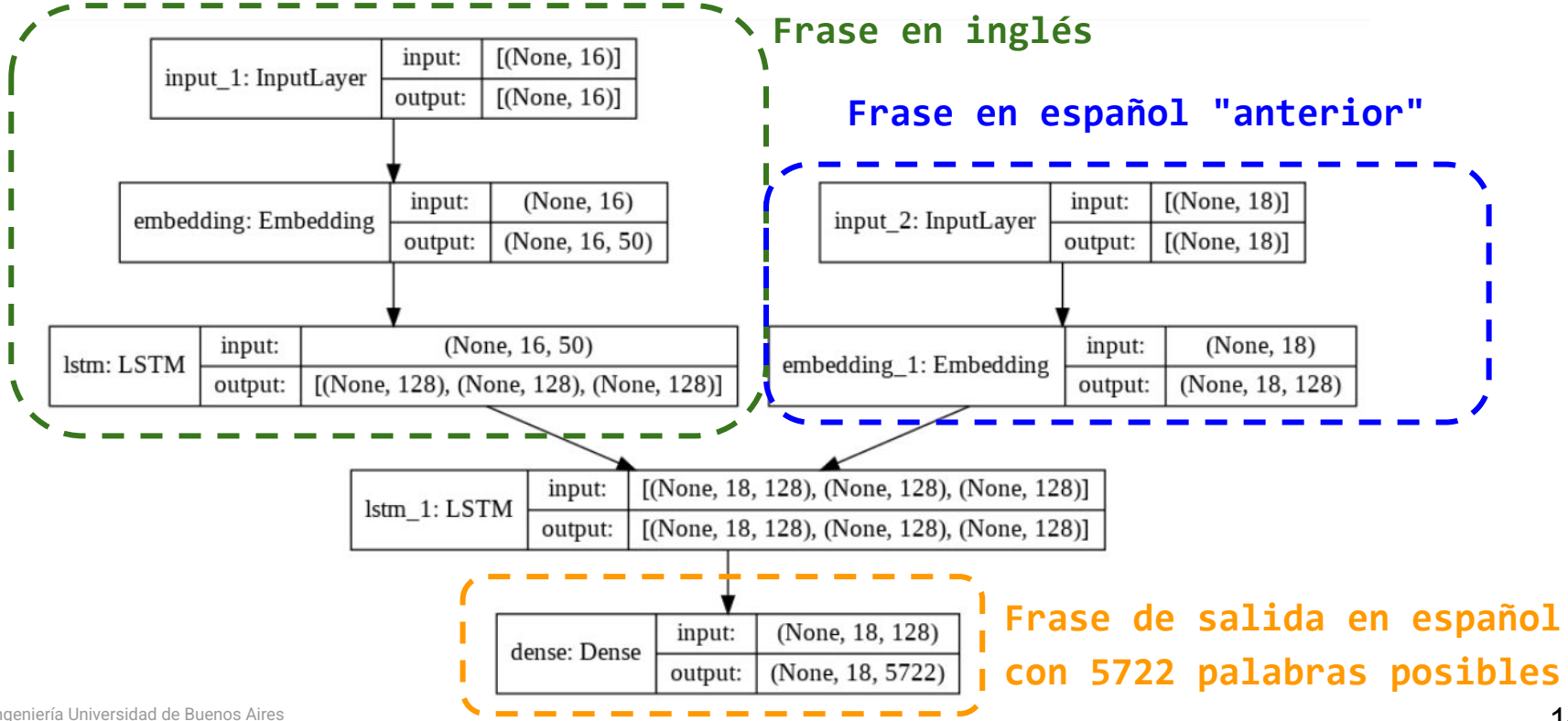
En este ejemplo de secuencia numérica usaremos el número "**0**" como **<sos>**.

El LabelEncoder le dará un **ID** a ese token especial (en este caso también 0)

Es importante que el token esté representado por una palabra/número que no exista en el vocabulario (para no confundirlo)

Traductores

En este ejemplo realizaremos un traductor de inglés a español, vectorizando las sentencias de entrada con Embeddings



Inferencia del traductor

El encoder inicializa el contexto (h1,c1) con la entrada del decoder en <sos>, luego la salida es realimentada.

```
('A deal is a deal.',
 'Un trato es un trato. <eos>',
 '<sos> Un trato es un trato.')
```

```
Input: Tom is naked.
Response: tom es un noche
```

Ensayo real, formó una oración coherente pero no era el resultado solicitado

```
Step 1:
A deal is a deal -> Encoder -> enc(h1,c1)

enc(h1,c1) + <sos> -> Decoder -> Un + dec(h1,c1)

step 2:
dec(h1,c1) + Un -> Decoder -> trato + dec(h2,c2)

step 3:
dec(h2,c2) + trato -> Decoder -> es + dec(h3,c3)

step 4:
dec(h3,c3) + es -> Decoder -> un + dec(h4,c4)

step 5:
dec(h4,c4) + un -> Decoder -> trato + dec(h5,c5)

step 6:
dec(h5,c5) + trato. -> Decoder -> <eos> + dec(h6,c6)
```

Traductor Seq2Seq

Link al Colab

LINK

Desafío 4 (traductor)

Replicar y extender el traductor:

- Replicar el modelo en PyTorch.
- Extender el entrenamiento a más datos y tamaños de secuencias mayores.
- Explorar el impacto de la cantidad de neuronas en las capas recurrentes.
- Mostrar 5 ejemplos de traducciones generadas.

- Extras que se pueden probar: Embeddings pre-entrenados para los dos idiomas; cambiar la estrategia de generación (por ejemplo muestreo aleatorio);